Lwarp

LATEX HTML5

The lwarp package

LATEX to HTML

v0.918 — 2025/01/28

© 2016–2025 Brian Dunn

GitHub: https://github.com/bdtc/lwarp

Abstract

The lwarp package converts LATEX to HTML by using LATEX to process the user's document and directly generate HTML tags. External utility programs are only used for the final conversion of text and images. Math may be represented by SVG images or MATHJAX. More than 500 LATEX packages and classes are supported, of which more than 90 also support MATHJAX.

Documents may be produced by DVI or PDF LATEX, LuaLATEX, XƎLATEX; by several CJK engines, classes, and packages; or by customized systems such as perltex and pythontex. A *texlua* script automates compilation, index, glossary, and batch image processing, and also supports *latexmk*. Configuration is semi-automatic at the first manual compile. Support files are self-generated. Print and HTML versions of each document may coexist.

Assistance is provided for HTML import into EPUB conversion software and word processors.

Requirements include the commonly-available POPPLER utilities (included with MiKTeX) and PERL. Detailed installation instructions are included for each of the major operating systems and TEX distributions.

A quick-start tutorial is provided, as well as extensive documentation for special cases, a general index, and a troubleshooting index. Automatic error testing is provided for configuration files, package load order, and image generation.

SVG math and many other generated images include LATEX expressions in the `alt` tags. MATHJAX may be used with advanced equation numbering under the direct control of lwarp.

Complicated tables are supported, which copy/paste well into LIBREOFFICE WRITER.

Supported classes and packages include memoir and koma-script, cleveref, caption, mdframed, siunitx, and many popular packages for tabulars, floats, graphics, theorems, the title page, bibliography, indexing, footnotes, and editorial work, as well as a number of CJK-related classes and packages.

TEX is a self-modifying tokenized macro-expansion language. Since lwarp is written directly in LATEX, it is able to interpret the document's meaning at a deeper level than external conversions which merely approximate TEX. HTML5 and CSS3 are leveraged to provide advanced features such as booktabs trim, multicolumns, side-by-side minipages, and JAVASCRIPT-free navigation.

For a quick-start tutorial, see section 5, Tutorial.

For a list of supported features, see table 2: Supported packages and features.

To update existing projects, see section 1: Updates.

Need help? See the General Index or the Troubleshooting Index.

Lwarp is still in development. Changes are likely.

Support TEX development

TEX and related projects:

- are mostly open-sourced and a volunteer effort;
- benefit students, academics, scientists, engineers, and businesses;
- help drive education, public and private research, and commercial activity;
- are used in the fields of mathematics, science, engineering, and humanities;
- are international in reach;
- span decades of development;
- are enduring — many older packages are still actively used and maintained;
- are largely backwards compatible;
- are portable across all the major computing platforms;
- are usable even on older computers and away from internet access;
- are continuing to maintain relevance with modern improvements;
- require no yearly subscription fees;
- and are supported by an active community of knowledgeable volunteers.

Please consider helping by joining and/or contributing to the TEX Users Group, a United States 501(c)(3) tax-exempt charitable organization. Contributions are accepted by credit card, check, or Pay Pal, via the United Way, or by USA or European bank transfer. Membership in TUG supports the development of TEXLive, the major TEX distribution.

Donations may be directed towards individual projects:

TUG Bursary Fund: Assistance for attending annual TUG meetings.
CTAN: The Comprehensive TEX Archive Network — Central storage for TEX.
TeX Development Fund: Support for specific projects.
EduTeX: Teaching and using TEX in schools and universities.
GUST e-foundry fonts: Enhanced for math and additional language groups.
LaTeX Project: Modernizing the LATEX core.
Libre Font Fund: Fonts, tools (FontForge), and distribution (the Open Font Library).
LuaTeX: Combining the PDF TEX engine and the Lua language.
MetaPost: Postscript graphics.
MacTeX: TEX for Mac.
PDF Accessibility: Modern PDF standards.
Other: Additional projects may be specified.

To make a contribution: https://www.tug.org/donate.html

For country-specific TEX users groups: http://tug.org/usergroups.html

For users of MiKTEX: https://miktex.org/donations.html

Contents

List of Figures

List of Tables

1 Updates

The following is a summary of updates to lwarp, highlighting new features and any special changes which must be made due to improvements or modifications in lwarp itself.

For a detailed list of the most recent changes, see the end of the Change History on page .

v0.918: Package updates.

- Added an automatic vertical scroll bar to the sidetoc.
- Updated mathtools, mismath, siunitx.

v0.917: Improved `alt` tags.

⚠ New labels
- Due to changes in cross referencing, execute **`lwarpmk clean`** before recompiling.

⚠ New images
- Due to changes in how automatically-generated SVG image file names are computed, after **`lwarpmk html`** use **`lwarpmk cleanlimages`** a single time, and then **`lwarpmk limages`** to generate the new images.
- Improved `alt` tag HTML sanitization.
- Updated backref, extramarks, fancyhdr, fancyvrb, fvextra, lipsum, listings, minted, musicography, orcidlink, pdfpages, siunitx, witharrows, xr.

v0.916: Now allows duplicate section names for file breaks.

- Adjusts file names to allow duplicate section names. See section 8.4.4.
- Fixed LaTeX3 key/value option handling.
- Fixed \<space> at end of a line.

v0.915: HTML list classes, meta tags.

- LaTeX lists now given the class `itemize`, `enumerate`, `description`, `hanging`.
- LaTeX list labels now given the class `listmarker`.
- Added \HTMLKeywords for the keywords meta tag.
- Added \HTMLMeta and \HTMLAddMeta for custom meta tags.
- Added `data-nosnippet` to MathJax customization <div>s.
- Updated pdfpages to v0.5y.

v0.914: Detects changing packages.

- Now verifies many definitions before patching, warning of possible problems if the original has changed.
- Fix: fontawesome5 for XƎLaTeX, LuaLaTeX.

v0.913: HTML sanitization for verbatims.

- Now at GitHub: https://github.com/bdtc/lwarp
- Added bibliography usage info to docs and tutorial.
- \verb now uses a CSS class of verb instead of texttt.
- Improved HTML sanitization for hyperlinks, fancybox, fancyvrb, fvextra, minted.
- Updated fancyvrb, fvextra, simplebnf.
- siunitx: Updated, and improved complex i,j.
- Added doipubmed.

v0.912: Updated for new LATEX label system.

⚠ New labels
- Due to changes in cross referencing, execute **lwarpmk clean** before recompiling.
- Fixed for updated kernel label system, name and back references.
- Updated memoir, tcolorbox.
- nameref: Now allowed to load before lwarp, such as by memoir.

v0.911: Updated mismath, tcolorbox.

v0.910: Updated fvextra, minted.

v0.909: \ref fix.

- Fixed \ref*, beamerarticle, lyluatex, realscripts.
- Updated mismath, nicematrix, pablance, pdfpages, simplebnf, tagpdf.

v0.908: Bug fix.

- Fixed obscure cross-reference issue, seen in some citations.

v0.907: Bug fix.

- Fixed SVG images for WINDOWS.

v0.906: Screen readers

- For each tabular, add a hidden HTML header cell to convince screen readers that the tables are data not layout. Also hide from the screen reader any final row used only to produce bottom borders.
- Adjusted SVG math for a margin change in *pdfcrop*.
- Added \Ref.
- Added docs regarding math in custom environments. See section 8.7.

v0.905: Bug fixes, internal improvements.

- Fixed conflict between cleveref and splitidx.
- Improved coexistence with \AtEndDocument.
- acronym: Updated to v1.47, added hyper links.

v0.904a: Fixed missing lwarp-common-mathjax-siunitx package.

v0.904: Added siunitx v3.

- Fixed HTML tags inside non-Latin text.
- MATHJAX now defaults to SVG rendering.
- Added siunitx v3. Updated siunitx-v2. See section 8.7.15 for limitations.
- Updated caption, chemmacros, fbox, hyperref, multicol, wrapfig2.

v0.903: Various updates and improvements.

lwarpmk
- Error if *pdftotext* not available. Ensures that POPPLER programs are installed.

core
- *ps2pdf*: Allow transparency due to recent changes in *ps2pdf*.

⚠ New images
- Due to changes in how automatically-generated SVG image file names are computed, after **lwarpmk html** use **lwarpmk cleanlimages** a single time, and then **lwarpmk limages** to generate the new images.
- Improved back refs.
- Fixed verbatim*.
- Various internal updates for recent LATEX release.

packages
- cuted: Updated to v2.0.
- flushend: Updated to v4.0.
- mathalpha: Updated for v1.14+.
- minted: Updated to v2.6.
- cases: Updated to v3.2.
- siunitx with MATHJAX: Improved \per, \numlist, \SIlist, comma decimal points.
- Added showlabels, wrapfig2.

v0.902: beamerarticle, footnotes, paragraph tags.

core
- Fixed footnotes inside descriptions, minipages, amsthm, \nameref.
- Improved various paragraph tags.

packages
- Improved parnotes, sympytex.
- Added beamerarticle.
- Updated luatexko, xetexko, tagpdf.

MATHJAX
- Added missing standard international text symbols for MATHJAX.

v0.901: Tabular columns, float caption CSS, MATHJAX packages.

core
- Added warpsvg to isolate SVG math, as opposed to warpMathJax.
- Improved float caption CSS for newer browsers.
- Improved emulation of \newcolumntype.
- Added \HTMLnewcolumntype. See section 7.6,
- >{\centering\arraybackslash}, etc. now sets HTML CSS text-align. Also detects \itshape, \bfseries, and \bfseries\itshape. See section 8.10.1.

MATHJAX
- Now uses MATHJAX 3.2 packages for centernot, colortbl, gensymb, mathtools, textcomp, upgreek.

packages
- dcolumn: Now works inside a lateximage.
- Added mwe.
- Added lltjp-tascmac, which fixed ascmac.

v0.900: Package updates.

core
packages

- Fix for detecting \usepackage{lwarp}.
- amsmath: Fixed alignat with MATHJAX.
- changes: Updated to v4.2.1.
- froufrou: Updated to v1.4.0.
- lipsum: Updated to v2.3.

v0.899: Minor updates.

core
packages

- *lwarpmk*: Warns if \usepackage{lwarp} is not detected.
- graphics: Added support for keepaspectratio.
- keyfloat: Fix: lw with h.
- multicol: Improved css.

v0.898: Minor updates.

- Fewer underfull \hbox warnings.
- wrapfig: Improved integration with keyfloat.

v0.897: siunitx rollback.

docs

- Added a table of file extensions to use with \includegraphics. See table 9.

core

- Added tests for additional incompatible packages.

packages

- siunitx: Supports rollback to v2. Does not yet support v3.
- fixme: Improved to work if the user modifies layouts.
- float: Improved integration with newfloat, keyfloat.
- Added centerlastline, decorule, fancypar, froufrou, pbalance.
- Verified works as-is with fnpct.

v0.896: Back references, accessibility.

⚠ New labels

- Due to changes in cross referencing, execute **lwarpmk clean** before recompiling.
- Increased sectioning nesting stack depth. Error if overflow stack.
- Fixed footnotes at the end of the document, or inside a description label.
- Added an error if using braces inside \usepackage options.

MATHJAX

- Fixed footnotes in bracket display math with MATHJAX.

theorems

- LATEX theorems, amsthm, ntheorem, theorem: Print theorem footnotes following theorems.

accessibility

- Added HTML <main> element to each page.
- Added ARIA math role to SVG math images, and note role to margin notes, footnotes, etc.

packages

- Improved citation backreferences for various packages.
- chemfig: Updated to v1.6a.
- bigdelim: Updated to v2.8.
- xetexko: Updated to v3.1.

- hyperxmp: Fix: Accept and discard additional keys.
- hyperref: Fix: Added *autorefname macros.
- biblatex: Fix: Back references.
- tocloft: Fix: \cftpagenumbersoff, \cftpagenumberson.
- threeparttablex: Fix: \TPTL@tnotex.
- amsthm: Fix: Footnotes inside environment optional argument.
- listings: Fixed labels. Accepts but ignores escapes w/o error.
- pdflscape: Fix: Added landscape environment.
- Added ccicons, classicthesis, orcidlink.
- Added enotez.
- Verified support for doi, doipubmed.

v0.895: Vector packages, greatly improved MathJax for siunitx.

core
- Fixed quotes in HTML tags while using old font packages with X∃LATEX and LuaLATEX.

MathJax
- Added \ifblank and \ifstrequal to MathJax emulation.

packages
- multirow: Allow \par per v2.7.
- acro: Updated to v3.5.
- fancyhdr: Updated to v4.0.
- changes: Updated to v4.0.1.
- epsfig, rotating: Now work inside lateximage.
- amscdx: Verified to work with SVG math. Warning added about use with MathJax.
- Added MathJax emulation for isomath, mattens, maybemath, skmath, tensor.
- Improved MathJax emulation for siunitx \ang, \num, \SI.
- Added epsf, impnattypo, isotope, lpic, luavlna, mdwmath, pinlabel, rlepsf, tikz-imagelabels, xevlna.
- Verified to work as-is: tensind.

v0.894: MathJax additions and improvements.

MathJax
- Improved warning message for enabling SVG graphics for select math expressions while using MathJax.
- Accept and ignore a star for \hspace.
- Ignores \arabic, \number, \noalign.

packages
- Added MathJax emulation for backnaur, colortbl, nicematrix.
- booktabs: MathJax emulation now absorbs and discards trim.
- menukeys: Updated to v1.6.1.

v0.893: Minor fixes, more packages.

MathJax
- Added MathJax emulation for \mathnormal.

packages
- Fixed pstricks pspicture*.
- Fixed tikz font macros.
- braket: Now uses the MathJax extension.

- Added esvect, fixmath, keystroke, mathastext, menukeys, picinpar, plimsoll, repltext, selectp, seqsplit, simplebnf, statistics, swfigure.
- Added MATHJAX emulation for mathspec.
- Verified to work as-is for apxproof, syntaxdi, venndiagram.

v0.892: minted, fvextra, MATHJAX \left/\right.

MATHJAX
- fourier, libertinust1math, newpxmath, newtxmath, newtxsf, unicode-math: Added MATHJAX\left/\right support for additional delimiters.

packages
- textpos: Updated to v1.10.
- xcolor: Fixed optional args for \fcolorbox and related.
- Added fvextra, minted.

v0.891: MATHJAX additions and improvements.

core
- Now displays inline \verb text as \texttt.
- Fixed alltt and verbatims with LATEX lists.
- Now generates an error if nested each of warpHTML, warpprint, warpMathJax inside itself.

MATHJAX
- Added MATHJAX *textmacros* extension, allowing formatting inside \text.

packages
- biblatex, hyperref: Added back page references.
- fancyvrb: Fixed BVerbatim with a label.
- listings: Fixed MATHJAX with captions, improved HTML sanitation.
- babel-french: Fixed \texorpdfstring conflict.
- Now honors Greek package options for mathdesign, mathpazo, mathptmx, newpxmath, newtxmath.
- Improved MATHJAX for colonequals, mathdesign, mathdots, mathfixs, mathtools, multiobjective, nicefrac, shuffle, units.
- unicode-math: Added Greek macros, as well as macros for the first several categories listed in **texdoc unimath-symbols**. Improved symbol shape macros with Greek. Improved documentation.
- Added bussproofs, cmbright, fourier, kpfonts, kpfonts-otf, libertinust1math, scalerel, txgreeks.

v0.89: Additional MATHJAX support.

core
- Adapted to upcoming LATEX kernel changes.
- Allows load of amsmath before lwarp.

lwarpmk
- Also removes *.bbl when cleaning aux files.

MATHJAX
- MATHJAX: Neutralized \protect, \mathcode and related, ligatures. Fixed nested environments.

packages
- caption: Updated for v3.5, fix for label sep.
- thmtools: Updated for v0.72. Fixed swapnumber, margin.
- Improved MATHJAX for centernot, mathtools, mismath, Slunits, siunitx, statmath.
- Added MATHJAX emulation for accents, hepunits, hhtensor, mathalpha, mathdesign, mathpazo, mathptmx, mleftright, newpxmath, newtxmath, newtxsf, pxfonts, shuffle, txfonts, upgreek, ushort.
- Verified to work as-is: authoraftertitle.

v0.88: Indexing, boxing, theorems.

- Now has programmed support for more than 500 packages and classes, of which more than 60 also support MATHJAX.

core
- Fixed: \ref*, and also added MATHJAX emulation.
- If starting a new paragraph, \hrulefill creates a <div> with a thin horizontal line across the page. Use instead of \hrule.
- Fixed: Use \chaptername where appropriate.
- Fixed: Inline links causing extraneous paragraphs.

lwarpmk
- Added **lwarpmk -v** to print the version number.

indexing
- Added the IndexRef option to control the display of index entries. See section 7.5.
- Added \IndexPageSeparator and \IndexRangeSeparator for custom index styles.
- Added support for gindex, *xindex*.
- Verified to work as-is with varindex.

packages
- cleveref, varioref: Fix for starred macros.
- varioref: Removed page-related text from HTML output.
- xfakebold: Updated to v0.08, using pdfrender.
- caption, scrextend: Fixed \caption*.
- Added fbox, shadethm, tcolorbox, termcal, thmbox, thmtools.

v0.87: MATHJAX, bibliography packages.

core
- Added boolean FixSmallCaps for fonts which render small caps as all caps.
- Fixed \bibliography to use the HTML version's .bbl file. Previously the HTML bibliography relied on the print version's .bbl, thus would fail if the print document had not yet been created.

MATHJAX
⚠ Removed
\DeclareIfstar
packages
- Added \ifstar and \ifnextchar to MATHJAX, and removed \DeclareIfstar. See section 8.7.7.
- physics: Now supports the MATHJAX v3 extension.
- mathtools: Improved \underbrakcet, \overbracket for MATHJAX.
- nccmath: Improved \underrel for MATHJAX.
- mhchem: Now supports the MATHJAX v3 extension for \ce inside math.
- cancel: Now supports the MATHJAX v3 extension.
- embrac: Neutralized kerning for improved HTML conversion.
- Added citeref, drftcite, jurabib, multibib, splitbib.
- Verified to work as-is with bibtopic, collref, mciteplus.

v0.86: MATHJAX major updates.

core
- Fixed: Filename if named files with *, parens, period in section name.
- Fixed: Labels in eqnarray, lateximage.

MATHJAX
- Updated to MATHJAX v3. New repository.
- Fixed forward references for MATHJAX.

- Improved MATHJAX equation number formatting, now compatible with amsmath \numberwithin for chapters, sections, subsections, as well as amsmath subequations. See section 8.7.7.

- Added \DeclareIfstar to define starred TEX macros in MATHJAX. See section 8.7.7.

- Generates an error if \MathJaxFilename file does not exist.

packages
- mathtools, nccmath, physics: Added starred macros for MATHJAX.

- nccmath: Fixed \nr, \displaybreak for MATHJAX.

- xcolor: Fixed \textcolor with babel-french.

v0.85: fontspec

packages
- fontspec: Fixed core font change macros for world languages.

⚠ acro formats
- acro: Due to v3 changes, when defining acronym formats, use \textbf instead of \bfseries, etc.

- Fixed idxlayout, mathtools, titlesec, url.

v0.84: Previous/next page links, numerous fixes.

- Added documentation of BlockClass and \InlineClass for css <div>s and <span>s. See section 7.8.

docs
- Added \LinkPrevious, \LinkNext page links. See section 7.6.

⚠ home page footer changed
- Added \FirstPageBottom. Home page no longer shares \PageBottom. See section 7.6.

- Improved coexistence with comment, support for nested environments.

core
- No longer requires but still supports the caption package.

- Improved filenames and HTML titles when using special characters.

⚠
- Change: Append -0 to section named Index previously _index to distinguish from index.html

- Fixed style tags for \multicolumn, \multirow.

- Fixed spacing in tabbing.

- Fixed lateximage for: quote, quotation, verse, center, flushleft, flushright, <par> tags, packages verbatim, alltt, epigraph.

- Fixed textcomp due to integration into LATEX kernel.

- Fixed \itshape, etc. Adapted to LATEX fontaxes integration.

- Fixed \@fnsymbol.

- Warns about section names with dollar-delimited math.

- Warns about a <span> containing a float, caption, section, mdframed, or other <div> object.

- Only warn about X∃TEX logo and graphics if actually used \Xe.

lwarpmk
- **lwarpmk clean** also removes comment_*.cut.

packages
- scrextend, scrartcl, scrbook: Added \titlehead, \subject, \subtitle, \publishers.

- titling: Fixed \printthanks.

- memoir, abstract: Fixed for updated memoir.

- memoir: Fixed \newcomment, pagenotes, crossreferences. Fixed setting a recursive name.

- Fixed or improved: amsthm, backref, biblatex, fixme, nfssext-cfr, ntheorem, parcolumns, realscripts, rotfloat, titling.
- Added boxedminipage, renamed from boxedminipage2e per author.
- Verified to work as-is with mcite.

v0.83: memoir fixes.

packages
- memoir: Various fixes and updates.
- physunits: Updated to v1.0.4.

v0.82: MATHJAX notes, xpinyin improvements, various updates.

MATHJAX
- Improved footnotes with MATHJAX.
- Added MATHJAX emulation for endnotes, marginnote, nccfoots, pagenote, parnotes, sidenotes.

packages
- xpinyin: Added pinyin with modern HTML.
- luatexko: Added \dotemph, \ruby, \uline, etc.
- soul: Fixed \<.
- chemfig: Updated to v1.5.
- draftwatermark: Updated to v2.0.
- ulem: Fixed: \dashuline.
- amsmath: Fixed: \intertext with MATHJAX.
- endnotes: Fixed: Marks in print mode.
- tocvsec2, tableof: Verified to work as-is.
- Added etoc (nullified).

v0.81: MATHJAX speedup and additional emulations.

core
- Improved warning regarding SVG math sizing / baselines and graphics / graphicx. See section 8.7.

MATHJAX
- Improved MATHJAX emulation processing speed.
- Added MATHJAX emulation for accsupp, axessibiltiy, colonequals, decimal, dotlessi, econometrics, engtlc, multiobjective, physunits, Slunits, stackrel, statmath.

packages
- axessibility: Updated to 2020/01/08 version.
- gridset: Updated to v0.3.
- Slunits: Fixed for math mode.
- Added DotArrow, nolbreaks, luamplib, returntogrid, statex2, tagpdf.
- Verified to work as-is with icomma, mathpunctspace, textualicomma.

v0.80: MATHJAX, biblatex.

MATHJAX
- Added docs and warning/info messages re: avoiding slow MATHJAX compilation. See section 8.7.7, Customizing MATHJAX.
- Added MATHJAX emulation for accessibility, autobreak, centernot, extarrows, fouridx, gensymb, leftidx, mathcomp, mathdots, mathfixs, mismath, nccmath, noitcrul, pdfcomment, relsize, rmathbr, subsupscripts, xfrac.
- Improved MATHJAX emulation for unicode-math.

packages
- biblatex, url: Now create hyperlinks.

- amsmath: Fix to center starred environments.
- xcolor, graphics: Made more macros robust.
- colortbl: Fix: Rule color in a lateximage.
- chemmacros: Updated to v5.10.
- Added fewerfloatpages, ghsystem, hhline, mismath, nccmath.

v0.79: MATHJAX, nested tabular.

MATHJAX
- Added or improved MATHJAX emulation for amsmath, ar, arydshln, bm, bigdelim, bigstrut, booktabs, braket, mathtools, multirow, physics, siunitx, slashed, unicode-math, xfakebold.
- Warn if using certain packages not supported by MATHJAX.

core
- tabular: Now may be nested.
- minipage, \parbox, fminipage, \makebox, \framebox: Fix: Adjust for virtual page size.
- Uses new iftex.

packages
- graphicx: Fix: Negative angles.
- caption: Fix: \captionlistentry with longtable.
- multirow: Fix: Centered vertical alignment.
- siunitx: Fix: \square, \cubed.
- booktabs: Fix: memoir with lateximage.
- babel and polyglossia: Added troubleshooting warnings.
- fontawesome, fontawesome5: Supports text color and size.
- transparent: Fix: lateximages.
- epigraph: Updated to v1.5e.
- xurl: Updated to v0.08.
- subcaption: Fixed with memoir.
- floatrow: Fix: \linewidth. No longer require float, graphics.
- floatflt, wrapfig, niceframe: Fix: Adjust for virtual page size.
- Added widetable, witharrows, steinmetz.
- Added awesomebox, catoptions.
- Added svg, supports svg-extract.
- Added parcolumns, pdfcolparcolumns,
- Added parallel, pdfcolparallel.
- Added pdfcol, pdfcolfoot, pdfcolmk.

v0.78: Fixes for support files, alt tags, hyperlinks, and the 2019/10 LaTeX release.

docs
- Docs: Improved documentation regarding package options. See section 8.1.
- Fix to overwrite existing support files using new filecontents environment.

packages
- breqn: Previously broken by the 2019/10 LaTeX update, but now working again.
- graphics: Fix for \includegraphics alt tags.
- babel-french: Fix for hyperlinks.
- media9, movie15, multimedia: Fix for the 2019/10 LaTeX update.
- accessibility: Added.

v0.77: Updates to fix recently-broken packages.

- booktabs: Updated to v1.6180339.
- chemformula: Updated to v4.15.

v0.76: MATHJAX, updates for LATEX 2019/10 release.

docs
- Docs: Expanded documentation regarding the use of multiple projects in the same directory. See section 5.18.

MATHJAX
- MATHJAX: Updated to v2.7.6.

packages
- xr: Updated to v5.05.
- xr-hyper: Updated to v6.1.
- Verified works as-is with xcite.
- acro: Updated to v2.10.

⚠ broken
- Currently broken in print mode by the 2019/10 LATEX update, and waiting for fixes: breqn, grffile, multimedia, movie15.

v0.75: keyfloat, wrapfig

- \minipage: Fix for \linewidth.

packages
- keyfloat: Improved color control.
- wrapfig: Fix for \linewidth.

v0.74: Docs, svg math, *lwarpmk*, html alt and title text, lyluatex

docs
- Added to the tutorial the section What next?. See section 5.20.
- Added documentation about localization options. See section 7.1.
- Added documentation about accessibility options. See section 7.2.

HTML alt text
⚠ changed names
- Renamed and updated HTML alt text macros:

Old	New
(hard coded as "image")	\ImageAltText
\mathimagename	\MathImageAltText
\packagediagramname	\PackageDiagramAltText

- Added \ImageAltText for the default HTML alt text for an image. See section 7.6.
- Added \ThisAltText, which may be used to assign a one-time HTML alt tag to the very next image generated by lwarp, such as a lateximage, picture, tikzpicture, an image generated by various chemistry or engineering packages, or an svg math image. This macro also adds a title tag to a reference or hyperlink. See section 7.6.

svg math
- Adjusted \LateximageFontScale default from .75 to 1.
- Fix: Font control for svg math.

misc
- Fix: Ignores negative \hspace.
- Warning if SideTOCDepth < FileDepth.

lwarpmk
- *lwarpmk*: **lwarpmk clean** removes additional files.
- *lwarpmk*: **lwarpmk epstopdf** and **lwarpmk pdftosvg** now honor directories.

packages
- lyluatex: Split images by system or per fullpage, improved margins and scaling.
- Tested to work as-is with mathspec, unicode-math.

v0.73: \include, memoir, koma-script, caption, xy, datatool, music scores.

- Fix for \include.
- Warning for a tabular inside a <span>.
- \color: Added HTML support for rules and frames, but not inline text. Use \textcolor if possible.
- Improved many HTML tags, reducing *tidy* warnings. See Change History.

packages
- memoir: Fixes for \frontmatter* and \mainmatter*. Added \book.
- koma-script: Fix for starred captions in the TOC.
- caption: Fix for starred captions.
- datatool: Added pie, bar, and plot charts.
- threeparttable: Added measuredfigure.
- intopdf: Updated to v0.2.1.
- tocdata: Updated to v2.03.
- quotchap: Updated to v1.2.
- versonotes: Updated to v0.4.
- backnaur: Now uses SVG images. Updated to v3.1.
- xy: Fix for \xybox, improved xy, also now compatible with qcircuit.
- fancyvrb: Fix for label HTML tags.
- Added stackengine.

music
- Added lyluatex. (Music scores.)
- musicography: Updated to 2019/05/28. Added support for lateximages.

v0.72: Font control, \multicolumn, xr and xr-hyper.

⚠ images
- Due to internal changes, images for inline SVG math and lateximages will have new hash values, and will have to be regenerated using

  Enter ⇒ **lwarpmk cleanlimages**

  and

  Enter ⇒ **lwarpmk limages**

- Docs: Color-codes package names in the table of supported packages and features, table 2, according to each package's level of support by lwarp.
- \multicolumn: Fix for paragraph columns.

packages
- xr, xr-hyper: Fixes for references, \externaldocument.
- soulutf8: Fix: Loads soul for emulation.
- boxedminipage2e: Added support for lateximages.
- zhlineskip: Updated to v1.0e.
- Added fontaxes, slantsc, tabfigures.
- Added nfssext-cfr, thus supporting cfr-lm and several other font packages.
- Added backnaur, hypbmsec, minibox, pdfcrypt, shapepar.

v0.71: Error handling, multimedia, tabular.

- tabular: Added support for '*' columns. Fix for paragraph tags.
- quotation: Fix for HTML tag.
- Docs: Added a section about error conditions tested by lwarp. See section 13.1.
- *lwarpmk*: If file lwarpmk.conf is an older version, or the incorrect operating system, displays the print command to use to recompile.

packages
- chemfig: Updated for v1.4.
- endfloat: Updated for v2.7.
- textpos: Updated for v1.9.1.

multimedia
- Added media9, movie15, multimedia.

v0.70: Error handling, MATHJAX, mathtools.

- Error handling for "Label(s) changed." Refuses to **lwarpmk limages** until recompile first.
- Fix: If Computer Modern font is used, ensures cm-super or lmodern is used.
- Fixes for \makebox.
- Fixes for \parbox inside a <span>.
- MATHJAX: Updated to v2.7.5. Loads the autoload-all.js extension. Added \MathJaxFilename to select custom scripts.

packages
- textcomp, xunicode: Fix for \textinterrobang.
- mhchem: Works with MATHJAX. See section 417.
- changes: Updated to v3.1.2.
- Added autonum, changelayout, inputtrc, mathtools, metalogox.

v0.69: Error handling, many fixes, improved keyfloat / tocdata.

- Fix for HTML corruption of lateximage displays.
- \makebox, \framebox: Fix for (⟨*width,height*⟩) arguments.
- fminipage: Honors \minipagefullwidth.

packages
- array, longtable: Fix for \tabularnewline.
- tabularx, tabulary: Fix to require the array package.
- supertabular, xtab: Fix to clear caption after use.
- graphics: Added a warning if used the \includegraphics scale option.
- multirow: Added an error if didn't use \mrowcell or \mcolrowcell when using \multirow or \multicolumnrow.
- keyfloat: Updated for v2.00, additional improvements.
- Added ctable, eqlist, eqparbox, ftcap, listliketab, minitoc, tocdata, topcapt.

v0.68: Error handling, tabulars, footnotes.

lwarpmk
- *lwarpmk*: Improved error handling for image generation if compile was incomplete.
- tabular: Fix for \warpprintonly.

packages
- longtable: Improved flexibility for \endhead, etc. Improved error reporting if \endhead, etc. incorrect for lwarp.

- threeparttable: Fix for caption type.
- hyperref: Fix for options with braces.
- morefloats: Fix to be loaded early for print output.
- listings: Updated for v1.7.
- Added bigfoot, fnpara, footnotebackref, manyfoot, tablefootnote, threeparttablex.
- Added layouts, niceframe, perpage, showtags.
- Prevented alg, algorithmic, pdfcprot, fncylab.

v0.67: Filename generation, symbol fonts.

docs
- Documentation fix for <project>-images, <project>-images.txt.
- Added discussion regarding section names. See section 8.4.

filenames
- Added \FilenameNullify and \FilenameSimplify for filename generation. See section 8.4.
- Core, textcomp, xunicode: Nullified additional symbols during filename generation.

packages
- color: Fix for version number warnings.
- Added academicons, bbding, dingbat, eurosym, fontawesome, fontawesome5, marvosym, pifont, typicons.
- Added changes, easyReview, fitbox, foreign, gloss, karnaugh-map, multicap, nomencl, notes, struktex, umoline, xfakebold.
- Tested to work as-is with askmaps, curves, euro, karnaughmap, tikz-karnaugh.

v0.66: xr, multiple projects, image names/directory, HTML formatting

⚠ Reset the configuration
- Due to changes in *lwarpmk*, recompile any existing project a single time using **pdflatex filename.tex** or similar, after which *lwarpmk* may then be used with the new configuration files.

lateximage
- Adds options ImagesDirectory and ImagesName to assign directory and name prefixes for lateximage images. The new defaults include the jobname, allowing the image directories for multiple projects to coexist.

⚠ existing projects
- To reuse existing lateximage directories, add lwarp options

```
\usepackage[
  ImagesDirectory={lateximages},
  ImagesName={lateximage-}
]{lwarp}
```

  If not reused, the existing lateximages directory and lateximages.txt file may be removed.

filenames
- Added \FilenameLimit to control the maximum length of the filenames generated by lwarp.

⚠ Possible filename changes
- Improved filename generation when special characters or macros are used in section names.

WINDOWS
- Fix for **lwarpmk cleanlimages** with WINDOWS.

floats
- Fixes for floats in the home page.

lists, table notes
- Improved CSS for definition lists, table notes.

tabular
- tabular: Fixes for \par in column specifier, minipage inside tabular.

indexing
- Indexing: Fix for a long line of multiple entries.

minipage
- \minipagefullwidth: Fix for global changes.

- Added \UseMinipageWidths and \IgnoreMinipageWidths. See section 8.3.3.

colors
- Improved \fbox, \fboxBlock, \fminipage to use current text color.

HTML
- Improved HTML output formatting.

docs
- Added discussion regarding invalid HTML. See section 8.1.1.

- Added discussion regarding math in section names, \imagegraphics scale option. See section 6.

- Added discussion regarding international languages in section names. See section 8.14.

packages
- caption: Fix for options clash.

- xr, xr-hyper: Now compatible.

- subcaption: Improved horizontal spacing.

- multicol: Fix for minipage inside multicols.

- multicolrule: Updated for v1.2.

- tocbasic: Minor update.

- acronym: Fix for acronym in float caption.

- kotexutf: Patch with *pdflatex* and new lwarp labels.

- extramarks, fancyhdr: Updated for v3.10.

- memoir: Added docs regarding version numbers. See section 8.13.

- zref: No longer required.

- Added ar, ed, indentfirst, nameauth, truncate.

- Verified to work as-is with changelog.

- Prevented colortab, epsf, hyper, picinpar, picins, sistyle, ucs.

v0.65: CSS layout, alt tags, Japanese.

page layout
- Moved the sideTOC to the left side, allowing improved CSS for margin notes.

- Improved page layout CSS.

image alt tags
- graphicx \includegraphics: Added the alt key to assign an alt tag to an image. Default is "image", assigned to pass validation.

duplicate HTML files
- Detects and causes an error if duplicate HTML file names are generated, caused by identical or similar sectioning names.

fixes
- Fix for tabular*.

- Fix for tabular border colors.

- Fixes \quad, \enskip, and figure captions to pass validation.

Japanese
- Added ltj* classes, bounddvi, gentombow, lltjext, plarydshln, plext, plextarydshln, plextcolortbl, pxatbegshi, pxeveryshi, pxftnright, pxjahyper, tascmac.

- Verified to work with plarray, plautopatch, plextarray, plextdelarray, pxgentombow, plsiunitx, pxpdfpages, pxpgfrcs, pxpgfmark.

packages
- Added support for fontspec \textsi and \sishape.

- Added multicol's \docolaction.

- Added embrac, footnoterange, multicolrule, versonotes.

v0.64: Koma-Script, Japanese, Chinese.

Japanese
- Added utarticle and related classes.
- Improved ujarticle and related classes.

Chinese
- Fix for biblatex with CTEX and other classes.

Koma-Script
- Fixes for scrlayer, scrlayer-scrpage.

packages
- addlines: Updated to v0.3.
- Added bsheaders, gmeometric, marginal, rmpage, scrpage2.

v0.63: mdframed, Chinese, Japanese, Korean

localization
- Added \linkhomename: A user-definable name for the **Home** link.
- Documented \sidetocname: A user-definable name for the sideTOC.

fixes
- Fix: \LinkHome for print output.

optimizations
- Moved package load checks to the lwarp core to reduce the number of lwarp-* files.

packages
- mdframed: Fix with amsthm, improved titles and font control. Improved rule widths.

Chinese
- Fixes for xeCJK.
- Added xpinyin, zhlineskip.
- Verified to work with cjkpunct, upzhkinsoku, zhspacing.

Japanese
- Verified to work with zxjatype, luatexja, luatexja-fontspec.
- Added bxjsarticle and related classes.
- Added ltjsarticle and related classes.
- Added pLATEX, upLATEX, ujarticle and related classes.
- Prevented utarticle and related classes.
- Prevented bxcjkatype.

Korean
- Verified to work with kotex, xetexko, luatexko.

v0.62: MiKTEX docs, HTML title, CTEX, xeCJK, bitpattern.

docs
- Docs: Setting a UTF-8 locale. See section 9.9.

MiKTEX
- MiKTEX: Docs for *MiKTeX Console* and miktex-poppler-bin.

HTML <title>
- HTML subpage titles: Added \HTMLTitleBeforeSection and \HTMLTitleAfterSection to select whether the HTML <title> displays the website name before or after the section name. See section 7.6.

fixes
- Fix for package options handling.
- Fixes for horizontal white space between fminipage, fcolorminipage, colorboxBlock, fcolorboxBlock.
- Logos: Fix for X∃TEX logo, improved CSS, made robust, improved search-engine optimization.
- \\[$1]: Additional HTML <br> if $1 > 0$ pt.
- Fixes for \includgraphics filename, and with FormatWP.
- Fix: CSS for \textup.
- Fix: Added \slshape.

Chinese
- Added ctex package and related classes, xeCJK.
- Prevented CJK, CJKutf8 unless xeCJK, ctex are used.

packages
- chemfig: Docs for new macro \polymerdelim.
- asymptote: Docs for compilation.
- chngpage: Fix to load lwarp-changepage.
- algorithm2e: Fix with non-book classes.
- register: Updated to v1.8.
- nicefrac: Improved font control and css, honors nice and ugly.
- units: Improved font control and css, honors tight and loose.
- xfrac: Improved css.
- textcomp and xunicode: Fix conflicts with \textcircled.
- ulem: Improved compatibility with CJKulem, lateximage.
- MATHJAX and siunitx: Removed inoperable extension.
- Added bitpattern, pdfcomment, pdfmarginpar, tram, unitsdef, xechangebar.
- Added musicography, octave, semantic-markup.
- Added 2in1, flippdf, notespages, rviewport, twoup.

v0.61: Custom compilation, EPS-related packages, documentation, indexes.

docs
- Split index into multiple indexes.
- Improved documentation regarding font selection. See section 7.4.
- Added documentation regarding debugging options. See section 37.
- Added documentation regarding HTML entities inside program listings. See section 8.2.1.

custom compiling
- Added options to specify the shell commands to execute for **lwarpmk print** and **lwarpmk html**, allowing the use of lwarp with perltex, pythontex, etc. If not specified, these are set automatically depending on the LATEX engine, --shell-escape, and lwarp options. See section 9.

⚠ changed names
- Changed macro names to match \displaymathother, \displaymathnormal:

Old	New
\StartDynamicMath	\inlinemathother
\StopDynamicMath	\inlinemathnormal

fixes
- Fix: Paragraph tags in a tabular.
- Fix: supertabular and xtab captions.
- Fix: DVI LATEX \includegraphics EPS images.
- Fix: newfloat lists.
- Fix: css footnotes text align, minipage tabular and footnote margins.

packages
- Added epsfig, psfrag, psfragx, pstool.
- Added copyrightbox, pdfprivacy, thinsp, threadcol, uspace.
- Added chkfloat, cmdtrack, dprogress, lua-visual-debug, refcheck, srcltx, srctex, vpe, xbmks.

v0.60: Fixes for longtable, listings.

fixes
- longtable, etc.: Fixes for slowdown and memory management for very long tables.
- listings: Fix for HTML entities, and also when used inside a list.
- diagbox: Fix for incorrect HTML par tags.

packages
- Added 2up, booklet.
- Added bophook, draftfigure, fullminipage, grid-system, layaureo.
- Added leading, widows-and-orphans.
- Added fancytabs, thumb, thumbs.

v0.59: DVI *latex*, MATHJAX, asymptote, pdftricks and pstricks, epstopdf, brqen.

⚠ Reset the configuration
- Due to changes in *lwarpmk*, recompile any existing project a single time using **pdflatex filename.tex** or similar, after which *lwarpmk* may then be used with the new configuration files.

lwarpmk
- Added an error if lwarpmk.conf's format has changed and the document must be recompiled.
- Added a warning if the lwarpmk.conf configuration file appears to be for the wrong operating system, in case files are transferred between systems.
- Added

        lwarpmk epstopdf <list-of-EPS-files>

  to quickly convert a document's EPS images to PDF or SVG. See section 8.8.

DVI *latex*
- Added support for DVI *latex*. See section 7.5.

latexmk
- Fix for --shell-escape with *latexmk*.

math
- Updated MATHJAX script to v2.7.4.
- Fix: MATHJAX chapter number removed from non-numeric tagged equations.
- Added MATHJAX support for nicefrac, units.
- Fix for \[and \] with \displaymathnormal.

images
- Fix for \includegraphics filename expansion.
- \includegraphics now works with .pdf and .eps filename extensions.

packages
- Moved amsmath out of the lwarp core.
- Fix for chemformula \NMR.
- Added asymptote, pdftricks, pstricks, pst-eps.
- Added breqn, SIunits.
- Added bxpapersize, canoniclayout, draftcopy, fnbreak, nccfancyhdr.
- Added accsupp, axessibility.
- Added xunicode.
- Improved and now supports epstopdf.
- Tested to work as-is: eepic, sepfootnotes.

docs
- Added information about setting up a development version of lwarp.

v0.58: Extensive improvements in indexing, glossaries. Adds PDF-inclusion packages.

⚠ Reset the configuration
- Due to changes in *lwarpmk*, recompile any existing project a single time using **pdflatex filename.tex** or similar, after which *lwarpmk* may then be used with the new configuration files.

lwarpmk
- *lwarpmk*: Added the -p option to specify the project name.

glossaries
- *lwarpmk*: Now uses *makeglossaries* for glossary generation, allowing the processing of multiple glossaries at once.

- Added lwarp option GlossaryCmd to specify the shell command used by **lwarpmk printglossary** and **lwarpmk htmlglossary**. Defaults to **makeglossaries**.

index and glossary
- Docs: Extra indexing options. See section 8.6.15.

- Added support for *makeindex*. (Previously supported only *xindy*.) Also added indexing packages listed below.

- Added lwarp options PrintIndexCmd, HTMLIndexCmd, and LatexmkIndexCmd to specify shell commands used by **lwarpmk printindex**, **lwarpmk htmlindex**, and *latexmk*. May be preset with the makeindex or xindy lwarp options. See section 7.5.

- Added lwarp options makeindex and xindy to set PrintIndexCmd, HTMLIndexCmd, and LatexmkIndexCmd to sensible values for a typical single index. See section 7.5.

- Added lwarp option makeindexStyle to tell *lwarpmk* to use a custom style instead of lwarp.ist. See section 8.6.21.

- Fix for index entries with \see, \seealso, \emph, \textbf, etc.

misc. fixes
- Replaced each \csuse with \@nameuse for improved error detection.

- Additional internal print/HTML macro selection improvements.

- Fix: \printindex finishes pending \index writes first.

- Fixes for memoir: makeidx, ccaption, multiple indexes, \specialindex.

packages
- Fixes for komascript: Indexing improvements.

- Added imakeidx, index, repeatindex, splitidx.

- Added attachfile, attachfile2, intopdf, pdfpages, pdfx.

- Added cases.

- Tested to work as-is: notes2bib, hvindex.

v0.57: algorithm2e, float styles, tabular packages, internal improvements.

- Added support for MathJax equations with \footnote, \footnotemark.

MathJax

math macros
- Added \StartDefiningMath and \StopDefiningMath for use when defining macros in the preamble which contain $. See section 8.7.9.

dynamic math
- Added \inlinemathother and \inlinemathnormal to delimit math expressions which depend on a variable condition such as a counter. Such expressions will not be hashed for reuse, and will be converted to svg math images even when MathJax is enabled. See section 8.7.10.

⚠ *new name*
- Renamed \EndDefiningTabulars to \StopDefiningTabulars.

lateximage alt tags
- Improved localization for lateximage HTML alt tags. For svg math images, the alt tag under some conditions will be set to \MathImageAltText, which defaults to math image. For packages, the alt tag is set using the package name followed by \PackageDiagramAltText, which defaults to diagram. Ex:

        (-xy- diagram)

  See section 7.6.

- Fix: Improved print/HTML macro selection.

misc. fixes
- Fix: \href text catcodes.

- Fix: \subref text.

- Fixes: Colored \rule and \boxframe.

packages
- **float, rotfloat:** Adds support for float styles ruled and boxed.

- **float:** Fix: Do not create \l@<type> until \listof is used.

- **marginnote:** Fix: Long optional argument.

- **ellipsis:** Adds \midwordellipsis.

- **breakurl:** Fix for text catcodes.

- Added **algorithm2e, register, ltablex, xltabular, xellipsis, trimclip, errata, vowel, xpiano.**

- Prevents **glossary.**

- Tested to work as-is with **gauss, phonrule, piano, SIunits, tikzcodeblocks.**

v0.56: Shell escape, tabular packages.

lwarpmk
- Added

    lwarpmk pdftosvg <list-of-PDF-files>

    to quickly convert a document's PDF images to SVG, for use with HTML. See section 8.8.

- Added support for --shell-escape. See section 7.3.

tabular
- Added support for **array** w and W columns.

- Fix: \multicolumn parameter handling.

- Added support for double \hlines, \midrules, and vertical rules.

- Added support for **arydshln** dashed lines with HTML tabular, but reverts to plain rules for lateximage and SVG math array.

misc. fixes
- Fix: \thinspace.

- Fix: **paralist** compact environments.

packages
- Added **parnotes, quoting, lua-check-hyphen, tocenter, underscore.**

- Added **bibunits.**

- Tested to work as-is with **babelbib, bodegraph, fast-diagram, nicematrix, structmech.**

v0.55: Various fixes.

misc fixes
- Fix: Extraneous space in file links, which also prevented *Calibre* EPUB conversions.

- Fix: Float optional argument regression.

- Fix: \ForceHTMLTOC with \phantomsection.

- Fix: Overfull boxes in lateximages.

- Fix: QED symbols in lateximage.

packages
- **koma-script:** Fix: Figure with \centering, etc.

- Added **clrdblpg.**

v0.54: Float \centering, improved image checks.

⚠ Reset the configuration
- Due to changes in *lwarpmk*, recompile any existing project a single time using **pdflatex filename.tex** or similar, after which *lwarpmk* may then be used with the new configuration files.

lwarpmk
- lwarpmk limages checks for the presence of the HTML version of the document and valid image references before attempting to create the lateximages.

- *lwarpmk*: Improved error message if configuration file does not exist.

BIBTEX
- Added documentation for avoiding error with BibTeX and \etalchar. See section 8.6.9.

polyglossia
- Added documentation regarding polyglossia. See section 8.15.4.

macros in section names
- Added documentation regarding the use of macros in section names. See section 8.1.

document encoding
⚠ New and revised encoding options
- Renamed and added package options:

Old Package Option	New Package Option
xdyFilename	xindyStyle
IndexLanguage	xindyLanguage
–	xindyCodepage
–	pdftotextEnc

Use these options along with inputenc or inputenx to process documents in an encoding other than UTF-8. See section 7.4.

floats with \centering, etc.
- Floats now honor \centering, \raggedright, \raggedleft, and their ragged2e equivalents, when placed directly after:

  \begin{floattype}
  \centering

misc. fixes
- tikz: \pgfpicture, fit, align, font.
- ragged2e: \centering etc.
- hyperref: \hypertarget was creating duplicate of \label.
- hyperref: Active chars inside \hyperref, \hyperlink.
- hyperref: \ref inside \hyperlink caused a nested HTML link.
- glossaries: Fix when not using babel or polyglossia.
- textcomp: \textperthousand.
- LATEX core verse environment: line spacing.

⚠
- Removed \citetitle, adjusted \attribution.

packages
- memoir: Minor update for v3.7g.
- Added inputenx, bibunits, chngpage, forest, magaz, gridset.
- Prevents loading ae, aecc, t1enc, and wasysym.

v0.53: Improved image checks.

lwarpmk
- *lwarpmk*: Added a warning about corrupted images due to the need to recompile the document one more time.
- *lwarpmk*: Added the **lwarpmk cleanlimages** command.
- Added documentation for **lwarpmk cleanlimages** and **lwarpmk pdftohtml**.

v0.52: Improved footnotes, svg math.

documentation
- Improved install instructions regarding lwarp_baseline_marker.png.
- Added documentation regarding footnotes in section headings, and footnotes with \VerbatimFootnotes from fancybox, fancyvrb. See section 8.5.4.
- Added documentation regarding font selection when using XƎLATEX or LuaLATEX with fontspec and traditional font packages. See section 7.4.

SVG math
- Fix: Limit the number of background tasks when generating lateximages.

- Added user-adjustable svg math font scaling. See section 86.3.

- Added warnings if lwarp_baseline_marker.png is not present, or if graphicx or graphics is not loaded.

- Improved \ensuremath hashing expansion.

- Fix: equation* with split.

- tabbing now works inside a lateximage. Use for math in tabbing.

MathJax
- Fix: MATHJAX script was not executing in some conditions.

- Added \CustomizeMathJax to add custom functions. See section 8.7.

footnotes
- Fix: Footnote numbering when using HTMLDebugComments.

- Fix: Footnote paragraph tags.

- Fix: FootnoteDepth defaults to \subsubsection.

misc. fixes
- Fix: \kill in a lateximage.

- Fix: \FileDepth, misc. others, when input encoding is not utf8.

- Fix: \texorpdfstring in a section name.

packages
- hyperref emulation: Fix for #, %, &, ~, _ characters in URLs.

- fancybox, fancyvrb: Initial support for \VerbatimFootnotes.

- nicefrac: Added with fix for \ensuremath.

- graphicx: Fix for option defaults. Added v1.1a/b options.

- endfloat: Updated for v2.6.

- url: Fixes for active characters.

2 Introduction

The lwarp project aims to allow a rich LaTeX document to be converted to a reasonable HTML5 interpretation, with only minor intervention on the user's part. No attempt has been made to force LaTeX to provide for every HTML-related possibility, and HTML cannot exactly render every possible LaTeX concept. Where compromise is necessary, it is desirable to allow the print output to remain typographically rich, and compromise only in the HTML conversion.

Several "modern" features of HTML5, CSS3, and SVG are employed to allow a fairly feature-rich document without relying on the use of JAVASCRIPT. Limited testing on older browsers shows that these new features degrade gracefully.

––––––––––––––––––

lwarp is a native LaTeX package, and operates by either patching or emulating various functions. Source-level compatibility is a major goal, but occasional user intervention is required in certain cases.

As a package running directly in LaTeX, lwarp has some advantages over other methods of HTML conversion. TeX itself is still used, allowing a wider range of TeX trickery to be understood. Lua expressions are still available with LuaTeX. Entire categories of LaTeX packages work as-is when used with lwarp: definitions, file handling, utilities, internal data structures and calculations, specialized math-mode typesetting for various fields of science and engineering, and anything generating plain-text output. Blocks of PDF output may be automatically converted to SVG images while using the same font and spacing as the original print document, directly supporting TikZ and `picture`. Numerous packages are easily adapted for HTML versions, either by loading and patching the originals, or by creating nullified or emulated replacements, and all without resorting to external programming. As a result, several hundred packages have already been adapted (table 2), and an uncounted number more work as-is.

Packages have been selected according to several criteria: perceived importance, popularity lists, recent CTAN updates, CTAN topics, mention in other packages, support by other HTML conversion methods, and from sample documents taken from public archives. These include some "obsolete" packages as well.[1]

––––––––––––––––––

Assistance is also provided for modifying the HTML output to suit the creation of EPUB documents, and for modifying the HTML output to ease import into a word processor.

––––––––––––––––––

pdflatex, *xelatex*, or *lualatex* may be used, allowing lwarp to process the usual image formats. While generating HTML output, SVG files are used in place of PDF. Other formats such as PNG and JPG are used as-is.

---

[1]An amazing number of decades-old packages are still in use today.

SVG images may be used for math, and are also used for picture, Ti*k*Z, and similar environments. The SVG format has better browser and e-book support than MathML (as of this writing), while still allowing for high-quality display and printing of images (again, subject to potentially bug-ridden[2] browser support).

Furthermore, SVG images allow math to be presented with the same precise formatting as in the print version. Math is accompanied by <alt> tags holding the LaTeX source for the expression, allowing it to be copy/pasted into other documents.[3] Custom LaTeX macros may be used as-is in math expressions, since the math is evaluated entirely inside LaTeX. An MD5 hash is used to combine multiple instances of the same inline math expression into a single image file, which then needs to be converted to SVG only a single time.

The MATHJAX JavaScript display engine may be selected for math display instead of using SVG images. Subject to browser support and Internet access, MATHJAX allows an HTML page to display math without relying on a large number of external image files.[4] lwarp maintains LaTeX control for cross-referencing and equation numbering, and attempts to force MATHJAX to tag equations accordingly.

———————————

A *texlua* program called *lwarpmk* is used to process either the print or HTML version of the document. A few external utility programs are used to finish the conversion from a LaTeX-generated PDF file which happens to have HTML5 tags, to a number of HTML5 plain-text files and accompanying images.

lwarp automatically generates the extra files necessary for the HTML conversion, such as CSS and .xdy files, and configuration files for the utility *lwarpmk*. Also included is a parallel version of the user's source document, <sourcename>-html.tex, which selects HTML output and then inputs the user's own source. This process allows both the printed and HTML versions to co-exist side-by-side, each with their own auxiliary files.

When requesting packages during HTML conversion, lwarp first looks to see if it has its own modified version to use instead of the standard LaTeX version. These lwarp-packagename.sty files contain code used to emulate or replace functions for HTML output.

---

[2]FIREFOX has had an on-again/off-again bug for quite some time regareding printing SVGs at high resolution.

[3]There seems to be some debate as to whether MathML is actually an improvement over LaTeX for sharing math. The author has no particular opinion on the matter, except to say that in this case LaTeX is much easier to implement!

[4]One SVG image file per math expression, except that duplicate inline math expressions are combined into a single file according to the MD5 hash function of its contents. A common scientific paper can easily include several thousand files, and in one case the MD5 hash cut the number of files in half and the rendering time by 30%.

2.1 Typesetting conventions

Font weight, family, and style are used to indicate various objects:

Table 1: Typesetting conventions

package	LaTeX package.
program	Program's executable name.
option	Program or package option.
filename	File name in the operating system.
BRAND NAME	Proper name for a program, operating system, etc.
commands	Commands to be entered by the user.
code	Program code.
\macroname	LaTeX macro.
environment	LaTeX environment.
counter	LaTeX counter.
boolean	LaTeX boolean.
<element>	HTML element.
attribute	HTML attribute.
User Interface	A user-interface item.
ACRO	Acronym.

subjects Blue-colored tags in the left margin aid in quickly identifying the subject of each paragraph. These are often the targets of index entries.

Prog Lwarp

index entries

Black-colored tags in the left marign are used to identify programming objects such as files, packages, environments, booleans, and counters. Items without a tag are command macros. Each of these also appears in the index as individual entries, and are also listed together under "files", "packages", "environments", "booleans", and "counters".

⚠ warnings Special warnings are marked with a warning icon.

2.2 Supported packages and features

Table 2 lists some of the various LaTeX features and packages which may be used.

Package names are colored according to their support level:

name: Supported as-is.

name: Modifed to work with HTML output, and perhaps also as print output in SVG math or `lateximage` environments.

name: Emulated for HTML output.

name: Ignored for HTML output, but provides source-level compatibility.

MJ: Supported as-is for MathJax, subject to limitations.

MJ: Emulated for MathJax using custom macros, subject to limitations.

MJ: Ignored by MathJax, but may be used in the document source. May be converted to SVG images.

Table 2: LaTeX lwarp package — Supported features

Category	Status and supported features.
Engines:	DVI LaTeX, PDF LaTeX, XƎLaTeX, LuaLaTeX, upLaTeX
LaTeX compiling:	*latexmk*, *make*, etc.
External compiling:	*perltex*, *pythontex*, sympytex
Classes:	article, book, report, scrartcl, scrbook, scrreprt, memoir, CJK-related as listed below.
Koma-script:	scrextend, scrhack, scrlayer. Others as listed below.
Memoir:	memhfixc
Beamer:	beamerarticle, but not the **beamer** class.
Languages:	babel, cjkpunct, impnattypo, luavlna, polyglossia, xeCJK, xevlna.
Chinese:	CTEX, ctex, upzhkinsoku, xpinyin, zhlineskip, zhspacing.
Japanese:	upLaTeX, LuaTeX-ja, gentombow, lltjext, plarray, plarydshln, plautopatch, plext, plextarray, plextarydshln, plextcolortbl, plextdelarray, pxatbegshi, pxeveryshi, pxftnright, pxgentombow, pxjahyper, pxpdfpages, pxpgfrcs, pxpgfmark, tascmac, zxjatype. bxjsarticle and related, ltjsarticle and related, luatexja, luatexja-fontspec, ujarticle and related, utarticle and related.
Korean:	kotex, luatexko, xetexko.

Page layout:	2in1, 2up, a4, a4wide, a5comb, addlines, anysize, atbegshi, balance, blowup, booklet, bophook, bounddvi, bxpapersize, canoniclayout, centerlastline, changelayout, changepage, chngpage, clrdblpg, continue, draftcopy, draftfigure, draftwatermark, ebook, everyshi, fancyhdr, fancytabs, flippdf, fullminipage, fullpage, fwlw, geometry, gmeometric, grid, grid-system, gridset, layaureo, layout, layouts, leading, lscape, ltxgrid, nccfancyhdr, notespages, nowidow, pagegrid, pagesel, parallel, parcolumns, pbalance, pdfcolparallel, pdfcolparcolumns, pdfcrypt, pdflscape, pdfprivacy, preview, ragged2e, returntogrid, rmpage, scrlayer-scrpage, scrpage2, setspace, selectp, textarea, threadcol, thumb, thumbs, titleps, tocenter, turnthepage, twoup, typearea, underlin, vmargin, watermark, widows-and-orphans, zwpagelayout.
Sectioning:	Adds FileDepth for splitting the HTML output. Files may be numbered sequentially or named according to section name. Common short words and punctuation are removed from the filenames. anonchap, bsheaders, decorule, fncychap, froufrou, hypbmsec, indentfirst, quotchap, section, sectionbreak, secdot, sectsty, titlesec, tocvsec2.
Table of contents, figures, tables:	Supported, with hyperlinks. etoc, minitoc, multitoc, shorttoc, tableof, titletoc, tocbasic, tocbibind, tocdata, tocloft, tocstyle, tocvsec2.
Title page:	\maketitle, titlepage, authblk, authoraftertitle, titling.
Front & back matter:	abstract, appendix.
Indexing:	*makeindex*, *xindy*, and *xindex* are supported, with hyperlinks. gindex, hvindex, idxlayout, imakeidx, index, makeidx, repeatindex, splitidx, varindex, xindex.
Glossary:	gloss, glossaries and *xindy*, nomencl.
Bibliography:	babelbib, bibtopic, backref, biblatex, bibunits, chapterbib, cite, citeref, collref, drftcite, hypernat, jurabib, mcite, mciteplus, multibib, natbib, notes2bib, splitbib, showtags.
Cross-references:	bookmark, breakurl, cleveref, fancyref, hypdestopt, hyperref, perpage, prettyref, titleref, url, varioref, xcite, xr, xr-hyper, xurl, zref.
Margin notes:	marginal, marginfit, marginfix, scrlayer-notecolumn, versonotes.

Footnotes:	Adds `FootnoteDepth` to print footnotes at section breaks. MATHJAX emulation for `\footnote`, and also as marked in the following: bigfoot, dblfnote, endheads, endnotes^{MJ}, enotez^{MJ}, fixfoot, fnbreak, fnpara, fnpct, fnpos, footmisc, footnote, footnotebackref, footnoterange, footnpag, manyfoot, marginnote^{MJ}, nccfoots^{MJ}, pagenote^{MJ}, parnotes^{MJ}, pdfcolfoot, pfnote, sepfootnotes, sidenotes^{MJ}, tablefootnote.
Math:	Converted to SVG images with HTML `<alt>` tags containing the LATEX source for the math expression. MATHJAX supported as an alternative. amsmath^{MJ}: $\mathcal{AMS}$ environments are supported. User-defined macros are available during converson, due to native LATEX processing.
Theorems:	Native LATEX theorems, amsthm, apxproof, ntheorem, shadethm, theorem, thmbox, thmtools.
Additional math:	Math fonts via SVG images, accents^{MJ}, amscd^{MJ}, amscdx, autobreak^{MJ}, autonum, backnaur^{MJ}, bm^{MJ}, braket^{MJ}, breqn^{MJ}, bussproofs^{MJ}, cases^{MJ}, centernot^{MJ}, cmbright^{MJ}, colonequals^{MJ}, decimal^{MJ}, delarray, DotArrow^{MJ}, dotlessi^{MJ}, dotlessj^{MJ}, esvect^{MJ}, extarrows^{MJ}, fixmath^{MJ}, fouridx^{MJ}, fourier^{MJ}, guass, hhtensor^{MJ}, icomma^{MJ}, isomath^{MJ}, jkmath, kpfonts^{MJ}, kpfonts-otf^{MJ}, leftidx^{MJ}, libertinust1math^{MJ}, mathalpha^{MJ}, mathastext^{MJ}, mathcomp^{MJ}, mathdesign^{MJ}, mathdots^{MJ}, mathfixs^{MJ}, mathpazo^{MJ}, mathptmx^{MJ}, mathpunctspace^{MJ}, mathspec^{MJ}, mathtools^{MJ}, mattens^{MJ}, maybemath^{MJ}, mdwmath^{MJ}, mismath^{MJ}, mleftright^{MJ}, multiobjective^{MJ}, nccmath^{MJ}, nicematrix^{MJ}, noitcrul^{MJ}, newpxmath^{MJ}, newtxmath^{MJ}, newtxsf^{MJ}, pb-diagram, pxfonts^{MJ}, resizegather^{MJ}, rmathbr^{MJ}, scalerel^{MJ}, shuffle^{MJ}, skmath^{MJ}, stackrel^{MJ}, statex2^{MJ}, statistics, statmath^{MJ}, subsupscripts^{MJ}, tensind, tensor^{MJ}, textualicomma^{MJ}, txfonts^{MJ}, txgreeks^{MJ}, unicode-math^{MJ}, upgreek^{MJ}, ushort^{MJ}, witharrows^{MJ}, xfakebold^{MJ}, xy. Many others work as-is.
Display math with `\displaymathother`:	Complicated math objects in display math, such as tikz-cd, etc.
Units and fractions:	nicefrac^{MJ}, Slunits^{MJ}, siunitx^{MJ}, units^{MJ}, unitsdef, xfrac^{MJ}.

Floats:	Appear where declared. capt-of, caption, cutwin, dblfloatfix, endfloat, fewerfloatpages, fix2col, flafter, float, floatflt, floatrow, fltrace, ftcap, hypcap, keyfloat, morefloats, multicap, newfloat, nonfloat, picinpar, placeins, rotfloat, stfloats, subcaption, subfig, subfigure, subfloat, swfigure, topcapt, trivfloat, wrapfig, wrapfig2.
Tabular:	tabular environment, array^{MJ}, arydshln^{MJ}, bigdelim^{MJ}, bigstrut^{MJ}, booktabs^{MJ}, colortbl^{MJ}, ctable, dcolumn, diagbox, hhline^{MJ}, longtable, ltablex, ltxtable, multirow^{MJ}, supertabular, tabularx, tabulary, threeparttable, threeparttablex, widetable, xltabular, xtab.
Graphics:	graphics and graphicx. \includegraphics supports width, height, origin, angle, and scale tags, and adds class. References to PDF files are changed to SVG, other image types are accepted as well. \rotatebox and \scalebox are supported as well as HTML can handle. rotating is emulated but all objects are unrotated in HTML. picture, tikz, and xy are converted to an SVG image.

asymptote, curves, datatool, eepic, epsf, epsfig, epstopdf, figsize, fitbox, grffile, lpic, luamplib, media9, movie15, multimedia, overpic, pict2e, pinlabel, psfrag, psfragx, pst-eps, pstool, pstricks, rlepsf, rviewport, svg, svg-extract, tikz, tikz-3dplot, tikz-imagelabels, xy |
xcolor:	Full package color names, any color models, and mixing. \textcolor, \colorbox , \fcolorbox . Enhanced for HTML compatibility.
Lists:	Standard LaTeX environments, enumerate, enumitem, eqlist, hang, listliketab, paralist.
Environments:	Standard LaTeX environments.
Paragraphs, minipage, \parbox:	Some HTML5-imposed limitations. Nested minipages are supported. eqparbox, fancypar, minibox, pbox, shapepar.
Quotations:	copyrightbox, csquotes, epigraph, quoting, verse.
Verbatim:	fancyvrb, fvextra, moreverb, shortvrb, verbatim.
Frames:	boxedminipage, boxedminipage2e, fancybox, fbox^{MJ}, framed, mdframed, niceframe, shadow, tcolorbox^{MJ}, vertbars.
Multi-columns:	adjmulticol, multicol, multicolrule, vwcol.
Margins:	fullwidth, hanging, midpage.
Line numbering:	fnlineno, lineno.

Direct formatting:	\emph, \textsuperscript, \textbf, etc are supported. \bfseries, etc. are only supported in some cases. cancel[MJ], ellipsis, embrac, enparen, hyphenat, lettrine, lips, lua-check-hyphen, luacolor, magaz, moresize, nolbreaks, normalcolor, pdfcol, pdfcolmk, pdfrender, realscripts, relsize[MJ], scalefnt, seqsplit[MJ], soul, soulpos, soulutf8, stackengine, textfit, thinsp, trimclip, truncate, ulem, umoline, underscore, uspace, xellipsis.
Acronyms:	acro, acronym.
Ordinals:	engord, fmtcount, nth.
Text ligatures:	Ligatures for symbols are supported. Ligatures for f, q, t are intentionally turned off because many simpler browsers do not display them correctly. Modern full-featured browsers re-create these ligatures on-the-fly.
Horizontal space:	HTML output for thin-unbreakable, unbreakable, \enskip, \quad, \qquad, \hspace.
Rules:	\rule with width, height, raise, text color.
HTML reserved characters:	\&, \textless, and \textgreater are converted to HTML entities.
Fonts:	Used as-is. Appear in SVG math expressions or embedded image environments. fontaxes, nfssext-cfr, slantsc, tabfigures.
	Tested to work as-is: Special font macros in cfr-lm and others which use nfssext-cfr.
	Also see the math section for math and MATHJAX support for math font packages.
Symbols:	Native LaTeX diacriticals, academicons, amssymb[MJ], bbding, ccicons, chemgreek, dingbat, euro, eurosym, fontawesome, fontawesome5, gensymb[MJ], latexsym[MJ], marvosym, metalogo, metalogox, pifont, textalpha, textcomp[MJ], textgreek, typicons, xunicode.
Files:	attachfile, attachfile2, hyperxmp, inputtrc, intopdf, pdfpages, pdfx, xmpincl.

Science and engineering:	algorithm2e, algorithmicx, ar[MJ], askmaps, axodraw2, bitpattern, blochsphere, bodegraph, bohr, bytefield, chemfig, chemformula, chemgreek, chemmacros, chemnum, circuitikz, doipubmed, econometrics[MJ], elements, engtlc[MJ], fast-diagram, ghsystem, hepnicenames, heppennames, hepunits[MJ], isotope[MJ], karnaughmap, karnaugh-map, keystroke, listings, listingsutf8, linop, menukeys, mhchem[MJ], minted, pgfgantt, phfqit, physics[MJ], physunits[MJ], plimsoll[MJ], qcircuit, register, simplebnf, simpler-wick, slashed[MJ], steinmetz[MJ], structmech, struktex, syntaxdi, tikz-karnaugh, tikzcodeblocks, venndiagram
Arts and humanities:	foreign, forest, lyluatex, musicography, nameauth, octave, phonrule, piano, schemata, semantic-markup, tikz-dependency, vowel, xpiano
Academic:	academicons, classicthesis, doi, doipubmed, orcidlink[MJ], termcal
Admonitions:	awesomebox, notes.
Editorial:	changebar, changelog, changes, easy-todo, easyReview, ed, errata, fixme, fixmetodonotes, pdfcomment[MJ], pdfmarginpar, todo, todonotes, tram, xechangebar.
Accessibility:	accessibility[MJ], accsupp[MJ], axessibility[MJ], pdfcomment[MJ], repltext[MJ], tagpdf.
Package handling:	catoptions.
Debug:	chkfloat, cmdtrack, dprogress, lipsum, lua-visual-debug, mwe, refcheck, showlabels, showkeys, srcltx, srctex, vpe, xbmks.
Working as-is:	Various utility, calculation, file, and text-only packages, such as calc, fileerr, somedefs, trace, xspace. Also, most math-only packages, including specialized typesetting for various fields of science and engineering.

3 Alternatives

Summarized below are several other ways to convert a LaTeX or other document to HTML. Where an existing LaTeX document is to be converted to HTML, lwarp may be a good choice. For new projects with a large number of documents, it may be worth investigating the alternatives before decided which path to take.

3.1 internet class

internet (*Cls*) The closest to lwarp in design principle is the internet class by Andrew Stacey — an interesting project which directly produces several versions of markdown, and also HTML and EPUB. https://github.com/loopspace/latex-to-internet

3.2 TeX4ht

TeX4ht (*Prog*) http://tug.org/tex4ht/

htlatex (*Prog*)

This system uses native LaTeX processing to produce a DVI file containing special commands, and then uses additional post-processing for the HTML conversion by way of numerous configuration files. In some cases lwarp provides a better HTML conversion, and it supports a different set of packages. TeX4ht produces several other forms of output beyond HTML, including ODT and a direct path to EPUB, and is still being developed.

3.3 Translators

These systems use external programs to translate a subset of LaTeX syntax into HTML. Search for each on CTAN (http:\ctan.org).

Hevea (*Prog*) **H^{E}v^{E}a:** http://hevea.inria.fr/ (not on CTAN)

TtH (*Prog*) **T$_T$H:** http://hutchinson.belmont.ma.us/tth/

GELLMU (*Prog*) **GELLMU:** http://www.albany.edu/~hammond/gellmu/

LaTeXML (*Prog*) **LaTeXML:** http://dlmf.nist.gov/LaTeXML/

Plastex (*Prog*) **PlasTeX:** https://github.com/tiarno/plastex

LaTeX2HTML (*Prog*) **LaTeX2HTML:** http://www.latex2html.org/ and http://ctan.org/pkg/latex2html.

TeX2page (*Prog*) **TeX2page:** http://ds26gte.github.io/tex2page/index.html

Finally, GladTeX may used to directly insert LaTeX math into HTML:

GladTeX (*Prog*) **GladTeX:** http://humenda.github.io/GladTeX/

3.4 AsciiDoc and Asciidoctor

AsciiDoc is one of the most capable markup languages, providing enough features to produce the typical technical-writing document with cross-references, and it writes LaTeX and HTML.

AsciiDoc (*Prog*) **Asciidoctor:** http://asciidoctor.org/ (More active.)

AsciiDoctor (*Prog*) **AsciiDoc:** http://asciidoc.org/ (The original project.)

3.4.1 Asciidoctor-LaTeX

The Asciidoctor-LaTeX project is developing additional LaTeX-related features.

Asciidoctor-LateX:
http://www.noteshare.io/book/asciidoctor-latex-manual

Asciidoctor-LaTeX (*Prog*) https://github.com/asciidoctor/asciidoctor-latex

3.5 Pandoc

Pandoc (*Prog*) A markup system which also reads and writes LaTeX and HTML.

Pandoc: http://pandoc.org/

(Watch for improvements in cross-references to figures and tables.)

3.6 Word processors

Word (*Prog*) It should be noted that the popular word processors have advanced through the
LibreOffice (*Prog*) years in their abilities to represent math with a LaTeX-ish input syntax, unicode
OpenOffice (*Prog*) math fonts, and high-quality output, and also generate HTML with varying success.
See recent developments in Microsoft ® *Word* ® and LibreOffice ™ *Writer*.

3.7 Commercial systems

Adobe (*Prog*) Likewise, several professional systems exist whose abilities have been advancing
FrameMaker (*Prog*) in the areas of typesetting, cross-referencing, and HTML generation. See Adobe ®
FrameMaker ®, Adobe *InDesign* ®, and Madcap *Flare* ™.
InDesign (*Prog*)

Flare (*Prog*)

Madcap (*Prog*) ### 3.8 Comparisons

AsciiDoc, Pandoc, and various other markup languages typically have a syntax which tries to be natural and human-readable, but the use of advanced features tends to require many combinations of special characters, resulting in a complicated mess of syntax. By contrast, LaTeX spells things out in readable words but takes longer to type, although integrated editors exist which can provide faster

entry and a graphic user interface. For those functions which are covered by the typical markup language it is arguable that LaTeX is comparably easy to learn, while LaTeX provides many more advanced features where needed, along with a large number of pre-existing packages which provide solutions to numerous common tasks.

Text-based document-markup systems share some of the advantages of LaTeX vs. a typical word processor. Documents formats are stable. The documents themselves are portable, work well with revision control, do not crash or become corrupted, and are easily generated under program control. Formatting commands are visible, cross-referencing is automatic, and editing is responsive. Search/replace with regular expressions provides a powerful tool for the manipulation of both document contents and structure. Markup systems and some commercial systems allow printed output through a LaTeX back end, yielding high-quality results especially when the LaTeX template is adjusted, but they lose the ability to use LaTeX macros and other LaTeX source-document features.

The effort required to customize the output of each markup system varies. For print output, LaTeX configuration files are usually used. For HTML output, a CSS file will be available, but additional configuration may require editing some form of control file with a different syntax, such as XML. In the case of lwarp, CSS is used, and much HTML output is adjusted through the usual LaTeX optional macro parameters, but further customization may require patching LaTeX code.

The popular word processors and professional document systems each has a large base of after-market support including pre-designed styles and templates, and often include content-management systems for topic reuse.

4 Installation

Table 3 shows the tools which are used for the LaTeX to HTML conversion. In most cases, these will be available via the standard package-installation tools.

Detailed installation instructions follow.

Table 3: Required software programs

Provided by your LATEX distribution:

> From TEXLive: http://tug.org/texlive/.

> **LATEX:** *pdflatex*, *xelatex*, or *lualatex*.

> **The lwarp package:** This package.

> **The *lwarpmk* utility:** Provided along with this package. This should be an operating-system executable in the same way that *pdflatex* or *latexmk* is. It is possible to have the lwarp package generate a local copy of *lwarpmk* called lwarpmk.lua. See table 4.

> ***luatex*:** Used by the *lwarpmk* program to simplify and automate document generation.

> ***xindy*:** The *xindy* program is used by lwarp to create indexes. On a MiKTEX system this may have to be acquired separately, but it is part of the regular installer as of mid 2015.

> ***latexmk*:** Optionally used by *lwarpmk* to compile LATEX code. On a MiKTEX system, *Perl* may need to be installed first.

> ***pdfcrop*:** Used to pull images out of the LATEX PDF.

POPPLER PDF utilities:

> ***pdftotext*:** Used to convert PDF to text.

> ***pdfseparate*:** Used to pull images out of the LATEX PDF.

> ***pdftocairo*:** Used to convert images to SVG.

> These might be provided by your operating-system package manager, and MiKTEX provides miktex-poppler-bin-* packages.

> From POPPLER: poppler.freedesktop.org.

> For MACOS®, see https://brew.sh/, install *Homebrew*, then

> > Enter ⇒ **brew install poppler**

> For WINDOWS, see MikTEX miktex-poppler-bin-*, or:
> https://sourceforge.net/projects/poppler-win32/ and:
> http://blog.alivate.com.au/poppler-windows/

***Perl*:**

> This may be provided by your operating-system package manager, and may be required for some of the POPPLER PDF utilities.

> strawberryperl.com (recommended), perl.org

Automatically downloaded from the internet as required:

> **MATHJAX:** Optionally used to display math. From: mathjax.org

---

4.1   Installing the lwarp package

There are several ways to install lwarp. These are listed here with the preferred methods listed first:

Pre-installed: Try entering into a command line:

> Enter ⇒   `kpsewhich lwarp.sty`

If a path to `lwarp.sty` is shown, then lwarp is already installed and you may skip to the next section.

TEX Live: If using a TEX Live distribution, try installing via *tlmgr*:

> Enter ⇒   `tlmgr install lwarp`

MiKTEX:

1. For newer versions of MiKTEX, install or update lwarp using the *MiKTeX Console* program.

2. For older versions of MiKTEX, to install lwarp the first time, use the *MiKTeX Package Manager (Admin)*. To update lwarp, use *MiKTeX Update (Admin)*.

3. Either way, also update the package miktex-misc, which will install and update the *lwarpmk* executable.

Operating-system package: The operating-system package manager may already have lwarp, perhaps as part of a set of TEX-related packages.

CTAN TDS archive: lwarp may be downloaded from the Comprehensive TEX Archive:

1. See http://ctan.org/pkg/lwarp for the lwarp package.

2. Download the TDS archive: `lwarp.tds.zip`

3. Find the TEX local directory:

   TEX Live:
   > Enter ⇒   `kpsewhich -var-value TEXMFLOCAL`

   MiKTEX:
   In the **Settings** window, **Roots** tab, look for a local TDS root.

   This should be something like:

   > `/usr/local/texlive/texmf-local/`

4. Unpack the archive in the TDS local directory.

5. Renew the cache:

   > Enter ⇒   `mktexlsr`

   — or —

   > Enter ⇒   `texhash`

   Or, for WINDOWS MiKTEX, start the program called *MiKTeX Settings (Admin)* and click on the button called **Refresh FNDB**.

CTAN `.dtx` and `.ins` files: Another form of TEX package is `.dtx` and `.ins` source files. These files are used to create the documentation and `.sty` files.

1. See http://ctan.org/pkg/lwarp for the lwarp package.

2. Download the zip archive `lwarp.zip` into your own `lwarp` directory.

3. Unpack `lwarp.zip`.

4. Locate the contents `lwarp.dtx` and `lwarp.ins`

5. Create the `.sty` files:

   Enter ⇒ **`pdflatex lwarp.ins`**

6. Create the documentation:

   ```
   pdflatex lwarp.dtx   (several times)
   makeindex -s gglo.ist -o lwarp.gls lwarp.glo
   makeindex -s gind.ist lwarp.idx
   pdflatex lwarp.dtx   (several times)
   ```

7. Copy the `.sty` files somewhere such as the TEX Live local tree found in the previous CTAN TDS section, under the subdirectory:

   `<texlocal>/tex/latex/local/lwarp`

8. Copy `lwarp_baseline_marker.png` and `lwarp_baseline_marker.eps` to the same place as the `.sty` files.

9. Copy the documentation `lwarp.pdf` to a source directory in the local tree, such as:

   `<texlocal>/doc/local/lwarp`

10. Renew the cache:

    Enter ⇒ **`mktexlsr`**

    — or —

    Enter ⇒ **`texhash`**

    Or, for WINDOWS MiKTEX, start the program called *MiKTeX Settings (Admin)* and click on the button called **Refresh FNDB**.

11. See section 4.2.1 to generate your local copy of *lwarpmk*.

12. Once the local version of `lwarpmk.lua` is installed, it may be made available system-wide as per section 4.2.

Project-local CTAN `.dtx` and `.ins` files: The `.dtx` and `.ins` files may be downloaded to a project directory, then compiled right there, alongside the document source files. The resultant `*.sty` and `lwarpmk.lua` files may be used as-is, so long as they are in the same directory as the document source. The files `lwarp_baseline_marker.png` and `lwarp_baseline_marker.eps` must also be copied as well. This approach is especially useful if you would like to temporarily test lwarp before deciding whether to permanently install it.

Just testing!

4.2 Installing the *lwarpmk* utility

(Note: If *lwarpmk* is not already installed, it is easiest to use a local copy instead of installing it system-wide. See section 4.2.1.)

After the lwarp package is installed, you may need to setup the *lwarpmk* utility:

1. At a command line, try executing **`lwarpmk`**. If the *lwarpmk* help message appears, then *lwarpmk* is already set up. If not, it is easiest to generate and use a local copy. See section 4.2.1.

2. For MiKTEX, try updating the miktex-misc package. This may install the *lwarpmk* executable for you.

   Otherwise, continue with the following:

3. Locate the file lwarpmk.lua, which should be in the scripts directory of the TDS tree. On a TEX Live or MiKTEX system you may use

   Enter ⇒ **`kpsewhich lwarpmk.lua`**

   (If the file is not found, you may also generate a local copy and use it instead. See section 4.2.1.)

4. Create *lwarpmk*:

   Unix: Create a symbolic link and make it executable:

   (a) Locate the TEX Live binaries:

   Enter ⇒ **`kpsewhich -var-value TEXMFROOT`**

   This will be something like:

   `/usr/local/texlive/<year>`

   The binaries are then located in the bin/<arch> directory under the root:

   `/usr/local/texlive/<year>/bin/<architecture>/`

   In this directory you will find programs such as *pdflatex* and *makeindex*.

   (b) In the binaries directory, create a new symbolic link from the binaries directory to lwarpmk.lua:

   Enter ⇒ **`ln -s <pathtolwarpmk.lua> lwarpmk`**

   (c) Make the link executable:

   Enter ⇒ **`chmod 0755 lwarpmk`**

   WINDOWS TEX Live: Create a new lwarpmk.exe file:

   (a) Locate the TEX Live binaries as shown above for UNIX.

   (b) In the binaries directory, make a *copy* of runscript.exe and call it lwarpmk.exe This will call the copy of lwarpmk.lua which is in the scripts directory of the distribution.

   WINDOWS MiKTEX: Create a new lwarpmk.bat file:

   (a) Locate the MiKTEX binaries. These will be in a directory such as:

   `C:\Program Files\MiKTeX 2.9\miktex\bin\x64`

   In this directory you will find programs such as pdflatex.exe and makeindex.exe.

   (b) Create a new file named lwarpmk.bat containing:

   `texlua "C:\Program Files\MiKTeX 2.9\scripts\lwarp\lwarp.texlua" %*`

   This will call the copy of lwarpmk.lua which is in the scripts directory of the distribution.

4.2.1 Using a local copy of *lwarpmk*

It is also possible to use a local version of *lwarpmk*:

1. When compiling the tutorial in section 5, use the lwarpmk option for the lwarp package:

   `\usepackage[lwarpmk]{lwarp}`

2. When the tutorial is compiled with *pdflatex*, the file lwarpmk.lua will be generated along with the other configuration files.

3. lwarpmk.lua may be used for this project:

Unix:

(a) Make lwarpmk.lua executable:

   Enter ⇒ **chmod 0755 lwarpmk.lua**

(b) Compile documents with

   Enter ⇒ **./lwarpmk.lua html**

   Enter ⇒ **./lwarpmk.lua print**

   etc.

(c) It may be useful to rename or link to a version without the .lua suffix.

WINDOWS:

Compile documents with either of the following, depending on which command shell is being used:

   Enter ⇒ **texlua lwarpmk.lua html**

   Enter ⇒ **texlua lwarpmk.lua print**

   etc.

Or:

   Enter ⇒ **lwarpmk html**

   Enter ⇒ **lwarpmk print**

   etc.

4.3   Installing additional utilities

To test for the existence of the additional utilities:

Enter the following in a command line. If each programs' version is displayed, then that utility is already installed. See table 3 on page 79.

   Enter ⇒ **luatex --version**

   Enter ⇒ **xindy --version**

   Enter ⇒ **latexmk --version**

   Enter ⇒ **perl --version**

   Enter ⇒ **pdfcrop --version**

   Enter ⇒ **pdftotext -v**

   Enter ⇒ **pdfseparate --version**

   Enter ⇒ **pdftocairo -v**

To install *xindy*, *latexmk*, and *pdfcrop*:

The TEX utilities *xindy*, *latexmk*, and *pdfcrop* may be installed in *TexLive* with *tlmgr*, installed by *MiKTeX*, provided by your operating system's package manager, or downloaded from the *CTAN* archive:

http://ctan.org/pkg/xindy
http://ctan.org/pkg/latexmk
http://ctan.org/pkg/pdfcrop

pdftotext (*Prog*) [requirement]   **To install the POPPLER utilities to a UNIX/LINUX system:**

pdfseparate (*Prog*) [requirement]   The tools from the POPPLER project should be provided by your operating
pdftocairo (*Prog*) [requirement]   system's package manager.

To install the POPPLER utilities to a MACOS machine:

1. Install *Homebrew* from https://brew.sh/:

/usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"

2. Install the POPPLER utilities:

Enter ⇒ **`brew install poppler`**

To install the POPPLER utilities to a WINDOWS machine:

If using MikTEX, install a miktex-poppler-bin-* package. Otherwise:

1. See table .

2. Download and extract the POPPLER utilities *pdftotext*, *pdfseparate*, and *pdfseparate* to a directory, such as Poppler.

3. In the **Start** window, type "Path" to search for results related to Path. Or, open the control panel and search for "Path".

4. Choose **Edit the system environment variables** in the control panel.

5. Choose the **Environment Variables** button.

6. Choose the **Path** variable, then the **Edit** button.

7. Choose the **New** button to make an additional entry.

8. Enter the `bin` directory of the POPPLER utilities, such as:

`C:\Users\<myname>\Desktop\Poppler\poppler-0.5_x86\poppler-0.5\bin`

Be sure to include `\bin`.

9. Click **Ok** when done.

perl (*Prog*) [requirement]   **To install PERL to a WINDOWS machine:**

1. Download and install a version of PERL, such as STRAWEBERRY PERL, to a directory without a space in its name, such as `C:\Strawberry`.

2. Edit the **Path** as seen above for the POPPLER utilities.

3. Enter the `bin` directory of the *perl* utility, such as:

`C:\Strawberry\perl\bin`

Be sure to include `\bin`.

4. Click **Ok** when done.

Any utilities installed by hand must be added to the PATH**.**

5 Tutorial

This section shows an example of how to create an lwarp document.

Need help? See the General Index for "how-to", and the Troubleshooting Index if something doesn't work. A Troubleshooting section is also available. The Index of Objects contains automated entries for each package, macro, environment, counter, boolean, and other objects; individually and also sorted by category.

5.1 Starting a new project

1. Create a new project directory called tutorial.

tutorial.tex (*file*)

2. Inside the tutorial directory, create a new file called tutorial.tex. This may be done several ways:

   Copy from the documentation PDF:

   A listing is in fig. 1, which may be copied/pasted from the figure directly into your own editor, depending on the quality of the PDF viewer and editor, or:

   Copy from the lwarp documentation directory:

lwarp_tutorial.txt (*file*)

⚠ Note: .txt suffix!

   Another copy may be found by entering into a command line:

   Enter ⇒ **texdoc -l lwarp_tutorial.txt**

   This should be in the doc/latex/lwarp/ directory along with this PDF documentation. Copy lwarp_tutorial.txt directly into your tutorial directory, renamed as tutorial.tex.

⚠ Bad formatting!

   When using Windows, use an editor other than Notepad, since Notepad does not accept the end-of-line from a Unix text file.

3. Compile the project:

   Enter ⇒ **pdflatex tutorial.tex**

   (several times)

   (*xelatex* or *lualatex* may be used as well. lwarp also supports DVI *latex* for use with .eps images.)

4. View the resulting tutorial.pdf with a PDF viewer.

A number of new files are created when tutorial.tex is compiled, as shown in table 4. These files are created by the lwarp package.

(Two of the new files are configuration files for the helper program *lwarpmk*. Whenever a print version of the document is created, the configuration files for *lwarpmk* are updated to record the operating system, LaTeX engine (*latex*, *pdflatex*, *xelatex*, or *lualatex*), the filenames of the source code and HTML output, and whether the additional helper program *latexmk* will be used to compile the document.)

Figure 1: tutorial.tex listing

Note: There are two pages!

```
% Save this as tutorial.tex for the lwarp package tutorial.

\documentclass{book}

\usepackage{iftex}

% --- LOAD FONT SELECTION AND ENCODING BEFORE LOADING LWARP ---

\ifPDFTeX
\usepackage{lmodern}             % pdflatex or dvi latex
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\else
\usepackage{fontspec}            % XeLaTeX or LuaLaTeX
\fi

% --- LWARP IS LOADED NEXT ---
\usepackage[
%   HomeHTMLFilename=index,      % Filename of the homepage.
%   HTMLFilename={node-},        % Filename prefix of other pages.
%   IndexLanguage=english,       % Language for xindy index, glossary.
%   latexmk,                     % Use latexmk to compile.
%   OSWindows,                   % Force Windows. (Usually automatic.)
%   mathjax,                     % Use MathJax to display math.
]{lwarp}
% \boolfalse{FileSectionNames}  % If false, numbers the files.

% --- LOAD PDFLATEX MATH FONTS HERE ---

% --- OTHER PACKAGES ARE LOADED AFTER LWARP ---
\usepackage{makeidx} \makeindex
\usepackage{xcolor}              % (Demonstration purposes only.)
\usepackage{hyperref,cleveref}   % LOAD THESE LAST!

% --- LATEX AND HTML CUSTOMIZATION ---
\title{The Lwarp Tutorial}
\author{Some Author}
\setcounter{tocdepth}{2}         % Include subsections in the \TOC.
\setcounter{secnumdepth}{2}      % Number down to subsections.
\setcounter{FileDepth}{1}        % Split \HTML\ files at sections
\booltrue{CombineHigherDepths}   % Combine parts/chapters/sections
\setcounter{SideTOCDepth}{1}     % Include subsections in the side\TOC
\HTMLTitle{Webpage Title}        % Overrides \title for the web page.
\HTMLAuthor{Some Author}         % Sets the HTML meta author tag.
\HTMLLanguage{en-US}             % Sets the HTML meta language.
\HTMLDescription{A description.}% Sets the HTML meta description.
\HTMLFirstPageTop{Name and \fbox{HOMEPAGE LOGO}}
\HTMLPageTop{\fbox{LOGO}}
\HTMLPageBottom{Contact Information and Copyright}
\CSSFilename{lwarp_sagebrush.css}

\begin{document}

\maketitle                       % Or titlepage/titlingpage environment.
```

```
% An article abstract would go here.

\tableofcontents                    % MUST BE BEFORE THE FIRST SECTION BREAK!
\listoffigures

\chapter{First chapter}

\section{A section}

This is some text which is indexed.\index{Some text.}

\subsection{A subsection}

See \cref{fig:withtext}.

\begin{figure}\begin{center}
\fbox{\textcolor{blue!50!green}{Text in a figure.}}
\caption{A figure with text\label{fig:withtext}}
\end{center}\end{figure}

\section{Some math}

Inline math: $r = r_0 + vt - \frac{1}{2}at^2$
followed by display math:
\begin{equation}
a^2 + b^2 = c^2
\end{equation}


\begin{warpprint}   % For print output ...
\cleardoublepage    % ... a common method to place index entry into TOC.
\phantomsection
\addcontentsline{toc}{chapter}{\indexname}
\end{warpprint}
\ForceHTMLPage       % HTML index will be on its own page.
\ForceHTMLTOC        % HTML index will have its own toc entry.
\printindex

\end{document}
```

Table 4: Configuration files created by print version

`tutorial.pdf`**:** The PDF output from LATEX. The print version of the document.

`tutorial_html.tex`**:** A small `.tex` file used to create a parallel HTML version of the document, which co-exists with usual the PDF version, and which will have its own auxiliary files. In this way, both PDF and HTML documents may co-exist side-by-side.

Auxiliary files: The usual LATEX files `.aux`, `.log`, `.out`, `.toc`, `.lof`, `.idx`. When an HTML version of the document is created, `_html` versions of the auxiliary files will also be generated.

`lwarpmk.conf`**:** A configuration file for *lwarpmk*, which is used to automate the compilation of PDF or HTML versions of the document.

`tutorial.lwarpmkconf`**:** Another configuration file used by *lwarpmk*, which is only useful if you wish to have several projects residing in the same directory.

`.css files`**:** `lwarp.css`, `lwarp_formal.css`, `lwarp_sagebrush.css` These files are standard for lwarp, and are not meant to be modified by the user.

`sample_project.css`**:** An example of a user-customized CSS file, which may be used for project-specific changes to the lwarp defaults.

`lwarp.ist`**:** Used by lwarp while creating an index using *makeindex*. This file should not be modified by the user. A custom file may be used instead, if necessary.

`lwarp.xdy`**:** Used by lwarp while creating an index using *xindy*. This file should not be modified by the user. A custom file may be used instead, if necessary.

`lwarp_one_limage.txt`**:** For WINDOWS only. Used to process SVG images in the background. Copied to `lwarp_one_limage.cmd` when images are generated.

`lwarp_mathjax.txt`**:** Inserted into the HTML files when MATHJAX is used to display math. Do not modify, see `\MathJaxFilename` instead.

`comment_*.cut`**:** Temporary files used by lwarp to conditionally process blocks of text. These files may be ignored.

When the `lwarpmk` option is given to the lwarp package:

`lwarpmk.lua`: A local copy of the *lwarpmk* utility.

On UNIX-related operating systems this file must be made executable:

```
chmod u+x lwarpmk.lua
```

This may be useful to have to archive with a project for future use.

5.2   Compiling the print version with *lwarpmk*

The *lwarpmk* utility program is used to compile either the printed or the HTML version of the document.

lwarpmk print is used to recompile a printed version of the document.

⚠ Enable *lwarpmk*
1. If you have not yet done so, add \usepackage{lwarp} to the document, then compile the project a single time using *pdflatex*, *lualatex*, or *xelatex*. This generates the file lwarpmk.conf, which then allows the *lwarpmk* program to be used.

2. Re-compile the print version:

   Enter ⇒   **lwarpmk print**

   lwarpmk prints an introduction then checks to see if the document must be recompiled. If it seems that the files are up-to-date, then *lwarpmk* informs you of that fact and then exits.

3. Make a small change in the original document, such as adding a space character.

4. Recompile again.

   Enter ⇒   **lwarpmk print**

   The document is recompiled when a change is seen in the source. Several compilations may be necessary to resolve cross-references.

5. Force a recompile to occur.

   Enter ⇒   **lwarpmk again**

   Enter ⇒   **lwarpmk print**

   lwarpmk again updates the date code for the file, triggering a recompile the next time the document is made.[5]

6. Process the index.[6] [7]

   Enter ⇒   **lwarpmk printindex**

7. Recompile again to include the index.

   Enter ⇒   **lwarpmk print**

8. To force a single recompile when needed, even if no changes were detected:

   Enter ⇒   **lwarpmk print1**

Note that the HTML customization commands are ignored while making the print version.

---

[5]Although, when using the utility *latexmk* (introduced later), the changed date is ignored and an actual change in contents must occur to cause a recompile.

[6]The command **lwarpmk printglossary** is also available to process a glossary produced with the glossaries package. See section 8.6.13.

[7]Also see section 8.6.16 for index options.

5.3 Compiling the HTML version with *lwarpmk*

lwarpmk html is used to recompile an HTML version of the document.

⚠ Enable *lwarpmk*
1. If you have not yet done so, add \usepackage{lwarp} to the document, then compile the project a single time using *pdflatex*, *lualatex*, or *xelatex*. This generates the file lwarpmk.conf, which then allows the *lwarpmk* program to be used.

2. Compile the HTML version:

    Enter ⇒ **lwarpmk html**

    (a) *lwarpmk* uses LaTeX to process tutorial_html.tex to create tutorial_html.pdf.

    (b) *pdftotext* is then used to convert to the file tutorial_html.html. This file is a plain-text file containing HTML tags and content for the entire document.

    (c) *lwarpmk* manually splits tutorial_html.html into individual HTML files according to the HTML settings. For this tutorial, the result is tutorial.html (the home page), along with First-chapter.html[8], Some-math.html, and the document's index in _Index.html.[9]

3. View the HTML page in a web browser.

    Open the file tutorial.html in a web browser.

math images
Note that math images have not yet been generated, so math is still displayed as its alt tag, which is set to the plain-text LaTeX source for that expression. Math may be displayed as SVG images (section 5.4) or by a MATHJAX script (section 5.5).

4. Force a recompile:

    Enter ⇒ **lwarpmk again**

    Enter ⇒ **lwarpmk html**

    Enter ⇒ **lwarpmk print**

5. Process the HTML index and recompile:[10][11]

    Enter ⇒ **lwarpmk htmlindex**

    Enter ⇒ **lwarpmk html**

    _Index.html is updated for the new LaTeX index.

6. Reload the web page to see the added index.

7. To force a single recompile when needed, even if no changes were detected:

    Enter ⇒ **lwarpmk html1**

---

[8]First-chapter.html also contains the first section, even though the second section is its own HTML page. This behavior is controlled by the boolean CombineHigherDepths.

[9]index.html is commonly used as a homepage, so the document index is in _Index.html.

[10]The command **lwarpmk htmlglossary** is also available to process a glossary produced with the glossaries package. See section 8.6.13.

[11]Also see section 8.6.16 for index options.

5.4   Generating the svg images

math as svg images   By default lwarp represents math as svg images, with the LaTeX source included in
`alt` attributes. In this way, the math is displayed as it was drawn by LaTeX, and the
LaTeX source may be copied and pasted into other documents.

picture and TikZ   lwarp uses the same mechanism for `picture` and TikZ environments.

1. Create the svg images:

   Enter ⇒   **`lwarpmk limages`**

   Enter ⇒   **`lwarpmk html`**

2. Move to the tutorial's html math page and reload the document in the
   browser.

3. The math images are displayed using the same font and formatting as the
   printed version.

4. Copy/paste a math expression into a text editor to see the LaTeX source.

⚠   adding/removing   When a math expression, `picture`, or TikZ environment is added or removed,
the svg images must be re-created by entering **`lwarpmk limages`** to maintain the
proper image-file associations. Inline svg math may be hashed and thus not need
to be recreated, but display math and objects such as TikZ may move to new image
numbers when the document is changed.

recompile first   Before attempting to create the svg image files, *lwarpmk* verifies that the html
version of the document exists and has correct internal image references.[12] If it
is necessary to recompile the document's html version one more time, *lwarpmk*
usually will inform the user with an error message, but there are some condi-
tions which cannot be detected, so the user should watch for the LaTeX recompile
warnings.

⚠   html instead of images   If html appears where an svg image should be, recompile the document one
more time to get the page numbers back in sync, then remake the images one
more time.

⚠   page counter   Incorrect svg images will also occur if the document changes the page counter:

`\setcounter{page}{<value>}`

The page counter must *not* be adjusted by the user.

⚠   Lots of files!   Expressing math as svg images has the advantage of representing the math exactly
as LaTeX would, but has the disadvantage of requiring an individual file for each
math expression. For inline math, and some other objects, lwarp uses an MD5 hash
on its LaTeX source to combine multiple instances of identical inline expressions
into a single image file, but display math and other environments such as `picture`
and TikZ require one image file each. For a document with a large amount of
math, see section 5.5 to use MathJax instead.

---

[12]This becomes important when dealing with a document containing thousands of images.

5.5 Using MathJax for math

math with MathJax Math may also be represented using the MathJax Javascript project.

1. In the tutorial's source code, uncomment the `mathjax` package option for lwarp:

   ```
   mathjax, % Use MathJax to display math.
   ```

2. Recompile

   Enter ⇒ **lwarpmk html**

3. Reload the math page.

⚠ MathJax requirements MathJax requires web access unless a local copy of MathJax is available, and it also requires that Javascript is enabled for the web page. The math is rendered by MathJax. Right-click on math to see several options for rendering, and for copying the LaTeX source.

While using MathJax has many advantages, it may not be able to represent complex expressions or spacing adjustments as well as LaTeX, and it may not support some math-related packages.

5.6 Changing the css style

For a formal css style, add to the preamble:

```
\usepackage{lwarp}
. . .
\CSSFilename{lwarp_formal.css}
. . .
\begin{document}
```

For a modern css style, `lwarp_sagebrush.css` is also provided:

```
\CSSFilename{lwarp_sagebrush.css}
```

See section 7.7 for more information about modifying the css styling of the document.

5.7 Customizing the html output

A number of settings may be made to control the html output, including filename generation, automatic compilation, math output, document splitting, meta data, and page headers and footers.

See section 7.6 for more information.

5.8 Using *latexmk*

latexmk is a LATEX utility used to monitor changes in source files and recompile as needed.

1. In the tutorial's source code uncomment the `latexmk` option for the lwarp package:

    ```
    latexmk, % Use latexmk to compile.
    ```

2. Recompile the printed version of the document.

    Enter ⇒ **`lwarpmk print`**

    lwarp updates its own configuration files (`lwarpmk.conf` and `tutorial.lwarpmkconf`) whenever the printed version of the document is compiled. These configuration files remember that *lwarpmk* should use *latexmk* to compile the document.

3. Recompile the document.

    Enter ⇒ **`lwarpmk print`**

    and/or

    Enter ⇒ **`lwarpmk html`**

Changes are detected by comparing checksums rather than modification times, so `lwarpmk` again will not trigger a recompile, but *latexmk* has a much better awareness of changes than the *lwarpmk* utility does and it is likely to correctly know when to recompile. A recompile may be forced by making a small change to forced single-pass recompile the source, and a single recompile may be forced with:

Enter ⇒ **`lwarpmk print1`**

and/or

Enter ⇒ **`lwarpmk html1`**

5.9   Using XƎLATEX or LuaLATEX

XƎLATEX or LuaLATEX may be used instead of LATEX.

1. Remove the auxiliary files for the project:

      Enter ⇒   **lwarpmk cleanall**

2. Use *xelatex* or *lualatex* to compile the printed version a single time.

      Enter ⇒   **xelatex tutorial.tex**

   — or —

      Enter ⇒   **lualatex tutorial.tex**

   When the compile occurs, the configuration files for *lwarpmk* are modified
   to remember which TEX engine was used. XƎLATEX or LuaLATEX will be used
   for future runs of *lwarpmk*.

3. To recompile the document:

      Enter ⇒   **lwarpmk print**

   -and-

      Enter ⇒   **lwarpmk html**

4. Also remember to update the indexes and recompile again:

      Enter ⇒   **lwarpmk htmlindex**

      Enter ⇒   **lwarpmk html**

      Enter ⇒   **lwarpmk printindex**

      Enter ⇒   **lwarpmk print**

5.10   Using DVI LATEX

Traditional DVI LaTeX may also be used along with .eps image files. An SVG version
of each image must also be provided. *lwarpmk* may be used to convert image
formats.

To convert EPS files to PDF:

   Enter ⇒   **lwarpmk epstopdf *.eps**   **(or a list of files)**

To convert PDF files to SVG:

   Enter ⇒   **lwarpmk pdftosvg *.pdf**   **(or a list of files)**

⚠ bitmapped fonts See section 7.4 regarding font selection to avoid the use of bitmapped fonts.

5.11   Using a bibliography

To process the bibliography for the HTML version:

   Enter ⇒   **bibtex <filename>_html**

or

   Enter ⇒   **biber <filename>_html**

To see the bibliography in the HTML version:

   Enter ⇒   **lwarpmk html1**

as many times as neccesary.

5.12   Using a glossary

lwarp supports the gloss and glossaries packages, although this tutorial does not
supply an example.

5.12.1   gloss package

See section 8.6.12.

5.12.2   glossaries package

To process the glossary for the print version:

   Enter ⇒   **lwarpmk printglossary**

⚠   (If **makeglossaries** is not found, see section 8.6.13.)

To process the glossary for the HTML version:

   Enter ⇒   **lwarpmk htmlglossary**

In each case, the document will have to be recompiled afterwards:

   Enter ⇒   **lwarpmk html1**

   Enter ⇒   **lwarpmk html**

   Enter ⇒   **lwarpmk print1**

   Enter ⇒   **lwarpmk print**

See section 8.6.13 to set options for processing glossaries.

5.13   Cleaning auxiliary files

To remove the auxiliary files .aux, .toc, .lof, .lot, .idx, .ind, .log, and .gl*, and a few others:

    Enter ⇒   **lwarpmk clean**

5.14   Cleaning auxiliary and output files

To remove the auxiliary files, and also remove the .pdf and .html files:

    Enter ⇒   **lwarpmk cleanall**

5.15   Cleaning the images from the `<project>-images` directory

The `<project>-images` directory contains svg images automatically generated for inline and display math, tikz, etc. To remove all the images from the `<project>-images` directory:

    Enter ⇒   **lwarpmk cleanlimages**

5.16   Converting PDF or EPS images to SVG

HTML cannot display PDF or EPS images, so any external PDF graphics images must be converted to SVG format. *pdftocairo* and *epstopdf* may be used one image at a time, but *lwarpmk* also provides a way to convert PDF or EPS images in bulk:

    Enter ⇒   **lwarpmk epstopdf *.eps**   **(or a list of files)**

    Enter ⇒   **lwarpmk pdftosvg *.pdf**   **(or a list of files)**

Be sure to always provide svg files for HTML output.

5.17   Creating HTML from an incomplete compile

During testing it may be useful to finish the HTML conversion even when the document had errors and did not compile successfully. To attempt an HTML conversion of an incomplete document:

    Enter ⇒   **lwarpmk pdftohtml [-p project]**

5.18   Processing multiple projects in the same directory

⚠   xr, xr-hyper, xcite

It is possible to have several projects in the same directory. *lwarpmk* has an optional parameter which is the document to compile.

To create each project:

    Enter ⇒   **pdflatex project_a**

Enter ⇒ **pdflatex project_b**

Each project is given its own configuration file:

project_a.lwarpmkconf, project_b.lwarpmkconf

To compile each project with lwarkmk:

Enter ⇒ **lwarpmk print -p project_a**

Enter ⇒ **lwarpmk print -p project_b**

Enter ⇒ **lwarpmk html -p project_a**

Enter ⇒ **lwarpmk html -p project_b**

To generate each project's images:

Enter ⇒ **lwarpmk limages -p project_a**

Enter ⇒ **lwarpmk limages -p project_b**

To clean each project's images:

Enter ⇒ **lwarpmk cleanlimages -p project_a**

Enter ⇒ **lwarpmk cleanlimages -p project_b**

To clean each project's auxiliary files:

Enter ⇒ **lwarpmk cleanall -p project_a**

Enter ⇒ **lwarpmk cleanall -p project_b**

If using *bibtex*, for example, the HTML version must also be processed:

Enter ⇒ **bibtex project_a_html**

5.19   Using the *make* utility

lwarpmk has an action which may be useful for integration with the common *make* utility:

lwarpmk pdftohtml [-p project]

make may be used to compile the code to PDF with HTML tags (project_html.pdf), then *lwarpmk* may be used to convert each target to HTML files.

5.20   What next?

How do I do something? See the General Index.

Something do not work! See the Troubleshooting Index or section 13: Troubleshooting.

Package options: See section 31, Package options.

HTML and filename settings: See section 7.6, Customizing the HTML output.

Footnote placement: See section 7.6, Customizing the HTML output.

Title page, indexing, glossaries: See section 8.6, Front and back matter.

Shell escape: See section 7.3, Shell escape.

CSS customization: See section 7.7, Customizing the CSS.

MATHJAX customization: See section 8.7.7, Customizing MATHJAX.

Localization: (languages) — See section 7.1, Localization.

Accessibility: (`alt` and `title` tags) — See section 7.2, Accessibility.

Converting an existing document: See section 6, Converting an existing document.

EPUB conversion: See section 10, EPUB conversion.

Word processor conversion: See section 11, Word-processor conversion.

6   Converting an existing document

To convert an existing document for use with lwarp:

1. Arrange the document in the following order:

    (a) Declare the \documentclass.

    (b) Load text fonts.

    (c) Load inputenc or inputenx, fontenc, or fontspec.

    (d) Load lwarp.

    (e) Load remaining packages.

2. Modify the document:

⚠ math in section names

    (a) If using named HTML files, in section names use parentheses math \(x+y\) instead of dollar math $x+y$. Parentheses math is removed from the file name. (Dollar math works, but it generates complicated filenames.) Or, use a short name for the TOC entry without the math, or use \texorpdfstring from the hyperref package:

    ```
    \section[Simplified name]{Name with \(1+2=3\) math}
    \section{Some math \texorpdfstring{$1+2=3$}{three}}
    ```

⚠ scale

    (b) Avoid using the \includegraphics scale option. Change:

    ```
    \includegraphics[scale=<xx>]{ . . . }
    ```
    to:
    ```
    \includegraphics[width=<yy>\linewidth]{ . . . }
    ```

⚠ tabular

    (c) Possible changes to tabular environments include: ⋆ columns, multirow, longtable, supertabular, xtab, bigdelim. See section 8.10.1.

⚠ package options

    (d) If using braces in package options, such as with caption, see section 8.1.

    (e) Possible option clashes with memoir. See section 8.13.

⚠ indexes

    (f) If using indexes, see section 8.6.16.

    (g) If using many indexes, glossaries, .aux files, etc., see section 8.6.16 regarding morewrites. If morewrites is already used, be sure to add the setup with allocate=10.

    (h) Other changes as per Special cases and limitations, section 8.

3. Convert any PDF images to SVG. See section 8.8.

4. Manually compile the print version with *latex*, *pdflatex*, *lualatex*, or *xelatex*.

5. lwarpmk print to finish the print version.

6. lwarpmk html to create the HTML version.

7. lwarpmk limages to create the SVG images of any SVG math, lateximage, TikZ, etc.

Need help?

See the General Index for "how-to", and the Troubleshooting Index if something doesn't work. A Troubleshooting section is also available. The Index of Objects contains automated entries for each package, macro, environment, counter, boolean, and other objects; individually and also sorted by category.

Table 5: Localization settings

Object names: LaTeX provides redefinable names for various objects, and lwarp adds a few more. Use \renewcommand to change these.

> **\abstractname:** This macro is honored by lwarp.
>
> **\linkhomename:** Displayed by the link to the homepage.
>
> **\linkpreviousname:** Displayed by the link to the previous page
>
> **\linknextname:** Displayed by the link to the next page.
>
> **\sidetocname:** Displayed at the head of the sideTOC.

HTML settings: See table 8 and section 7.6 for details.

> **\HTMLLanguage:** The language to declare for each web page.
>
> **\ImageAltText,   \MathImageAltText,   \PackageDiagramAltText, \AltTextOpen, \AltTextClose:** The defaults used for HTML alt text for images. See section 7.2.
>
> **\CSSFilename:** The name of the CSS file to use.
>
> **\MathJaxFilename:** The name of the MATHJAX script to use.

Package options:

> **ImagesName and ImagesDirectory:** These options control the filenames used by lwarp when it automatically generates images. See table 7 and section 7.5.
>
> **xindyStyle, xindyLanguage, xindyCodepage:** When using *xindy*, these options may be set according to local use. See section 8.6.22.
>
> **pdftotextEnc:** To adjust the encoding of *pdftotext*.

7   Additional details

7.1   Localization

Regional localization is supported by lwarp via the package options and macros shown in table 5.

7.2   Accessibility

lwarp provides several methods for improving access to the document using tools such as text-only browsers, copy / paste, text-to-speech readers, or Braille readers. lwarp can use the HTML alt text attribute for images, as describe below. lwarp can also use the HTML title attribute, which usually generates a pop-up text. lwarp can add this to a reference or hyperlink. lwarp also uses standard HTML5 elements which are pre-assigned ARIA roles for increased accessibility, and lwarp assigns the math role for SVG math images, and the note role for footnotes, end notes, margin paragraphs and notes, etc. MATHJAX also has provisions for improved accessibility as well. See table 6.

Table 6: Accessibiltiy settings

\ImageAltText: The default HTML alt text for \includegraphics and
   lateximages. Set with \renewcommand.

\includegraphics alt key: For \includegraphics, lwarp adds the alt key/
   value. For example:

   \includegraphics[alt={Some text.}]{filename}

svg math: For simple svg math, lwarp places the LaTeX math expression in the
   alt text, so that the LaTeX expression may be copied and pasted to another
   document as plain text.

\MathImageAltText: For complicated svg math, such as enclosed in
   \InlineMathOther/\InlineMathNormal,   or   \DisplayMathOther/
   \DisplayMathNormal, the HTML alt text will be set to \MathImageAltText.
   Set with \renewcommand.

MathJax: For MathJax, the accessibility tools provided by MathJax are enabled
   by default by lwarp's MathJax scripts.

\PackageDiagramAltText: Various   packages   create   diagrams   which
   lwarp converts into svg images.   These are given alt text set to
   \PackageDiagramAltText. Set with \renewcommand.

\ThisAltText: The HTML alt text of the next image may be set with:

   \ThisAltText{Custom text about the image.}
   <SVG math, Tikz, picture, etc.>

   The next single image will be generated with the given text, and the following
   images will revert to back to their defaults.

   \ThisAltText may also be used to assign an HTML title to the next refer-
   ence or hyperlink.

   \ThisAltText{Custom text about the link.}
   Text ... \ref{label_name} ... text.

   See section 7.6.

\AltTextOpen and \AltTextClose: By default, HTML alt text is enclosed by
   parentheses.   This may be changed by redefining \AltTextOpen and
   \AltTextClose. Set with \renewcommand.

7.3   Shell escape

-\/-shell-escape (*Opt*)   Some documents require the use of an external program, which is allowed when using the `--shell-escape` command-line option. When the document is first compiled manually, and also whenever the print version is recompiled, lwarp detects and remembers whether shell escape is enabled. If so, it will also be enabled when the document is recompiled with *lwarpmk*.

7.4   Font and UTF-8 support

⚠ type 3 bitmapped fonts   lwarp uses *pdftotext* to convert PDF output into UTF-8-encoded text. This process requires that UTF-8 information be embedded in the PDF file, which may prevent the use of older "type 3" bit-mapped fonts, and of older packages such as ae. The lwarp option pdftotextEnc may be useful in some situations. See section 7.5.

vector fonts   While using older versions of DVI *latex* or PDF *pdflatex*, if no font-related package is
Computer Modern   specified then the default COMPUTER MODERN font is used, which may be a "type
⚠ *pdflatex*   3" bit-mapped font which may not convert well to plain text. A "type 1" vector
⚠ DVI *latex*   font is required.

cm-super (*Pkg*)   To use the updated cm-super's type 1 fonts instead of Computer Modern, install the cm-super font package.

lmodern (*Pkg*)   To use Latin Modern instead, add

    \usepackage{lmodern}

to the preamble.
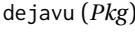
dejavu (*Pkg*)   Another useful option is the Deja Vu series of fonts, which have an increased coverage of language and glyphs:

    \usepackage{dejavu}

font size in math   To adjust the size of the font used in SVG math, see section 8.7.4.

⚠ Missing characters   To avoid "Missing character" warnings and empty or missing characters in HTML and math output, if using a font with an enhanced set of characters also specify a monospace font with similar coverage. lwarp uses the mono font for HTML and math output. Many font packages provide a monospace font automatically.

latex, *pdflatex*, T1, UTF8   While using DVI *latex* or PDF *pdflatex*, lwarp automatically loads fontenc with T1 encoding. fontenc may be loaded with an additional encoding after lwarp. inputenc is automatically loaded with UTF8 encoding if it has not yet been loaded, but may also be specified with another encoding such as latin1. See the next section regarding index encoding.

⚠ *xelatex*, *lualatex*,   X͟ǝLATEX and LuaLATEX users must use the fontspec package. Do NOT use fontenc!
fontspec
Place fontspec or fontenc, xunicode, and other font and UTF-8 related commands after the \documentclass command and before \usepackage{lwarp}.

⚠ package conflicts   In some cases, a package conflict may require that a font package be loaded after lwarp, which should work as well:

1. documentclass{article/book/report} comes first, followed by any of:

2. Font and UTF-8 related commands:

- For X피LATEX or LuaLATEX:

fontspec (*Pkg*)   – fontspec and font choices

ligatures   lwarp sets the following to turn off TEX ligatures during the generation of HTML tags, and turn off common ligatures in regular text, since older browsers may not display them correctly and newer browsers can automatically re-create them.

```
\defaultfontfeatures[\rmfamily]{Ligatures={NoCommon,TeX}}
\defaultfontfeatures[\sffamily]{Ligatures={NoCommon,TeX}}
\defaultfontfeatures[\ttfamily]{Ligatures=NoCommon}
```

- For *pdflatex*:

lmodern (*Pkg*)   (a) \usepackage{lmodern}, or other font-related packages

fontenc (*Pkg*)   (b) \usepackage[T1]{fontenc}

inputenc (*Pkg*)   (c) \usepackage[utf8]{inputenc}, or latin1, etc. Or use inputenx.

inputenx (*Pkg*)   (d) \usepackage{newunicodechar} along with related definitions.

newunicodechar (*Pkg*)   (e) To assist with the PDF-HTML conversion:

glyphtounicode.tex (*file*)
      i. \input glyphtounicode.tex
      ii. \input glyphtounicode-cmr.tex% from the pdfx package
      iii. \pdfgentounicode=1

    (f) Another option to assist with the PDF-HTML conversion, such as

⚠ dotless j   the dotless j (\j):

cmap (*Pkg*)   – \usepackage{cmap}   — *or* —

mmap (*Pkg*)   – \usepackage{mmap}   — *or* —

    – \usepackage[noTeX]{mmap}

textcomp (*Pkg*)   (g) \usepackage{textcomp}

3. \usepackage{newtxmath} or other math-related font packages. Many of these load amsmath, which may now be loaded before lwarp.

4. \usepackage{lwarp} (section 7.5) is placed after any of the above, followed by:

⚠ **fontspec** with **monospaced fonts**   5. \setmonofont{TeX Gyre Cursor} or similar may be required if using X피LATEX or LuaLATEX and fontspec along with traditional font packages such as txfonts, newtxtext, etc. This is required to turn off the monospaced font's ligatures with fontspec after loading the traditional font packages. Monospaced output ligatures must be turned off to produce the correct HTML characters.

⚠ JETBRAIN MONO
⚠ HTML corrupted   Any monospace font with built-in ligatures may require these ligatures to be disabled for HTML. In one example, JETBRAIN MONO, it is required to use

```
\setmonofont{JetBrains Mono}[%
. . .
Contextuals=AlternateOff,
]
```

After lwarp is loaded, the ligature may be re-enabled for print mode by using \setmonofont again inside a warpprint environment.

6. … the rest of the preamble and the main document.

⚠ UTF-8 locale   In some cases, an external program may require a UTF-8 "locale". See section 9.9.

7.4.1 Indexes, glossaries, and encoding

lwarp supports *makeindex*, *xindy*, *xindex*, and glossaries, gloss, and nomencl.

See section 8.6.15 for indexing, and section 8.6.13 for the glossaries package.

7.5   lwarp package loading and options

lwarp supports book, report, and article classes, as well as the equivalent Koma-script classes and memoir, and various CJK-related classes and packages.

Load the lwarp package immediately after the font and UTF-8 setup commands.

Package options may be set while loading lwarp, or later with

> \lwarpsetup{⟨key=value, . . . ⟩}

lwarp (*Pkg*) lwarp package options are as follows:

mathsvg (*Opt*) **mathsvg** and **mathjax:** Selects SVG images or MATHJAX for math display. See sec-
mathjax (*Opt*) tion 8.7.
Default: mathsvg

latexmk (*Opt*) **latexmk:** Tells *lwarpmk* to use *latexmk* to recompile the document several times if
Default: false necessary. Otherwise, *lwarpmk* attempts to determing for itself whether to
recompile. See section 7.6.

dvips (*Opt*) **dvips:** Tells *lwarpmk* to use *dvips* and *ps2pdf* to convert DVI output to PDF.
Default: false

dvipdfm (*Opt*) **dvipdfm:** Tells *lwarpmk* to use *dvipdfm* to convert DVI output to PDF.
Default: false

dvipdfmx (*Opt*) **dvipdfmx:** Tells *lwarpmk* to use *dvipdfmx* to convert DVI output to PDF.
Default: false

HomeHTMLFilename (*Opt*) **HomeHTMLFilename:**
Default: \BaseJobname
Filename of the homepage, without the ".html" suffix. Defaults to the
\BaseJobname. A common setting is:

> HomeHTMLFilename=index

causing the homepage to be the file index.html. Underscores are allowed in
filename underscores HomeHTMLFilename and HTMLFilename options, but may need to be escaped
elsewhere, such as when appearing in a list:

> \item [\href{file_name.pdf}{text}] \

See section 7.6.1 for examples of naming and numbering HTML files.

HTMLFilename (*Opt*) **HTMLFilename:** A filename prefix for the rest of the HTML web pages. Useful for
Default: <empty> numbered web pages with a common prefix. May be empty. See section 7.6.1
for examples of naming and numbering HTML files.

ImagesName (*Opt*) **ImagesName:** The prefix for the images automatically generated by lwarp for ob-
Default: image- jects such as SVG math and lateximages.

ImagesDirectory (*Opt*) **ImagesDirectory:** The directory for the images automatically generated by lwarp
Default: \jobname-images for objects such as SVG math and lateximages. By default, these images
will appear in a directory named <jobname>-images, and the images will be
named and numbered image-<nn>.

Table 7: Lwarp package options

Option	Description
mathsvg	Show math using SVG images.
mathjax	Show math using MATHJAX.
latexmk	Use *latexmk* for compiling documents.
dvips	Use *dvips* and *ps2pdf* to convert DVI documents.
dvipdfm	Use *dvipdfm* to convert DVI documents.
dvipdfmx	Use *dvipdfmx* to convert DVI documents.
HomeHTMLFilename	The filename of the home page.
HTMLFilename	A prefix for the filenames of the remaining web pages.
ImagesName	A prefix for the filenames of generated images.
ImagesDirectory	The directory used to hold generated images.
PrintLatexCmd	The shell commands for **lwarpmk print**.
HTMLLatexCmd	The shell commands for **lwarpmk html**.

For indexing (section 8.6.16) and glossaries (section 8.6.13):

makeindex	Use *makeindex* to generate indices.
makeindexStyle	Set a custom style for *makeindex*.
xindy	Use *xindy* to generate indices.
xindyStyle	Set a custom style for *xindy*.
xindyLanguage	The *xindy* language option used for index generation.
xindyCodepage	The *xindy* codepage option used for index generation.
xindex	Use *xindex* to generate indices.
xindexConfig	Set a custom configuration file for *xindex*.
PrintIndexCmd	Shell commands executed by **lwarpmk printindex**.
HTMLIndexCmd	Shell commands executed by **lwarpmk htmlindex**.
LatexmkIndexCmd	Shell commands executed by *latexmk*.
IndexRef	How to format index links.
GlossaryCmd	Shell command executed by **lwarpmk printglossary** and **lwarpmk htmlglossary**.

Seldom necessary:

OSWindows	Force compatibility with MS-WINDOWS.
pdftotextEnc	Set the encoding for *pdftotext*.
lwarpmk	Generate a local copy of lwarpmk.lua.

Used internally by lwarp:

warpprint	Generate print output, and also generate configuration files.
warpHTML	Generate HTML output.
BaseJobname	The \jobname to use. Set to the \jobname of the printed version even while generating HTML.
warpdisable	Disables most of lwarp for testing purposes.

PrintLatexCmd (*Opt*)
Default: <automatic>

PrintLatexCmd: Sets the shell commands executed by **lwarpmk print**. If not specified, will automatically be set according to the detected LATEX engine and the use of --shell-escape.

HTMLLatexCmd (*Opt*)
Default: <automatic>

HTMLLatexCmd: Sets the shell commands executed by **lwarpmk html**. If not specified, will automatically be set according to the detected LATEX engine and the use of --shell-escape.

makeindex (*Opt*)
Default: makeindex

makeindex: Sets PrintIndexCmd, HTMLIndexCmd, and LatexmkImageCmd to use *makeindex* when generating indexes with **lwarpmk printindex**, **lwarpmk htmlindex**, or *latexmk*. If neither makeindex nor xindy is used, makeindex is assumed.

makeindexStyle (*Opt*)
Default: lwarp.ist

makeindexStyle: If you wish to use a custom .ist file for index generation, see section 8.6.21.

xindy (*Opt*)
Default: makeindex

xindy: Sets PrintIndexCmd, HTMLIndexCmd, and LatexmkImageCmd to use *xindy* when generating indexes with **lwarpmk printindex**, **lwarpmk htmlindex**, or *latexmk*.

xindyStyle (*Opt*)
Default: lwarp.xdy

xindyStyle: If you wish to use a custom .xdy file for index generation, see section 8.6.22.

xindyLanguage (*Opt*)
Default: english

xindyLanguage: If using an index or glossary, see section 31.

xindyCodepage (*Opt*)
Default: utf8

xindyCodepage: If using an index, see section 31.

xindex (*Opt*)
Default: makeindex

xindex: Sets PrintIndexCmd, HTMLIndexCmd, and LatexmkImageCmd to use *xindex* when generating indexes with **lwarpmk printindex**, **lwarpmk htmlindex**, or *latexmk*.

xindexConfig (*Opt*)
Default: <empty>

xindexConfig: If you wish to use a custom xindex-*.lua file for index generation, see section 8.6.23.

PrintIndexCmd (*Opt*)
Default: <automatic>

PrintIndexCmd: Sets the shell commands executed by **lwarpmk printindex**. If not specified, will be set by the selection of makeindex or xindy. May be used to specify the creation of multiple indexes. See section 8.6.16.

Examples:

```
makeindex -s lwarp.ist projectname.idx              (makeindex)
xindy -M lwarp.xdy -L english -C utf8 projectname.idx   (xindy)
```

automatic setting

The use of the makeindex or xindy options sets PrintIndexCmd to sensible values for each of those programs while compiling a single index. lwarp's makeindexStyle, xindyStyle, xindyLanguage, and xindyCodepage options will be used if specified.

⚠ xindy

If specifying PrintIndexCmd manually, be sure to assign an *xindy* language and codepage with the -L and -C *xindy* options, as the lwarp xindyLanguage and xindyCodepage options are not used for the PrintIndexCmd option when it is set manually.

This option is stored in the configuration files lwarpmk.conf and *.lwarpmkconf, and is then passed by the **lwarpmk printindex** command to the operating system to compile the print indexes. Since the command string is parsed by TEX, written to a file, read from the file by LuaTEX, and finally passed to the operating system, any attempt at quoting will be problematic. For complicated commands, it would be best to create a shell script, and simply refer to the script with the lwarp PrintIndexCmd option.

HTMLIndexCmd (*Opt*)
Default: <automatic>

HTMLIndexCmd: Sets the shell commands executed by **lwarpmk htmlindex**. If not specified, will be set by the selection of makeindex or xindy. May be used to specify the creation of multiple indexes. See section 8.6.16.

⚠ filenames

Example settings are similar to PrintIndexCmd, but append _html to the filenames:

```
makeindex -s lwarp.ist projectname_html.idx          (makeindex)
xindy -M lwarp.xdy -L english -C utf8 projectname_html.idx
(xindy)
```

automatic setting

The use of the makeindex or xindy options sets HTMLIndexCmd to sensible values for each of those programs while compiling a single index. lwarp's makeindexStyle, xindyStyle, xindyLanguage, and xindyCodepage options will be used if specified.

⚠ xindy

If specifying HTMLIndexCmd manually, be sure to assign an *xindy* language and codepage with the -L and -C *xindy* options, as the lwarp xindyLanguage and xindyCodepage options are not used for the HTMLIndexCmd option when it is set manually.

As with PrintIndexCmd, to generate complicated indexes it may be worthwhile to use a shell script, then refer to that script with HTMLIndexCmd.

LatexmkIndexCmd (*Opt*)
Default: <automatic>

LatexmkIndexCmd: Sets the shell commands executed by *latexmk*. Unlike PrintIndexCmd and HTMLIndexCmd, LatexmkIndexCmd does not include any filenames, which will be provided instead by *latexmk*. See section 8.6.16.

Example settings are similar to PrintIndexCmd, but without a filename:

```
makeindex -s lwarp.ist                                     (makeindex)
xindy -M lwarp.xdy -L english -C utf8                         (xindy)
```

automatic setting

The use of the makeindex or xindy options sets LatexmkIndexCmd to either of the two settings show above. lwarp's makeindexStyle, xindyStyle, xindyLanguage, and xindyCodepage options will be used if specified. Unlike PrintIndexCmd and HTMLIndexCmd, *latexmk* uses either of the single-line settings of LatexmkIndexCmd shown above to compile each of multiple indexes if necessary.

⚠ xindy

If specifying LatexmkIndexCmd manually, be sure to assign an *xindy* language and codepage with the -L and -C *xindy* options, as the lwarp xindyLanguage and xindyCodepage options are not used for the LatexmkIndexCmd option when it is set manually.

IndexRef (*Opt*)
Default: cref

IndexRef: Describes how to display the index entries for HTML output. Possible values are ref, nameref, refnameref, cref, crefnameref, autoref, or a text string such as (link) or (*) for each index entry reference. (Adding parentheses around a single character makes the link larger and easier to click on.) The default is cref, which is available even if the print document does

not use cleveref, as the lwarp package relies on cleveref during HTML output. Option autoref gives the same results as cref.

\ref and \cref to starred or otherwise unknown links will display as (*) instead of ??.

⚠ ??   If using cref (the default), and if a reference appears as ?? with a non-functional link, use cleveref's \crefname to give a name to that type of label.

In general, crefnameref gives the most information, but the index can become quite verbose. Using (*) or similar yields a very compact index.

GlossaryCmd (*Opt*)
Default: **makeglossaries**

GlossaryCmd: Sets the shell command executed by **lwarpmk printglossary** and **lwarpmk htmlglossary**. The print or HTML glossary filename is appended to this command. See section 8.6.13.

OSWindows (*Opt*)

OSWindows: lwarp attempts to automatically sense WINDOWS, but it may be forced with this option. See section 7.9.

pdftotextEnc (*Opt*)
Default: UTF-8

pdftotextEnc: Used to specify the encoding used by *pdftotext* during the PDF-HTML conversion. In most situations, the default is the correct choice.

lwarpmk (*Opt*)

lwarpmk: If you wish to have lwarp generate a local copy of lwarpmk.lua for archival or local-installation purposes, compile the print version with the lwarpmk option set. See section 31.

The following options are used internally by lwarp, and usually are not used in the user's document:

---

warpprint (*Opt*)
warpHTML (*Opt*)

warpprint and **warpHTML:** Usually controlled by *lwarpmk*, and not set in the document. Select the warpprint option to generate print output (default), or the warpHTML option to generate HTML5 output. The default is print output, so the print version may be compiled with the usual *pdflatex*, etc. When lwarp is loaded in print mode, it creates <project>_html.tex, which sets the warpHTML option before calling the user's source code <project>.tex. In this way, <project>.tex can \usepackage{lwarp} without any options to create a printed version, while <project>_html.tex will create an HTML version.

BaseJobname (*Opt*)
Default: \jobname

BaseJobname: Not intended for the user. Used internally by lwarp when creating the *_html.tex file used to compile the HTML version. See section 31.

warpdisable (*Opt*)

warpdisable: Internally disables both warpprint and warpHTML. This disables most of lwarp, which may be useful for testing purposes to see whether lwarp is causing a problem.

7.6   Customizing the HTML output

Table 8 shows several settings may be used to customize the HTML output. Watch
for the correct placement of each!

⚠ Placement!

⚠ Changes!   Note that if changes are made, it is best to first:

1. Clear all the HTML, PDF, and auxiliary files:

   Enter ⇒   **lwarpmk cleanall**

2. Recompile the print version in order to recreate the configuration files for
   lwarpmk:

   Enter ⇒   **lwarpmk print**

3. Finally, recompile the HTML version with the new settings:

   Enter ⇒   **lwarpmk html**

Placed in the preamble before \begin{document}:

\HTMLFirstPageTop
Default: \<empty\>

\HTMLFirstPageTop: {⟨*contents*⟩}   A user-definable custom action ap-
plied to the top of the home page. Useful for logos, etc. \LinkNext
may be used to link to the next web page. Defaults empty. Ignored in
print output.

\HTMLFirstPageBottom
Default: \<empty\>

\HTMLFirstPageBottom: {⟨*contents*⟩}   A user-definable custom action ap-
plied to the bottom of the home page. Useful for logos, etc. \LinkNext
may be used to link to the next web page. Defaults empty. Ignored in
print output.

\linkhomename
Default: Home

\linkhomename: Name of the link to the home page. Paragraphs are allowed.
Redefine with \renewcommand.

\linkpreviousname
Default: Previous

\linkpreviousname: Name of the link to the previous page. Paragraphs are
allowed. Redefine with \renewcommand.

\linknextname
Default: Next

\linknextname: Name of the link to the next page. Paragraphs are allowed.
Redefine with \renewcommand.

tocdepth (*Ctr*)

tocdepth: Sectioning depth of the table of contents. See section 16 for a list
of LATEX stack depths.

SideTOCDepth (*Ctr*)
Default: 1

SideTOCDepth: Sectioning depth of the sideTOC. Defaults to 1, causing the
sideTOC to show sections but not subsections.

sideTOC

Each subpage of the website has its own small table of contents on the
side (the "sideTOC"). Its depth is set by SideTOCDepth. This sideTOC
is only shown if the browser display is wide enough. When using a
narrow web browser window, "responsive web design" is used to show
the sideTOC at the top of the page, as well as a link back to **Home** at
the top and bottom.

It is recommended to set:

   SideTOCDepth = FileDepth

or

Table 8: HTML settings

Macro/Cntr/Bool	Loc*	Description
\linkhomename	P	Name of the link to the homepage.
\linkpreviousname	P	Name of the link to the previous page.
\linknextname	P	Name of the link to the next page.
SideTOCDepth	P	Sectioning depth of the sideTOC.
\sidetocname	P	Name of the sideTOC.
FileDepth	P	Sectioning depth of the file splits.
CombineHigherDepths	P	Combine higher section levels.
FileSectionNames	P	Use section names for file names, else use numbers.
\FilenameLimit	P	Maximum length of the generated filenames.
FootnoteDepth	P	Sectioning depth of footnotes.
\abstractname	P	The name of the abstract.
\ImageAltText	PD	\includegraphics and other images' alt tag.
\ThisAltText {⟨text⟩}	PD	Assigns an alt/title tag for the next image or link.
\MathImageAltText	PD	The SVG math image lateximage alt tag.
\PackageDiagramAltText	PD	The suffix for a package's lateximage alt tags.
\AltTextOpen	PD	Start an HTML alt tag.
\AltTextClose	PD	End an HTML alt tag.
\CSSFilename	PS	The CSS for the following files.
\MathJaxFilename	PS	The MATHJAX script for the following files.
\HTMLLanguage	PS	The HTML lang tag.
\HTMLTitle	PS	The homepage's <title>, overriding \title.
\HTMLTitleBeforeSection	PS	Set subpage <title>s to \HTMLTitle - sectionname
\HTMLTitleAfterSection	PS	Set subpage <title>s to sectionname - \HTMLTitle
\HTMLAuthor	PS	The HTML author meta tag, overriding \author.
\HTMLDescription	PS	The HTML meta description tag.
\HTMLKeywords	PS	The HTML meta keywords tag.
\HTMLMeta	PS	Clear and set the custom meta tag.
\HTMLAddMeta	PS	Add another meta tag.
\HTMLFirstPageTop	P	Heading for the home page.
\HTMLFirstPageBottom	P	Footer for the home page.
\HTMLPageTop	PS	Heading for the other pages.
\HTMLPageBottom	PS	Footer for the other pages.
\HTMLnewcolumntype	D	\newcolumntype for HTML.
\IndexPageSeparator	P	Index page list separator.
\IndexRangeSeparator	P	Index page range separator.
FixSmallCaps	P	Set true if small caps rendered as all caps.
HTMLDebugComments	P	Boolean to generate HTML comments.

* **P:** Preamble, **D:** Anywhere in the document. **S:** Before a section.

SideTOCDepth = FileDepth+1

⚠ inaccessible pages
If SideTOCDepth < FileDepth, web pages will be inaccessible via the sideTOC.

\sidetocname
Default: Contents

\sidetocname: Name of the sideTOC. Paragraphs are allowed. Redefine with \renewcommand.

FileDepth (*Ctr*)
Default: -5

FileDepth: Sectioning depth of file splits. Defaults to -5, causing the entire HTML website to be one single file.

- To place the entire file into one HTML page, use:
  \setcounter{FileDepth}{-5}
- To split the HTML file at \section depth, use:
  \setcounter{FileDepth}{1}

⚠

- To ensure that the HTML pages/files are accessible:
  Place a \tableofcontents somewhere before the first section break (therefore in the "home page"), and set
  tocdepth >= FileDepth

CombineHigherDepths (*bool*)
Default: true

CombineHigherDepths: Combine a higher section with its first lower subsections, down to the FileDepth. Defaults to true. Set to false to simulate the concept of a chapter opening on its own page, for example.

⚠ Inaccesible pages!

The file splits are controlled by the counter FileDepth and the boolean CombineHigherDepths. Setting FileDepth to 0 splits the file at chapters, 1 at sections, etc. CombineHigherDepths controls whether to combine pages at levels higher than the chosen FileDepth, such as in this tutorial where the page which opens the chapter also contains the first section. Be careful to set tocdepth and SideTOCDepth to allow access to each page of the website. Set tocdepth and SideTOCDepth to be greater than or equal to FileDepth.

⚠ Lost in an old page!

When making changes to the file structure, it is possible to end up with the web browser pointing to an old file which is no longer in use. When this occurs, changes to the web site will not appear in the browser, even if reloading the page, because that page is no longer in use. It is best to return to the home page, clean the files (lwarpmk cleanall), change FileDepth and/or CombineHigherDepths, then finally recompile and renavigate to the desired page using the new file structure.

FileSectionNames (*bool*)
Default: true

FileSectionNames: If true, web page filenames are derived from a sanitized version of the section names. If false, web pages are numbered. Either way, the HTMLFilename option is used as a prefix. See section 7.6.1 for examples of naming and numbering HTML files. The user must ensure that filenames are unique after being sanitized. For example, math in the section name is removed before creating the filename, so the rest of the filename must be sufficiently unique to avoid name collisions.

⚠ Unique filename!

\FilenameLimit
Default: 80

\FilenameLimit: The maximum length of the filenames generated by lwarp. ".html" is added to this length. Redefine with \renewcommand.

FootnoteDepth (*Ctr*)
Default: 3

FootnoteDepth: Determines where to place pending footnotes. 3 places footnotes before each break down to the \subsubsection level. 1 places footnotes before each \section break. Any pending footnotes are also placed at the bottom of each page before each file break.

FixSmallCaps (*bool*)
Default: false

FixSmallCaps: Set true if SMALL CAPS are rendering in all caps ("SMALL

CAPS"). May be required for some fonts (erewhon, utopia, fbb, et al.), and packages such as embrac.

HTMLDebugComments (*bool*) Default: false	**HTMLDebugComments:** Set true to generate HTML comments, such as which section or <div> is being opened or closed.
\abstractname Default: Abstract	**\abstractname:** The name of the abstract. This may also be over-written by the babel package. Defaults to "Abstract". Redefine with \renewcommand.
\IndexPageSeparator Default: ", "	**\IndexPageSeparator:** Index page list separator. Adjust to match index style file. If using gindex, this is set automatically to gindex's \indexpagessep.
\IndexRangeSeparator Default: "--"	**\IndexRangeSeparator:** Index page range separator. Adjust to match index style file. If using gindex, this is set automatically to gindex's \indexrangesep.

Placed before \begin{document}, or before any sectioning command which causes a file break:

\CSSFilename Default: lwarp.css	**\CSSFilename:** {⟨*filename.css*⟩} Sets the CSS file to use for the following files. May be changed before each sectioning command which would cause a file split.

The CSS styles of the web pages are set by the \CSSFilename command. If \CSSFilename is not used, a default plain style is used to mimic printed LATEX output. lwarp_sagebrush.css is a semi-fancy colored style as shown in this tutorial. Change it to lwarp_formal.css for a more formal look, or comment out the \CSSFilename command to see the default. \CSSFilename may be used before each file break to set the CSS for individual pages of the website.

\MathJaxFilename Default: lwarp_mathjax.txt	**\MathJaxFilename:** {⟨*filename*⟩} Sets the MATHJAX script file to use for the following files. May be changed before each sectioning command which would cause a file split.

The MATHJAX script file is copied into the head of each HTML file. This may be used to point to a local repository, add extensions, or change the script somewhere in the middle of the document. \MathJaxFilename may be used before each file break to set the script file for individual pages of the website.

\HTMLLanguage Default: en-US	**\HTMLLanguage:** {⟨*langauge*⟩} The HTML file's HTML lang meta tag. Defaults to en-US.
\HTMLTitle Default: \thetitle	**\HTMLTitle:** {⟨*title*⟩} Overrides \title for the HTML header's meta title. Defaults to \thetitle, which is set by \title, or empty otherwise. Unlike the author, \thetitle is set by \title even if not using the titling package.
\HTMLTitleBeforeSection Default: \HTMLTitleBeforeSection	**\HTMLTitleBeforeSection:** Sets subpage <title> tags to show the website title followed by the section name.
\HTMLTitleAfterSection	**\HTMLTitleAfterSection:** Sets subpage <title> tags to show the section name followed by the website title.
custom <title>	To customize subpage <title>s, redefine \theHTMLTitleSection, which defaults to:

```
\def\theHTMLTitleSection{%
  \theHTMLTitle\theHTMLTitleSeparator\theHTMLSection%
}
```

\HTMLAuthor Default: \theauthor	**\HTMLAuthor:** {⟨*author*⟩} The HTML header's meta author. Defaults to \theauthor, which is set by \author if using the titling package, but is empty otherwise. There are several ways to represent the author and affiliations, especially if using the authblk package, most of which do not result in a sensible \theauthor, so \HTMLAuthor is useful to create a list of authors without their affiliations.
\HTMLDescription Default: <empty>	**\HTMLDescription:** {⟨*description*⟩} Sets the HTML description tag for the following files. May be changed before each sectioning command which would cause a file split.
\HTMLKeywords Default: <empty>	**\HTMLKeywords:** {⟨*keywords*⟩} Sets the HTML keywords tag for the following files. May be changed before each sectioning command which would cause a file split.
\HTMLMeta Default: <empty>	**\HTMLMeta:** {⟨*name*⟩} {⟨*contents*⟩} Clears then sets a new user-definable custom meta tag used for the following pages. Replaces any prior custom meta tags previously set by \HTMLMeta and \HTMLAddMeta.
\HTMLAddMeta Default: <empty>	**\HTMLAddMeta:** {⟨*name*⟩} {⟨*contents*⟩} Add to the user-definable custom meta tags for the following pages. May be used more than once to add multiple tags. Use \HTMLMeta to empty and start over with a new tag.
\HTMLPageTop Default: <empty>	**\HTMLPageTop:** {⟨*contents*⟩} A user-definable custom action applied to the top of pages other than the home page. Useful for logos, etc. Defaults empty. \LinkHome may be used to place a link back to the homepage, as well as \LinkPrevious and \LinkNext. Ignored in print output.
\HTMLPageBottom Default: <empty>	**\HTMLPageBottom:** {⟨*contents*⟩} A user-definable custom action applied to the bottom of pages other than the home page. Useful for authors, copyright notices, contact information, etc. Defaults empty. \LinkHome may be used to place a link back to the homepage, as well as \LinkPrevious and \LinkNext. Ignored in print output.
\LinkHome	**\LinkHome:** Creates a link to the home page. Usually used in \HTMLPageTop and related.
\LinkPrevious	**\LinkPrevious:** Creates a link to the previous HTML page, unless already at the home page. Usually used in \HTMLPageTop and related.
\LinkNext	**\LinkNext:** Creates a link to the next HTML page, unless already at the end. Usually used in \HTMLPageTop and related.

Placed in the home page before the first sectioning command which causes a file break:

\tableofcontents ⚠ TOC on the homepage!	**\tableofcontents:** Used to place a table of contents on the home page. This command must be used before the first file split, so that a way is available to navigate to other files from the homepage. Links to each chapter/section are provided, as selected by tocdepth.

Placed in the document wherever necessary:

\ImageAltText Default: image	**\ImageAltText:** Redefine with \renewcommand. \includegraphics and other images are assigned an HTML alt tag according to \ImageAltText along with \AltTextOpen and \AltTextClose. This text is visible in the

browser if images are not loaded, and appears when the text is copied and pasted. The default is "image", and it may be changed according to the document's language. This may be set in the preamble, or changed as necessary inside the document, where it will affect the following \includegraphics and other images.

\ThisAltText   **\ThisAltText:** {⟨*text*⟩} \ThisAltText can be used to assign an HTML alt text attribute to the next image generated by a lateximage, picture, tikzpicture, or any other similar environment which generates an image, or the next SVG math expression. This tag is cleared after use. The tag is also cleared after each MATHJAX expression, in case the user changes between SVG math and MATHJAX.

    \ThisAltText also may be used to add an HTML title to a reference or hyperlink, such as a \ref, \cref, \href, \url, \hyperref, or \hyperlink. In each case, the alternative text is cleared after use.

\MathImageAltText
Default: math image   **\MathImageAltText:** Redefine with \renewcommand. When creating an SVG math image, its HTML alt tag may be set to the math expression, which may be hashed for image reuse. In the case of \ensuremath or after \inlinemathother, where the contents require a unique image for each instance of the same expression, the alt tag is set to \MathImageAltText, along with \AltTextOpen and \AltTextClose, and the image is not reused.

    This alt expression is visible in the browser if images are not loaded, and appears when the text is copied and pasted. The default is "math image", and it may be changed according to the document's language. This may be set in the preamble, or changed as necessary inside the document, where it will affect the following SVG math images.

\PackageDiagramAltText
Default: diagram   **\PackageDiagramAltText:** Redefine with \renewcommand. For many packages, the output is placed inside a lateximage with an HTML alt tag set to the package name followed by \PackageDiagramAltText. For example:

      (-xy- diagram)

    This expression is visible in the browser if images are not loaded, and appears when the text is copied and pasted. The default is "diagram", and may it be changed according to the document's language. This may be set in the preamble, or changed as necessary inside the document, where it will affect the following package diagrams.

\AltTextOpen
Default: (**\AltTextOpen:** Redefine with \renewcommand.

\AltTextClose
Default:)   **\AltTextClose:** Redefine with \renewcommand. HTML alt text is enclosed by the macros \AltTextOpen and \AltTextClose, which default to an opening and closing parenthesis.

\HTMLnewcolumntype   **\HTMLnewcolumntype:** \newcolumntype may not always work with lwarp for HTML output, since it often involves TEX boxes and fills. To provide a simplified column type for HTML, add \HTMLnewcolumntype in addition.

warpprint (*env.*)   **warpprint:** An environment which is only used while generating print output. Place inside anything which does not apply to HTML and which may cause problems with lwarp. If lwarp knows about and emulates or supports a package then its related macros, lengths, counters, etc. probably won't have to be placed inside a warpprint environment, but

unknown packages may cause problems which may be isolated from lwarp using this environment.

⚠   *Do not place anything else on the same line as* \end{warpprint}. *Also do not nest* warpprint *inside itself.*

warpHTML (*env.*)   **warpHTML:** An environment which is only included while generating HTML output. This is useful for website logos and other items which have no purpose in printed output.

⚠   *Do not place anything else on the same line as* \end{warpHTML}. *Also do not nest* warpHTML *inside itself.*

\warpprintonly   **\warpprintonly:** {⟨*contents*⟩}   A macro version of the warpprint environment.

\warpHTMLonly   **\warpHTMLonly:** {⟨*contents*⟩}   A macro version of the warpHTML environment.

7.6.1   Example HTML file naming

Examples of ways to name or number HTML files:

Numbered HTML nodes:

Example: Homepage index.html, and node-1, node-2. [13]

```
\usepackage[
    HomeHTMLFilename=index,
    HTMLFilename={node-}
]{lwarp}
\boolfalse{FileSectionNames}
```

Named HTML sections, no prefix:

Example: index.html, and About.html, Products.html

```
\usepackage[
    HomeHTMLFilename=index,
    HTMLFilename={}
]{lwarp}
\booltrue{FileSectionNames}
```

Named HTML sections, with prefix:

Example: Homepage mywebsite.html, and additional pages such as mywebsite-About.html, mywebsite-Products, etc.

```
\usepackage[
    HomeHTMLFilename=mywebsite,
    HTMLFilename={mywebsite-}
]{lwarp}
\booltrue{FileSectionNames}
```

---

[13]See \SetHTMLFileNumber to number in groups by chapter, for example.

7.7   Customizing the css

\CSSFilename   {⟨*filename*⟩}
Default: lwarp.css

\CSSFilename may be used to choose which .css file is used to display each page
of the web site. Use \CSSFilename before \begin{document} to assign the style of
the home page. If different parts of the website should have different styles, call
\CSSFilename again before each section heading which creates a new file. This
may be changed numerous times throughout the file, resulting in different HTML
pages having different css files assigned:

```
...
\CSSFilename{myCSS.css}
\chapter{Another Chapter}
...
```

The styles provided by lwarp include:

lwarp.css: A default style if \CSSFilename is not used. This style is comparable to
    a plain LATEX document. To set this style, you may use \CSSFilename{lwarp.css},
    or no \CSSFilename call at all.

lwarp_formal.css: A formal style with a serif fonts and a traditional look.

lwarp_sagebrush.css: A style with muted colors, gradient backgrounds, addi-
    tional borders, and rounded corners.

To see each style in use, change the \CSSFilename entry in the tutorial, lwarpmk html
again, and then reload the tutorial webpage.

Custom css   A customized style may also be created. For each new project a file called
sample_project.css is generated. This may be renamed to <project>.css then
used by assigning \CSSFilename{<project>.css}.

⚠ Rename it!   Note that sample_project.css is overwritten whenever lwarp is loaded in print
mode. It is therefore important to rename the file to something like <project>.css
before using it, so that your own changes are not overwritten.

<project>.css has an entry which loads lwarp.css, and this entry may be
changed to load lwarp_formal.css or lwarp_sagebrush.css if desired. Additional
changes to the css may be made by making entries later in the <project>.css
file.

lwarp.css (*file*)   It is best to make a local project-specific css file such as project.css, containing
project.css (*file*)   only things which are different from lwarp.css. The file project.css should refer
sample_project.css (*file*)   to lwarp.css as follows:

```
/* ( --- Start of project.css --- ) */
/* ( --- A sample project-specific CSS file for lwarp --- ) */

/* Uncomment one of the following: */
@import url("lwarp.css") ;
/* @import url("lwarp_formal.css") ; */
/* @import url("lwarp_sagebrush.css") ; */
```

```
/* Project-specific CSS setting follow here. */
/* . . . */

/* ( --- End of project.css --- ) */
```

Finally use \CSSFilename{<project>.css} in the document to activate the custom css.

7.8   Assigning css classes and styles

html css classes and styles may be assigned to fragments of the document.

BlockClass (*env.*)   [⟨*style*⟩] {⟨*class*⟩}

An entire block of text, including paragraphs, may be assigned a css class and optional css style using the BlockClass environment. The result is placed inside a <div>. A BlockClass may nest other BlockClasses or \InlineClasses.

\InlineClass   (⟨*wp css style*⟩) [⟨*web css style*⟩] {⟨*css class*⟩} {⟨*text*⟩}

A section of text without paragraphs may be assinged a css class and optional css style using the \InlineClass macro. The result is placed inside a <span>. \InlineClass may be nested, but per the html standard it must not contain BlockClass, nor may it contain a paragraph, nor several other objects such as html figures. \InlineClass also accepts a second optional parameter, enclosed inside parentheses, which assigns the style while generating output for a word processor, while ignoring the web style.

Nullified versions of BlockClass and \InlineClass are provided for the print version, so they may be used in the document without placing them inside warpHTML or \warpHTMLonly.

7.9   Selecting the operating system

Unix (*Prog*)   lwarp tries to detect which operating system is being used. Unix / Mac OS / Linux
Mac OS (*Prog*)   is the default (collectively referred to as "Unix" in the configuration files), and
Linux (*Prog*)   MS-Windows is supported as well.

MS-Windows (*Prog*)   If MS-Windows is not correctly detected, use the lwarp option OSWindows.
Windows (*Prog*)

OSWindows (*Opt*)   When detected or specified, the operating-system path separator used by lwarp is modified, and the boolean usingOSWindows is set true. This boolean may be tested by the user for later use.

7.10   Selecting actions for print, html, or MathJax output

The following environments and macros are used to select actions which only apply to either traditional LaTeX print-formatted pdf generation, or to html generation, or to html with MathJax.

For most of built-in LaTeX and many additional packages there is user-level source code support or emulation, so no special handling will be required. For those cases

which lwarp does not handle by itself, the following environments and macros may be used to isolate sections of code for print-only or HTML-only.

These environments are also useful for creating a special version of the titlepage for print and another for HTML.

warpHTML (*env.*) Anything which is to be done only for HTML5 output is surrounded by a warpHTML environment:

```
\begin{warpHTML}
 ... something to be done only during \HTML\ generation
\end{warpHTML}
```

⚠ \end{warpHTML} Do *not* place anything else on the same line as \end{warpHTML}. The exact phrase
⚠ nesting is used to mark the end of the environment. Do not nest warpHTML inside itself. warpMathJax may be used inside warpHTML.

warpprint (*env.*) Anything which is to be done only for print output is surrounded by a warpprint environment:

```
\begin{warpprint}
 ... something to be done only during traditional \PDF\ generation
\end{warpprint}
```

⚠ \end{warpprint} As above, do not place anything else on the line with \end{warpprint}. Do not
⚠ nesting nest warpprint inside itself.

warpall (*env.*) Anything which is to be done for any output may be surrounded by a warpall environment. Doing so is optional.

```
\begin{warpall}
 ... something to be done during print \PDF\ or \HTML\ output
\end{warpall}
```

⚠ \end{warpall} As above, do not place anything else on the line with \end{warpall}. Do not nest
⚠ nesting warpall inside itself.

Macros are also provided for print-only or HTML-only code:

\warpprintonly {⟨*actions*⟩}

Performs the given actions only when print output is being generated.

\warpHTMLonly {⟨*actions*⟩}

Performs the given actions only when HTML output is being generated.

warpMathJax (*env.*) Anything which is to be done only while using HTML output with MATHJAX is surrounded by a warpMathJax environment. Usually, this is \CustomizeMathJax, used to add emulation macros. \end{warpMathJax} must appear on its own line.
⚠ \end{warpMathJax} Do not nest warpMathJax inside itself. warpMathJax may be used inside warpHTML.
⚠ nesting

warpsvg (*env.*)

⚠ \end{warpsvg}

⚠ nesting

Anything which is to be done only while using print output or HTML output with SVG math is surrounded by a warpsvg environment. \end{warpsvg} must appear on its own line. Do not nest warpsvg inside itself. warpsvg may be used inside warpHTML.

\LWR@formatted

To define macros or environments which behave differently depending on print or HTML output, see section 38.

7.11 Commands to be placed into the `warpprint` environment

Certain print-related commands should always be placed inside a warpprint environment, or may need other special handling. These are unrelated to HTML output, but are hard to isolate automatically. For example:

- Paragraph formatting: \parindent \parskip

- Manual page positions such as the textpos package, which is emulated but only in a limited way.

- Anything changing the page counter. lwarp requires that the page counter not be adjusted during HTML output.

Some packages require additional setup commands. Where these packages are emulated for HTML, setup commands may work for the emulated HTML output as well as for print output. See the details for each package in this document for more information.

Also see section 13: Troubleshooting.

7.12 Title page

In the preamble, place an additional block of code to set the following:

```
\title{Document Title} % One line only
\author{Author One\affiliation{Affiliation One} \and
    Author Two\affiliation{Affiliation Two} }
\date{Optional date}
```

The title is used in the meta tags in the HTML files, unless overridden by \HTMLTitle, and the rest are used in \maketitle. To use a \subtitle or \published field, see section 71.8.

\maketitle

Use \maketitle just after the \begin{document}, as this will establish the title of the homepage. Optionally, use a titlepage environment instead.

titlepage (*env.*)

The titlepage environment may be used to hold a custom title page. The titlepage will be set in a <div> class titlepage, and \printtitle, etc. may be used inside this environment.

titlingpage (*env.*)

Another form of custom title page, where \maketitle is allowed, and additional information may be included as well.

\title

{⟨*title*⟩}

⚠ HTML corrupted   Avoid newlines in the \title; these will interfere with the file break and css de-
⚠   newlines   tection. Use a \subtitle command instead (section 71.8). The title will appear in
the document \maketitle as a heading <h1>. The HTML meta title tag will also
have this title, unless \HTMLTitle is used to set the meta title to something else
instead.

\author   {⟨*author*⟩}

In \author, \protect may be needed before some formatting commands. In HTML,
the author will appear in a <div> of class author in the \maketitle. If the titling
package is used, the author will also appear in a HTML meta tag, but \HTMLAuthor
may be necessary to create a plain list of names if \author had affiliations added.
\affiliation is a new addition to lwarp.

\date   {⟨*date*⟩}

\date works as expected. In HTML, this will appear in a <div> class titledate.

\thanks   {⟨*text*⟩}

\thanks are allowed in the titlepage fields, and will be rendered as HTML notes at
the bottom of the title page.

7.13   HTML page meta descriptions

\HTMLDescription   {⟨*A description of the web page.*⟩}
Default: (none)

Each page of HTML output should have its own HTML meta description, which
limitations   usually shows up in web search results. Usually limited to around 150 characters
in length, and should not include the ASCII double quote character (").

placement   Use \HTMLDescription just before \begin{document} to set the description of the
home page, and also just before each sectioning command such as \chapter or
\section where a new file will be generated, depending on FileDepth. For exam-
ple, if FileDepth is 1, use \HTMLDescription just before each \section command,
and that description will be placed inside the HTML page for that \section. The
same descrition will be used for all following HTML files as well, until reset by a
new \HTMLDescription. It is best to use a unique description for each HTML file.

disabling   To disable the generation of HTML description meta tags, use:

       \HTMLDescription{}

7.14   HTML page meta keywords

\HTMLKeywords   {⟨*Keywords for the web page.*⟩}
Default: (none)

\HTMLKeywords behaves like \HTMLDescription, but adds HTML meta keywords
for the following web pages.

disabling   To disable the generation of HTML keyword meta tags, use:

       \HTMLKeywords{}

7.15   HTML homepage meta title

\HTMLTitle   {⟨*title*⟩}

Default: \HTMLtitle{\thetitle}

Sets the contents of the web page <meta name="title"> element. May be set empty to cancel the meta title tag.

See section 7.6 for \HTMLTitleBeforeSection and \HTMLTitleAfterSection, used to set the title for HTML subpages.

7.16   HTML page meta author

\HTMLAuthor   {⟨*author*⟩}

Default: \HTMLAuthor{\theauthor}

Sets the contents of the web page <meta name="author"> element. May be set empty to cancel the meta author tag.

\author may be used to create a list of authors and their affiliations, in several formats if using authblk, and these may not successfully parse properly into a sensible list for \theauthor. \HTMLAuthor may be used to set the meta tag to a simple list of names.

8   Special cases and limitations

Some commonly-used LaTeX expressions should be modified as follows to allow for a smooth conversion to both HTML and print-formatted outputs.

See the General Index for "how-to", and the Troubleshooting Index if something doesn't work. A Troubleshooting section is also available. The Index of Objects contains automated entries for each package, macro, environment, counter, boolean, and other objects; individually and also sorted by category.

8.1   Things to avoid

In the document, avoid the following:

⚠ options with braces **Package options:** Package options may cause problems with lwarp, especially if they include curley braces.

If selecting options with braces in \usepackage does not work:

    \usepackage[font={it,small}]{caption}% does not work

... try instead selecting the package options before loading lwarp:

    \PassOptionsToPackage{font={it,small}}{caption}

    . . .
    \usepackage{lwarp}
    . . .
    \usepackage{caption}

... or try setting package options after the package has been loaded:

    \usepackage{caption}
    \captionsetup{font={it,small}}

page counter: Do not adjust the page counter. If doing so is required for the print version, place the adjustment inside a warpprint environment.

Custom math environment macros: Do not use expressions such as \beq as a replacement for \begin{equation}.

Custom macros in section, figure, table names: Custom macros which appear in sectioning commands or float captions then appear in the .toc, .lof, and .lot lists, and should be made robust using \newrobustcmd or \robustify from etoolbox, xparse, etc.

When setting FileSectionNames to true to name the HTML files from the section names, the file names are created from sanitized versions of the chapter or section names, but the section names must be plain text or something which expands into plain text. Robust macros will not work at the sectioning level which is used for file names, but a robust macro or other complicated name may be used for the manditory argument of \chapter, \section, etc., if a plain-text version is also included in the optional argument:

    \chapter[Plain Name]{\ARobustMacro{Fancy Name}}

8.1.1   Invalid HTML

Additionally, some objects are valid LaTeX, but invalid HTML. An example is a tabular inside \textbf, since HTML does not allow a table inside a span. lwarp

will create the table, and the browser may support it, but the result is technically invalid.

8.2 Formatting

8.2.1 Text formatting

⚠ **\bfseries, etc.** \textbf, etc. are supported, but \bfseries, etc. work only in some situations.

⚠ **HTML special chars** &, <, and > have special meanings in HTML. If \&, \textless, and \textgreater are used, proper HTML entities will be used, but there may be HTML parsing problems if these special characters occur unescaped in program listings or other verbatim text.

program listings For program listings, the listings package is supported, and its literate option is used to automatically convert &, <, and > to proper HTML entities.

minted sanitizes HTML automatically by its colorizing, which splits the special characters from the rest of the tag.

⚠ **verbatim** The fancyvrb and fvextra packages automatically sanitize HTML entities, but the core LATEX verbatim-related environments do not, nor does the verbatim package, so care must be taken to avoid accidentally including valid HTML code inside these environments. It may be sufficient to add a space on either side of &, <, and >.

⚠ **gobble** fancyvrb does not sanitize HTML when using the gobble option.

8.2.2 Small caps

FixSmallCaps (*bool*) Some fonts, such as erewhon, utopia, or fbb, and some packages such as embrac, copy/paste "SMALL CAPS" as all caps ("SMALL CAPS"), which lwarp then reads as all caps, so the text is printed in all caps. If small caps are being rendered as all caps, set:

> \booltrue{FixSmallCaps}

⚠ **CJK fonts** Some CJK fonts may not work if FixSmallCaps is set true.

8.2.3 Horizontal and vertical space and rules

\hspace \hspace is converted to an inline HTML span of the given width, except that 0 width is ignored, a width of .16667em is converted to an HTML thin breakable space (U+2009), and a \fill is converted to a \qquad.

\vspace \vspace is ignored for HTML.

˜
\, ˜ and \, are converted to HTML entities.

\kern \kern and \hskip are entered into the HTML PDF output as-is, then interpreted by
\hskip *pdftotext*, and thus usually appear as a single space.

\rule \rule is converted to an HTML rule of the same dimensions, of the currently

selected text color.

\hrule
\vrule

Both \hrule and \vrule are ignored for HTML. To create a horizontal dividing rule across the page, use \hrulefill in its own paragraph.

\hrulefill

\hrulefill usually creates a one-inch rule, similar to a "fill in the blank". If it is used at the start of a new paragraph, it creates a <div> with a thin horizontal border across the page, as would often be done with \hrule.

8.2.4 Text alignment

Use the environments center, flushright, flushleft instead of the macros \centering, \raggedright, \raggedleft.

⚠ figure & table alignment

\centering, etc. are honored in a figure or table if they are the first command inside the float:

```
\begin{table*}
\centering
\caption{A Table}
. . .
```

8.2.5 Accents

Native LaTeX accents such as \" will work, but many more kinds of accents are available when using Unicode-aware XƎLaTeX and LuaLaTeX. If using accents in section names which will become file names, it is recommended to use the LaTeX accents such as \" and \v instead of Unicode accents. The LaTeX accents will have the accents stripped when creating the filenames, whereas the Unicode accents will appear in the file names, which may cause issues with some operating systems.

8.2.6 textcomp package

textcomp (*Pkg*)

Some textcomp symbols do not have Unicode equivalents, and thus are not supported.

⚠ missing symbols

Many textcomp symbols are not supported by many system / browser fonts. In the css try referencing fonts which are more complete, but expect to see gaps in coverage.

8.2.7 Superscripts and other non-math uses of math mode

Use ^{x} instead of ^{x}

8.2.8 Empty \item followed by a new line of text or a nested list:

lists

Use a trailing backslash: \item[label] \

8.2.9   relsize package

relsize (*Pkg*)   For HTML, only the inline macros are supported: \textlarger, \textsmaller, and \textscale. Each becomes an inline span of a modified font-size.

\relsize, \larger, \smaller, and \relscale are ignored.

While creating SVG math for HTML, the original definitions are temporarilty restored, and so should work as expected.

⚠ not small   The HTML browser's setting for minumum font size may limit how small the output will be displayed.

8.3   Boxes and minipages

8.3.1   Marginpars

\marginpar   [⟨*left*⟩] {⟨*right*⟩}   \marginpar may contains paragraphs, but in order to remain inline with the surrounding text lwarp nullifies block-related macros inside the \marginpar. Paragraph breaks are converted to <br /> tags.

\marginparBlock   [⟨*left*⟩] {⟨*right*⟩}   To include block-related macros, use \marginparBlock, which takes the same arguments but creates a <div> instead of a <span>. A line break will occur in the text where the \marginBlock occurs.

8.3.2   Save Boxes

⚠ HTML corrupted

⚠ boxes   TEX boxes are placed inline and do not allow line breaks, so boxes with long contents may overflow the line during HTML conversion. lwarp uses methods which help avoid this problem.

⚠ minipage, \parbox   \savebox and related do not (yet) support minipage or \parbox.

8.3.3   Minipages

⚠ inline   A line of text with an inline minipage or \parbox will have the minipage or \parbox placed onto its own line, because a paragraph is a block element and cannot be made inline-block.

placement   minipages and \parboxes will be placed side-by-side in HTML unless you place a \newline between them.

side-by-side   Side-by-side minipages may be separated by \quad, \qquad, \enskip, \hspace, \hfill, or a \rule. When inside a center environment, the result is similar in print and HTML. Paragraph tags are suppressed between side-by-side minipages and these spacing commands, but not at the start or end of the paragraph.

⚠ minipage in a span   There is limited support for minipages inside an HTML <span>. An HTML <div> cannot appear inside a <span>. While in a <span>, minipages, and \parboxes, and any enclosed lists have limited HTML tags, resulting in an "inline" format, without markup except for HTML breaks. Use \newline or \par for an HTML break.

⚠ minipage size When using minipage, \parbox, and fminipage, a virtual 6×9 inch text area is used for \linewidth, \textwidth, and \textheight, both for sizing the minipage, and also for its contents.

if width is \linewidth If a minipage or \parbox is assigned a width of exactly \linewidth, in HTML it is automatically given no HTML width, thus allowed to fill the line as needed, similar to how it appears in print output.

full-width if HTML A new macro \minipagefullwidth requests that, during HTML output, the next single minipage or \parbox be generated without an HTML width attribute, allowing it to be the full width of the display rather than the declared print-output width. This may be useful where the printed version's width makes no sense in HTML.

⚠ tabular, multicols Inside a tabular or multicols environment, where the width depends on the browser window, \minipagefullwidth is effectively used by default for every
\UseMinipageWidths minipage or \parbox inside the environment. \UseMinipageWidths may be used
\IgnoreMinipageWidths to tell lwarp to honor the specified widths of all following minipages and \parboxes until the end of the local scope, and \IgnoreMinipageWidths may be used to tell lwarp to ignore the specified widths.

⚠ multicol Inside a multicols, \linewidth is divided by the specified number of columns.

⚠ text alignment Nested minipages adopt their parent's text alignment in HTML, whereas in regular LATEX PDF output they do not. Use a flushleft or similar environment in the child minipage to force a text alignment.

8.3.4 Side-by-side minipages

Place side-by-side minipages inside a center environment, with horizontal space between them, such as \quad, \qquad, \hspace, or \hfill. The result is similar in print and HTML. Do not use space commands at the start or end of the line.

8.3.5 Framed minipages and other environments

\fbox can only be used around inline <span> items during HTML output, but HTML cannot place a block element such as a <div> for a minipage or a list inside of a <span>. Several options are provided for framing an object, depending on which kind of object and which packages are loaded:

\fbox For a framed object, options include:

\fboxBlock
fminipage (*env.*) **To remove the frame in HTML output:** Place the \fbox command and its closing brace inside warpprint environments. This will nullify the frame for HTML output.

To frame the contents inline with some formatting losses in HTML: This is the
For inline text: default action of \fbox when enclosing a minipage. During HTML output, \fbox nullifies the HTML tags for minipage, \parbox, and lists. The contents are included as inline text inside the \fbox's <span> of class framebox. For lists, line breaks are converted to HTML breaks. The result is a plain-text inline version of the contents, framed inline with the surrounding text, but lacking any extra HTML markup.

To frame the contents on their own line with improved formatting in HTML: A
For inline minipage and lists: new command \fboxBlock is included, intended to be a direct replacement

for \fbox for cases where the \fbox surrounds a minipage, table, or list. For print output, this behaves as \fbox. For HTML output, the contents are placed inside an HTML <div> with the class framed, resulting in the contents being placed on their own line with a frame surrounding them. The contents preserve their HTML formatting, so lists and minipages look nicer, and valid HTML is created for a tabular. While an \fbox containing a tabular is valid LaTeX code, the result in HTML is problematic since a table is a <div> not a <span>, so use \fboxBlock around a tabular, or else place the tabular inside a minipage, or use fminipage, described next. Also see below regarding the "Misplaced alignment tab character &." error.

To create a framed minipage in both print and HTML: A new environment fminipage is included. For print output, this is identical to minipage, except that it is also framed. For HTML output, this forms a <div> of class framed, the contents preserve their HTML formatting, and valid HTML is created for a tabular. Also see section 91 for a new environment fcolorminipage. Also see below regarding the "Misplaced alignment tab character &." error.

For display tabular, minipages, and lists:

colored boxes and frames: **To create colored frames and boxes:** See section 681 for xcolor's \colorbox and \fcolorbox, and lwarp's additional \colorboxBlock and \fcolorboxBlock.

To frame tables or verbatim environments: Place the contents inside a fminipage, or perhaps a \fboxBlock for a tabular. Also, if using \fboxblock with tabular, you will have to use \StartDefiningTabulars before the start of the macro which uses \fboxBlock and the tabular, and \StopDefiningTabulars afterwards. Also see the lwarp documentation for the fancybox package.

⚠ *Misplaced alignment tab character &*

To frame equations: See section 264 for the fancybox package.

For fancy framed minipages: See packages boxedminipage, shadow, fancybox, framed, mdframed.

Custom environments: Use a custom environment to create a sidebar, containing a BlockClass environment with custom css formatting, and \warpprintonly{\hrule} command:

```
\begin{BlockClass}{frameminipage}% ignored in print output
  % use \CSS\ to format div class framedminipage
\warpprintonly{\hrule} % only appears in print output
Contents
\warpprintonly{\hrule} % only appears in print output
\end{BlockClass}
```

8.3.6 fancybox package

fancybox (Pkg)
framed equation example

fancybox's documentation has an example FramedEqn environment which combines math, \Sbox, a minipage, and an \fbox. This combination requires that the entire environment be enclosed inside a lateximage, which is done by adding \lateximage at the very start of FramedEqn's beginning code, and \endlateximage at the very end of the ending code. Unfortunately, the HTML alt attribute is not used here.

```
\newenvironmentFramedEqn
{
\lateximage% NEW
\setlength{\fboxsep}{15pt}
...}{...
\[\fbox{\TheSbox}\]
\endlateximage% NEW
}
```

framing alternatives \fbox works with fancybox. Also see lwarp's \fboxBlock macro and fminipage environment for alternatives to \fbox for framing environments.

framed table example The fancybox documentation's example of a framed table using an \fbox containing a tabular does not work with lwarp, but the FramedTable environment does work if \fbox is replaced by \fboxBlock. This method does lose some HTML formatting. A better method is to enclose the table's contents inside a fminipage environment. The caption may be placed either inside or outside the fminipage:

```
\begin{table}
\begin{fminipage}{\linewidth}
\begin{tabular}{lr}
...
\end{tabular}
\end{fminipage}
\end{table}
```

⚠ framed verbatim lwarp does not support the verbatim environment inside a span, box, or fancybox's \Sbox, but a verbatim may be placed inside a fminipage. The fancybox documentation's example FramedVerb may be defined as:

```
\newenvironment{FramedVerb}[1] % width
{
  \VerbatimEnvironment
  \fminipage{#1}
  \beginVerbatim
}{
  \endVerbatim
  \endfminipage
}
```

framed \VerbBox fancybox's \VerbBox may be used inside \fbox.

indented alignment LVerbatim, \LVerbatimInput, and \LUseVerbatim indent with horizontal space which may not line up exactly with what *pdftotext* detects. Some lines may be off slightly in their left edge.

lwarp sanitizes HTML for fancybox verbatims, except for the contents of \VerbBox and any \verb inside.

8.3.7 mdframed package

mdframed (*Pkg*) Most basic functionality is supported, including frame background colors and
support single-border colors and thickness, title and subtitle background colors and borders and thickness, border radius, and shadow. CSS classes are created for mdframed environments and frame titles.

⚠ loading   When used, lwarp loads mdframed in HTML with framemethod=none.

font   For title font, use

        frametitlefont=\textbf,

instead of

        frametitlefont=\bfseries,

where \textbf must appear just before the comma and will receive the following text as its argument (since the text happens to be between braces in the mdframed source). Since lwarp does not support \bfseries and friends, only one font selection may be made at a time.

theoremtitlefont   theoremtitlefont is not supported, since the following text is not in braces in the mdframed source.

ignored options   userdefinedwidth and align are currently ignored.

css classes   Environments created or encapsulated by mdframed are enclosed in a <div> of class mdframed, and also class md<environmentname> for new environments.

Frame titles are placed in a <div> of class |mdframedtitle|. Subtitles are in a <div> of class |mdframedsubtitle|, and likewise for subsubtitles.

8.3.8 tcolorbox package

tcolorbox (*Pkg*)   tcolorbox is emulated for HTML and MATHJAX, and supported as-is inside a lateximage or SVG math.

What has been tested to work (at least partly) includes:

- tcolorbox, \tcbox.
- Title, subtitle.
- Upper, lower parts.
- Colors and title fonts.
- Floating objects.
- Some layered box features.
- Counters, labels, references.
- listings, listingsutf8.
- theorems: Theorems are supported. math, ams equation, etc. are not supported.

⚠ math   Use a tcolorbox with regular math inside it. \tcboxmath and \tcbhighmath are suppored in SVG math, and emulated in MATHJAX.
- Fitting features: \tcboxfit becomes \tcbox in HTML.

⚠ footnotes   - Footnote numbering does not match the printed output.
- MATHJAX emulation is provided for common macros.

⚠ undefined references   If using cleveref, it may be necessary to name theorems such as:

        \crefname{tcb@cnt@mytheo}{my theorem}{my theorems}

8.4   Section names

If using named HTML files, by selecting \booltrue{FileSectionNames}, several steps should be taken to avoid problematic file names.

8.4.1 Formatting in section names

⚠ macros in section names

When using special formatting in the section name, use the optional short form:

```
\section[Simplified name]{Fancy name with formatting}
```

Remember to \protect LATEX commands which appear in section names and TOC captions.

8.4.2 Math in section names

⚠ math in section names

If using named HTML files, in section names use parentheses math \(x+y\) instead of dollar math $x+y$. Parentheses math is removed from the file name. (Dollar math works, but it generates complicated filenames.) Or, use a short name for the TOC entry without the math, or use \texorpdfstring from the hyperref package:

```
\section[Simplified name]{Name with \(1+2=3\) math}
\section{Some math \texorpdfstring{$1+2=3$}{three}}
```

8.4.3 Simplifying file names

The generated filenames may be simplified by using \FilenameSimplify and \FilenameNullify:

\FilenameSimplify {⟨*text*⟩}

To remove common short words from the automatically-generated filenames, replacing each with a single hyphen "-", use \FilenameSimplify:

```
\FilenameSimplify*{-in-}
\FilenameSimplify*{A-}
```

The first example removes the word "in" in the middle of a filename, and the second example removes "A" at the start of the filename. The star forces the arguments to be detokenized, which is required for a plain-text comparison. (The unstarred form is used for a token-sensitive comparison, which is seldom required by the user.) After simplfication, repeated hyphen characters will be further simplified to a single hyphen "-". Finally, single hyphens at the start or end of the filename are removed.

\FilenameNullify {⟨*macros*⟩}

Macro names may appear in the automatically-generated file names. To remove these, create *non-robust* nullified versions of the macros, ensuring that each line ends with a percent character % as shown below. These are placed inside \FilenameNullify, which adds them to the list of macros which are nullfiied during filename generation. Low-level macros such as \begingroup will cause problems when nullfied. Many macros such as \textbf are already nullfied. lwarp also already nullifies built-in symbol and textcomp macros, including if defined by xunicode, but not all xunicode macros. See the definition of \LWR@nullfonts for a complete list.

```
\FilenameNullify{%
  \renewcommand*{\macroname}[1]{#1}%
  \renewcommand*{\anothermacro}{}%
}
```

8.4.4 Preventing duplicate file names

⚠ duplicate filename Section names at levels which result in HTML file splits may be duplcates, but the
resulting file names must be unique. lwarp will generate a warning if a duplicate
section name occurs, then lwarp will append a unique file number to the resulting
file name, thus avoiding file name clashes. These unique file numbers may change
as sections are added or removed. As a result, old and orphaned HTML files may be
left behind. To remove these leftover files, use **lwarpmk cleanall** and recompile.
Also, as file names are adjusted, external links from outside to these files may
be broken. To use fixed file names, use the optional short-form name, or use
\texorpdfstring from hyperref:

```
\section[Unique Name]{Duplicate name}
\section{\texorpdfstring{Duplcate Name}{Unique Name}}
```

8.5 Cross-references

labels Labels with special characters may be a problem. It is best to stick with alpha-
⚠ label characters numeric, hyphen, underscore, and perhaps the colon (if not French).

\nameref \nameref refers to the most recently-used section where the \label was defined.
⚠ empty link If no section has been defined before the \label, the link will be empty. Index
entries also use \nameref and have the same limitation.

8.5.1 Page references

⚠ LᴬTᴇX page numbers The printed page does not translate to the HTML page, so \pageref references are
converted to parentheses containing \pagerefPageFor, which defaults to "see ",
followed by a hyperlink to the appropriate object.

Ex:

```
\ref{sec:name} on page \pageref{sec:name}
```
in HTML becomes:
"Sec. 1.23 on page (**see** sec. 1.23)".

\pagerefPageFor may be redefined to "page for ", empty, etc. See page 513.

8.5.2 cleveref and varioref packages

cleveref (*Pkg*) cleveref and varioref are supported, but printed page numbers do not map to HTML,
varioref (*Pkg*) so a section name or a text phrase are used for \cpageref and \cpagerefrange.
⚠ cleveref page numbers This phrase includes \cpagerefFor, which defaults to "for".

Ex:

```
\cpageref{tab:first,tab:second}
```
in html becomes:
    "pages **for** table 4.1 and **for** table 4.2"

See \cpagerefFor at page 753 to redefine the message which is printed for page number references.

⚠ varioref types   cleveref changes the behavior of varioref in that the reference type is automatically printed if cleveref is loaded. Lwarp requires cleveref, so the HTML version will always automatically print the reference types even if the print mode does not. The simplest way to make them match is to require the cleveref package for the document.

8.5.3 Hyperlinks, hyperref, and url

hyperref (*Pkg*)   lwarp emulates hyperref, including the creation of active hyperlinks, but does not
url (*Pkg*)   require that hyperref be loaded by the document.

⚠ comments between   Do not place a comment with a % character between arguments for \hyperref,
arguments   etc., as it is neutralized for inclusion in HTML URLs.

lwarp can also load url, but url should not be used at the same time as hyperref, since they both define the \url command. lwarp does not (yet) attempt to convert url links into hyperlinks during HTML output, nor does the print version of url create hyperlinks.

⚠ backref   When generating HTML, lwarp's emulation of hyperref does not automatically load backref, so backref must be loaded explicitly.

8.5.4 Footnotes, endnotes, and page notes

lwarp uses native LATEX footnote code, although with its own \box to avoid the LATEX output routine. The usual functions mostly work as-is.

footnote numbering   To have footnote numbers reset each time footnotes are printed:

```
\setcounter{footnoteReset}{1}
```

For bigfoot, manyfoot, or perpage:

```
\MakePerPage{footnoteX}
— or —
\MakeSortedPerPage{footnoteX}
```

The footnotes are reset when they are printed, according to section level as set by FootnoteDepth, which is not necessarily by HTML page. This is recommended for \alph, \Alph, or \fnsymbol footnotes, due to the limited number of symbols which are available.

MATHJAX   Also for MATHJAX, \footnotename is used for a \footnotemark if the actual footnote number is not known. To redefine it, provide it before loading lwarp:

```
\providecommand{\footnotename}{something}
\usepackage{lwarp}
```

Similar for **sidenotes**. For **endnotes**:

```
\def\endnotename{something}% \def allows name to start with
"end"
```

For the **pagenote** package, there is no \pagenotename to define, since there is no \pagenotemark command.

footmisc The **footmisc** `stable` option is emulated by **lwarp**.

⚠ **sectioning commands** When using footnotes in sectioning commands, to generate consistent results between print and HTML, use the **footmisc** package with the `stable` option, provide a short TOC entry, and \protect the \footnote:

```
\usepackage[stable]{footmisc}
. . .
\subsection[Subsection Name]
    {Subsection Name\protect\footnote{A footnote.}}
```

memoir with footmisc
⚠ **memoir** If using **memoir** class, with which **lwarp** preloads **footmisc**, the `stable` option must be declared before **lwarp** is loaded:

```
\PassOptionsToPackage{stable}{footmisc}
\usepackage{lwarp}
. . .
```

Do not use a starred sectioning command. As an alternative, it may be possible to adjust \secnumdepth instead.

fancybox, fancyvrb
⚠ **\VerbatimFootnotes**
⚠ **sectioning or displaymath** If using **fancybox** or **fancyvrb** with \VerbatimFootnotes, and using footnotes in a sectioning command or display math, use \footnotemark and \footnotetext:

```
\subsection[Subsection Name]
    {Subsection Name\protect\footnotemark}
\footnotetext{A footnote with \verb+verbtim+.}
```

and likewise for equations or display math.

pfnote
⚠ **pfnote numbers** While emulating **pfnote**, **lwarp** is not able to reset HTML footnote numbers per page number to match the printed version, as HTML has no concept of page numbers. **lwarp** therefore uses continuous footnote numbering even for **pfnote**.

bigfoot, manyfoot
⚠ **verbatim** Verbatim footnotes are not yet supported.

If using the **bigfoot** package, and possibly also **manyfoot**, problems may occur with counter allocation because **lwarp** uses many counters, and there is a difference in how counters numbered 256 and up are handled in PDF LaTeX. With **bigfoot** this has been known to show up as an error related to one footnote insert being forbidden inside another. Another problem showed up as a input stack error, and which of these problems occurred depended on how many counters were allocated.

As a possible solution, try creating several new counters before defining **bigfoot** or **manyfoot** footnotes, hoping to shift the problematic counter above the 256 threshold. It may instead be necessary to use XƎLATEX or LuaLATEX instead of PDF LATEX.

8.5.5   xr, xr-hyper, and xcite packages

See section 5.18.

8.6   Front and back matter

8.6.1   Custom classes with multiple authors and affiliations

Some classes allow multiple authors and affiliations. Often it is possible to emulate these using a standard class along with authblk:

```
%\documentclass{customclass} % for print document
\documentclass{article} % for html document

\usepackage{lwarp}
\begin{warpHTML}
\usepackage{authblk}
\let\affiliation\affil % maybe required
\end{warpHTML}
```

8.6.2   Starred chapters and sections

HTML page and TOC   The following describes \ForceHTMLPage and \ForceHTMLTOC, which may be used for endnotes, glossaries, tocbibind, bibliographies, and the index. See the following sections where applicable. Continue here if interested in the reason for adding these commands to lwarp.

Some packages use \chapter* or \section* to introduce reference material such as notes or lists, often to be placed in the back matter of a book. These starred sections are placed inline instead of on their own HTML pages, and they are not given TOC entries.

lwarp provides a method to cause a starred section to be on its own HTML page, subject to FileDepth, and also a method to cause the starred section to have its own TOC entry during HTML output.

\ForceHTMLPage   To place a starred section on its own HTML page, use \ForceHTMLPage just before the \chapter* or \section*. lwarp will create a new page for the starred sectional unit.

A starred sectional unit does not have a TOC entry unless one is placed manually. The typical method using \phantomsection and \addcontentsline works for inline text but fails when the new starred section is given its own webpage after the TOC entry is created, or when creating an EPUB where the TOC entry will point to the page before the starred section. If the starred section has its own HTML page
⚠ inaccessible HTML page   but no correct TOC entry pointing to that page, the page will be inaccessible unless some other link is created.

\ForceHTMLTOC   To automatically force the HTML version of the document to have a TOC entry for a starred section, use \ForceHTMLTOC just before the \chapter* or \section*, and place \phantomsection and \addcontentsline inside a warpprint environment.

For print output, \ForceHTMLTOC and \ForceHTMLPage have no effect.

8.6.3 abstract package

abstract (*Pkg*)

⚠ missing TOC

If using the number option with file splits, be sure to place the table of contents before the abstract. The number option causes a section break which may cause a file split, which would put a table of contents out of the home page if it is after the abstract.

8.6.4 titling and authblk

titling (*Pkg*)
authblk (*Pkg*)
package support
⚠ load order
\published and \subtitle

lwarp supports the native LaTeX titling commands, and also supports the packages authblk and titling. If both are used, authblk should be loaded before titling.

If using the titling package, additional titlepage fields for \published and \subtitle may be added by using \AddSubtitlePublished in the preamble. See section 71.8.

8.6.5 tocloft package

titles (*Opt*) [tocloft]
tocloft (*Pkg*)
tocloft (*Pkg*)
⚠ tocloft & other packages

If using tocloft with tocbibind, anonchap, fncychap, or other packages which change chapter title formatting, load tocloft with its titles option, which tells tocloft to use standard LaTeX commands to create the titles, allowing other packages to work with it.

8.6.6 appendix package

appendix (*Pkg*)
⚠ incorrect TOC link

During HTML conversion, the option toc without the option page results in a TOC link to whichever section was before the appendices environment. It is recommended to use both toc and also page at the same time.

8.6.7 pagenote package

pagenote (*Pkg*)

pagenote works as-is, but the page option is disabled.

⚠ labels

Note that labels in page notes do not appear as expected, even in the print version.

8.6.8 endnotes package

endnotes (*Pkg*)

If using cleveref, endnotes displays as a link to an endnote, rather than a section. A comma-separated list of end notes does not work with \cref and related. (In print mode, such as list simply displays a link to the section.)

table of contents

To place the endnotes in the TOC, use:

```
\usepackage{endnotes}
\appto\enoteheading{\addcontentsline{toc}{section}{\notesname}}
\renewcommand*{\notesname}{Endnotes} % optional
```

HTML page

To additionally have the endnotes on their own HTML page, if FileDepth allows:

```
\ForceHTMLPage
\theendnotes
```

⚠ **\endnotemark numbering** If using MATHJAX, see section 8.5.4 regarding the use of \endnotemark and \endnotetext.

8.6.9 *BibTeX*

To update the HTML version of the bibliography:

Enter ⇒ **bibtex <filename>_html**

\etalchar Displays a superscript "+" to indicate "and others".

⚠ **Modify *.bib** When enough authors are cited for a source, BIBTEX may use the \etalchar command to display a math superscript with a + character to indicate "and others". Without modification, this will result in an "Improper \prevdepth" error. At present, lwarp requires that \etalchar be replaced by a text superscript. To do so, add to the start of the .bib file the following:

```
@PREAMBLE{"\let\etalchar\relax \newcommand{\etalchar}[1]{\textsuperscript{#1}}"}
```

8.6.10 *biber*

To update the HTML version of the bibliography:

Enter ⇒ **biber <filename>_html**

8.6.11 **xcite package**

See section 5.18.

8.6.12 **gloss package**

gloss (*Pkg*) To process the HTML glossary:

⚠ **compiling**
```
bibtex <projectname>_html.gls
```

8.6.13 **glossaries package**

glossaries (*Pkg*) *lwarpmk* has the commands **lwarpmk printglossary** and **lwarpmk htmlglossary**,
processing glossaries which process the glossaries created by the glossaries package using that package's
GlossaryCmd (*Opt*) *makeglossaries* program.
Default: makeglossaries
printglossary (*Opt*) [lwarpmk] The shell command to execute is set by the lwarp option GlossaryCmd, which
htmlglossary (*Opt*) [lwarpmk] defaults to **makeglossaries**. The print or HTML glossary filename is appended to
this command.

⚠ *makeglossaries* **not** In some situations it may be required to modify the default command, such as to
found add the **perl** command in front:

```
\usepackage[
    GlossaryCmd={perl makeglossaries},
] {lwarp}
```

xindy language   To set the language to use for processing glossaries with *xindy*:

```
\usepackage[
    GlossaryCmd={makeglossaries -L english},
] {lwarp}
```

Other options for *makeglossaries* may be set as well.

placement and TOC options   The glossaries may be placed in a numbered or unnumbered section, given a TOC entry, and placed inline or on their own HTML page:

Numbered section, on its own HTML page:

```
\usepackage[xindy,toc,numberedsection=nolabel]{glossaries}
. . .
\printglossaries
```

Unnumbered section, inline with the current HTML page:

```
\usepackage[xindy,toc]{glossaries}
. . .
\printglossaries
```

Unnumbered section, on its own HTML page:

```
\usepackage[xindy,toc]{glossaries}
. . .
\ForceHTMLPage
\printglossaries
```

⚠ glossary style   The default `style=item` option for **glossaries** conflicts with lwarp, so the style is forced to `index` instead.

⚠ number list   The page number list in the printed form would become \namerefs in HTML, which could become a very long string if many items are referenced. For now, the number list is simply turned off.

print/HTML versions   The print and HTML versions of the glossary differ in their internal page numbers. Separate commands for generating print and HTML glossaries are used, even though the page number is currently ignored.

8.6.14   nomencl package

nomencl (*Pkg*)   To process the HTML nomenclature:

```
makeindex    <project>_html.nlo    -s    nomencl.ist    -o
<project>_html.nls
```

8.6.15   Indexing overview

There are many ways to process indexes for a LaTeX document, including native LaTeX capabilities, a number of packages and classes, the possible availability of shell escape and *latexmk*, and the need to process print and HTML versions. lwarp attempts to provide easy recompilation of indexes along with the rest of the document, but the various indexing options must be set correctly. Numerous examples are given below. Some differ in minor details, so the important parts are highlighted in red, and options are in green.

Once set up properly, the entire document may be recompiled with **lwarpmk print**
and **lwarpmk html**. In some cases, it will also be necessary to compile the in-
dexes with **lwarpmk printindex** and **lwarpmk htmlindex**. A recompile may then
be forced with **lwarpmk print1** and **lwarpmk html1**.

manual processing   The user may continue to process indexes manually or by shell script without
the use of *lwarpmk*, but adjustments will be required to process HTML indexes as
well. In general, *.idx and *.ind files will be accompanied by *_html.idx and
*_html.ind files.

custom index style   If using a custom indexing style file, see sections 8.6.21 to 8.6.23.

link appearance   To control how the index links appear in the HTML output, see the IndexRef option
in section 7.5, page 109.

source code   See section 81 for lwarp's core index and glossary code, section 347 for index,
section 578 for splitidx, section 345 for imakeidx, section 635 for tocbibind, and
section 702.17 for memoir's indexing patches.

8.6.16   Indexing with makeidx, *makeindex*, *xindy*, *xindex*, gindex

lwarpmk processing   The following allow the user to process indexes automatically, or using *lwarpmk*'s
commands:

> Enter ⇒   **lwarpmk printindex**

> Enter ⇒   **lwarpmk htmlindex**

makeindex (*Prog*)   **For a single index using *makeindex*:**

> \usepackage[makeindex,latexmk] {lwarp}

The usual .idx and .ind files will be used, along with the new lwarp.ist
style file. When creating the HTML index, "_html" is automatically appended
to each of the names.

lwarpmk will use *latexmk* if specified, in which case *latexmk* will create the
index automatically. Otherwise, use

> Enter ⇒   **lwarpmk printindex**

> Enter ⇒   **lwarpmk htmlindex**

to compile the indexes.

To use a custom configuration file, see section 8.6.21.

xindy (*Prog*)   **For a single index using *xindy*:**

> \usepackage[
>   xindy,
>   xindyLanguage=english,   <optional>
>   xindyCodepage=utf8,   <optional>
>   latexmk   <optional>
>]{lwarp}

The usual .idx and .ind files will be used, along with the new lwarp.xdy
style file.

lwarpmk will use *latexmk* if specified, in which case *latexmk* will create the
index automatically. Otherwise, use

> Enter ⇒   **lwarpmk printindex**

Enter ⇒ **lwarpmk htmlindex**

to compile the indexes.

To use a custom configuration file, see section 8.6.22.

xindex (*Prog*) **For a single index using *xindex*:**

```
\usepackage[
    xindex,
    latexmk                                    <optional>
]{lwarp}
```

The usual `.idx` and `.ind` files will be used.

lwarpmk will use *latexmk* if specified, in which case *latexmk* will create the index automatically. Otherwise, use

Enter ⇒ **lwarpmk printindex**

Enter ⇒ **lwarpmk htmlindex**

to compile the indexes.

To use a custom configuration file, see section 8.6.23.

gindex (*Pkg*) **For a single index using gindex:**

```
\usepackage[
    makeindex,
    makeindexStyle=gindex.ist,
    . . . or . . .
    makeindexStyle=gindexh.ist,
    latexmk                                    <optional>
]{lwarp}
```

The usual `.idx` and `.ind` files will be used.

lwarpmk will use *latexmk* if specified, in which case *latexmk* will create the index automatically. Otherwise, use

Enter ⇒ **lwarpmk printindex**

Enter ⇒ **lwarpmk htmlindex**

to compile the indexes.

To use a custom configuration file, copy `gindex.ist` to a new file, modify, then specify it with `MakeindexStyle` as above. lwarp will automatically adapt to gindex's `\indexpagessep` and `\indexrangesep` settings.

8.6.17   Indexing with index

index (*Prog*)

lwarp is told how to use *makeindex* using the `PrintIndexCmd` and `HTMLIndexCmd` options. The file `lwarp.ist` is specified, which generates index letter heads for print output and also allows special HTML formatting for HTML output.

For multiple indexes using *makeindex* and index:

(Assuming that the second index has file extensions `.sist` and `.sind`)

```
\usepackage[
  makeindex, latexmk,
  PrintIndexCmd={
    makeindex -s lwarp.ist <projectname>.idx ;
    makeindex -s lwarp.ist
      -o <projectname>.sind <projectname>.sidx
  },
  HTMLIndexCmd={
    makeindex -s lwarp.ist <projectname>_html.idx ;
    makeindex -s lwarp.ist
      -o <projectname>_html.sind <projectname>_html.sidx
  }
]{lwarp}
\usepackage{index}
. . .
\makeindex
\newindex{secondname}{sidx}{sind}{Second Index}
```

⚠ Windows   *For Windows, replace the two ";" characters with "&".*

When creating the HTML index, "_html" is automatically appended to the index filenames.

Use

     Enter ⇒   **lwarpmk printindex**

     Enter ⇒   **lwarpmk htmlindex**

to compile the indexes.

If the latexmk option is selected for lwarp, *latexmk* will compile the document but will *not* compile the indexes. **lwarpmk printindex** and **lwarpmk htmlindex** will still be required.

8.6.18 Indexing with splitidx

splitidx (*Prog*)

lwarp is told how to use *splitindex* using the PrintIndexCmd and HTMLIndexCmd options. The file lwarp.ist is specified, which generates index letter heads for print output and also allows special HTML formatting for HTML output.

If the latexmk option is selected for lwarp, *latexmk* will compile the document but will *not* compile the indexes. **lwarpmk printindex** and **lwarpmk htmlindex** will still be required.

⚠ \thepage   When using \AtWriteToIndex or \AtNextWriteToIndex, the user must not refer to \thepage during HTML output, as the concept of a page number is meaningless. Instead, do

```
\addtocounter{LWR@autoindex}{1}
\LWR@new@label{LWRindex-\arabic{LWR@autoindex}}
```

where the \index-like action occurs, and then refer to \arabic{LWR@autoindex} instead of \thepage where the reference should occur.

See section 702.17 in the lwarp-patch-memoir package for the \@@wrspindexhyp macro as an example.

For multiple indexes using *makeindex* and splitidx:

```
\usepackage[
  makeindex, latexmk,
  PrintIndexCmd={
    splitindex <projectname> -- -s lwarp.ist
  },
  HTMLIndexCmd={
    splitindex <projectname>_html -- -s lwarp.ist
  }
]{lwarp}
\usepackage{splitidx}
. . .
\makeindex
\newindex[Second Index]{secondname}
```

When creating the HTML index, "_html" is automatically appended to each of the names.

Use

  Enter ⇒ **lwarpmk printindex**

  Enter ⇒ **lwarpmk htmlindex**

to compile the indexes.

For multiple indexes using *xindy* and splitidx:

```
\usepackage[
  xindy, latexmk,
  PrintIndexCmd={
    splitindex -m xindy <projectname> -- -M lwarp.xdy
      -L english -C utf8                        <optional>
  },
  HTMLIndexCmd={
    splitindex  -m  xindy  <projectname>_html  --  -M
lwarp.xdy
      -L english -C utf8                        <optional>
  }
]{lwarp}
\usepackage{splitidx}
. . .
\makeindex
\newindex[Second Index]{secondname}
```

When creating the HTML index, "_html" is automatically appended to each of the names.

Use

  Enter ⇒ **lwarpmk printindex**

  Enter ⇒ **lwarpmk htmlindex**

to compile the indexes.

8.6.19 Indexing with imakeidx

imakeidx (*Prog*)

Due to the number of methods which may be used to process multiple indexes, the options for style file and *xindy* language and codepage must be specified in one of several different ways. These are described in detail later in this section, but are summarized here.

If shell escape is used, imakeidx will automatically compile the indexes by itself. Options specifying a custom style file and *xindy* language and codepage must be specified for each \makeindex command using its options= option, which must include lwarp's special lwarp.ist or lwarp.xdy file, or a file based on them. If using a custom indexing style file, see sections 8.6.21 to 8.6.23.

The splitindex option is also available of shell escape is used, in which case the splitidx package and *splitindex* program will also be used.

If shell escape is not possible, *latexmk* may be used to automatically compile the indexes. The style, language, and codepage options are specified with lwarp's makeindexStyle, xindyStyle, xindyLanguage, and xindyCodepage options. These are passed to *latexmk* by *lwarpmk*'s **lwarpmk printindex** and **lwarpmk htmlindex** commands.

Where shell escape and *latexmk* are not possible, *lwarpmk* may be used to manually compile the indexes. lwarp's PrintIndexCmd and HTMLIndexCmd options are used.

For a single or multiple indexes using *makeindex* and imakeidx:

> The index style lwarp.ist is automatically used for HTML output. This file turns on letter headings, so it may be desirable to specify it as an option, in which case it will also be used for print output, which will help match the print and HTML output.
>
> ```
> \usepackage[makeindex,latexmk] {lwarp}
> \usepackage[makeindex]{imakeidx}
> . . .
> \makeindex[options={-s lwarp.ist}]
> \makeindex[name=secondname,options={-s lwarp.ist}]
> ```
>
> imakeidx will automatically compile the indexes. Shell escape is not required while using *makeindex*. latexmk may be specified, and if so it will be used for **lwarpmk print** and **lwarpmk html**, but *imakeidx* will actually create the indexes.

For a single or multiple indexes using *makeindex* and *splitindex* with imakeidx:

> The index style lwarp.ist is automatically used for HTML output. This file turns on letter headings, so it may be desirable to specify it as an option, in which case it will also be used for print output, which will help match the print and HTML output.
>
> ```
> \usepackage[makeindex,latexmk] {lwarp}
> \usepackage[makeindex,splitindex]{imakeidx}
> . . .
> \makeindex[options={-s lwarp.ist}]
> \makeindex[name=secondname,options={-s lwarp.ist}]
> ```

⚠ enable shell escape   Shell escape is required while using *splitindex*. For the first compile, use

  Enter ⇒   **pdflatex `--shell-escape` projectname.tex**

  Enter ⇒   **pdflatex `--enable-write18` projectname.tex   (MiKTeX)**

or similar with *xelatex* or *lualatex*. lwarp will remember that shell escape was used.

imakeidx will automatically execute *splitindex*, and will also use *makeindex* to compile the indexes.

latexmk may be specified, and if so it will be used for **lwarpmk print** and **lwarpmk html**, but *imakeidx* will actually create the indexes.

For multiple indexes using *xindy* and imakeidx, using shell escape:

Options may be given to imakeidx's \makeindex command. The style file lwarp.xdy is automatically used for HTML output, and is not necessary for print output since the output will be similar. If language or codepage must be set, they should be specified as options for \makeindex, since imakeidx will process the indexes.

```
\usepackage[xindy,latexmk] {lwarp}
\usepackage[xindy,splitindex]{imakeidx}
...
\makeindex[
  options={ -M lwarp.xdy -L english -c utf8 }
]
\makeindex[
  name=secondname,
  options={ -M lwarp.xdy -L english -c utf8 }
]
```

⚠ enable shell escape   For the first compile, use

  Enter ⇒   **pdflatex `--shell-escape` projectname.tex**

  Enter ⇒   **pdflatex `--enable-write18` projectname.tex   (MiKTeX)**

or similar with *xelatex* or *lualatex*. lwarp will remember that shell escape was used.

imakeidx will automatically execute *splitindex* if selected, and will also use *xindy* to compile the indexes.

If selected, *latexmk* will automatically recompile the entire document as necessary.

For indexes using *xindy* and *imakeidx*, without shell escape, but *with latexmk*:

lwarp's options are used, and are passed to *latexmk*.

```
\usepackage[
  xindy,
  xindyLanguage=english,                              <optional>
  xindyCodepage=utf8,                                 <optional>
  latexmk,
]{lwarp}
\usepackage[xindy]{imakeidx}
. . .
\makeindex
\makeindex[name=secondname]
```

latexmk will create the indexes automatically when **lwarpmk print** and **lwarpmk html** are executed.

For indexes using *xindy* and *imakeidx*, without shell escape, and *without latexmk*:

lwarpmk must be told how to create the indexes:

```
\usepackage[
  xindy,
  PrintIndexCmd={
    xindy -M lwarp.xdy -L english -C utf8
      <projectname>.idx ;
    xindy -M lwarp.xdy -L english -C utf8
      secondname.idx
  },
  HTMLIndexCmd={
    xindy -M lwarp.xdy -L english -C utf8
      <projectname>_html.idx ;
    xindy -M lwarp.xdy -L english -C utf8
      secondname_html.idx
  }
]{lwarp}
\usepackage[xindy]{imakeidx}
. . .
\makeindex
\makeindex[name=secondname]
```

⚠ Windows   *For Windows, replace the two ";" characters with "&".*

<projectname> is the \jobname: if compiling "name.tex", use the filenames name.idx and name_html.idx.

Use

     Enter ⇒   **lwarpmk printindex**

     Enter ⇒   **lwarpmk htmlindex**

to compile the indexes.

For multiple indexes using *xindex* and imakeidx, using shell escape:

xindex, *makeindex*, *imakeidx*, and *splitindex* can all work together:

```
\usepackage[%
  xindex,
  xindexConfig=-imakeidx,
  latexmk
] {lwarp}
\usepackage[makeindex,splitindex]{imakeidx}
. . .
\makeindex[%
  options={ -s lwarp.ist} }
]
\makeindex[
  name=secondname,
  options={ -s lwarp.ist} }
]
```

⚠ enable shell escape   For the first compile, use:

Enter ⟹   **pdflatex --shell-escape projectname.tex**

Enter ⟹   **pdflatex --enable-write18 projectname.tex   (MiKTeX)**

or similar with *xelatex* or *lualatex*. lwarp will remember if shell escape was used.

xindex will use *imakeidx*, and *imakeidx* will automatically execute *splitindex* if selected.

If selected, *latexmk* will automatically recompile the entire document as necessary.

8.6.20 Indexes with memoir

For a single index with memoir and *makeindex*:

```
\documentclass{memoir}
\usepackage[makeindex,latexmk]{lwarp}
. . .
\makeindex
```

The usual .idx and .ind files will be used, along with the lwarp.ist style file.

lwarpmk will use *latexmk* if specified, in which case *latexmk* will create the index automatically. Otherwise, use

Enter ⟹   **lwarpmk printindex**

Enter ⟹   **lwarpmk htmlindex**

to compile the indexes.

For multiple indexes with memoir and *makeindex*, using *latexmk*:

lwarp's options are used, and are passed to *latexmk*.

```
\documentclass{memoir}
\usepackage[makeindex,latexmk]{lwarp}
. . .
\makeindex
\makeindex[secondname]
```

lwarpmk will use *latexmk* to create the indexes automatically when the user executes **lwarpmk print** and **lwarpmk html**.

For multiple indexes with memoir and *makeindex*, *without* latexmk:

lwarpmk must be told how to create the indexes:

```
\documentclass{memoir}
\usepackage[
  makeindex,
  PrintIndexCmd={
    makeindex -s lwarp.ist <projectname>.idx ;
    makeindex -s lwarp.ist secondname.idx
  },
  HTMLIndexCmd={
    makeindex -s lwarp.ist <projectname>_html.idx ;
    makeindex -s lwarp.ist secondname_html.idx
  }
]{lwarp}
. . .
\makeindex
\makeindex[secondname]
```

⚠ WINDOWS   *For Windows, replace the two ";" characters with "&".*

<projectname> is the \jobname: if compiling "name.tex", use the filenames
name.idx and name_html.idx.

Use

Enter ⇒   **lwarpmk printindex**

Enter ⇒   **lwarpmk htmlindex**

to compile the indexes.

For a single index with memoir and *xindy*:

```
\documentclass{memoir}
\usepackage[
    xindy,
    xindyLanguage=english,                    <optional>
    xindyCodepage=utf8,                       <optional>
    latexmk                                   <optional>
]{lwarp}
. . .
\xindyindex
\makeindex
```

The usual .idx and .ind files will be used, along with the lwarp.xdy style
file.

lwarpmk will use *latexmk* if specified, in which case *latexmk* will create the
index automatically. Otherwise, use

Enter ⇒   **lwarpmk printindex**

Enter ⇒   **lwarpmk htmlindex**

to compile the indexes.

For multiple indexes with memoir and *xindy*, using latexmk:

lwarp's options are used, and are passed to *latexmk*.

```
\documentclass{memoir}
\usepackage[
    xindy,
    xindyLanguage=english,                          <optional>
    xindyCodepage=utf8,                             <optional>
    latexmk
]{lwarp}
. . .
\xindyindex
\makeindex
\makeindex[secondname]
```

lwarpmk will use *latexmk* to create the indexes automatically.

For multiple indexes with memoir and *xindy*, *without* latexmk:

lwarpmk must be told how to create the indexes:

```
\documentclass{memoir}
\usepackage[
  xindy,
  PrintIndexCmd={
    xindy -M lwarp.xdy -L english -C utf8
      <projectname>.idx ;
    xindy -M lwarp.xdy -L english -C utf8
      secondname.idx
  },
  HTMLIndexCmd={
    xindy -M lwarp.xdy -L english -C utf8
      <projectname>_html.idx ;
    xindy -M lwarp.xdy -L english -C utf8
      secondname_html.idx
  }
]{lwarp}
. . .
\xindyindex
\makeindex
\makeindex[secondname]
```

⚠ WINDOWS   *For Windows, replace the four ";" characters with "&".*

<projectname> is the \jobname: if compiling "name.tex", use the filenames
name.idx and name_html.idx.

Use

    Enter ⇒   **lwarpmk printindex**

    Enter ⇒   **lwarpmk htmlindex**

to compile the indexes.

8.6.21   Using a custom *makeindex* style file

makeindex (*Prog*)   When using *makeindex*, *lwarpmk* uses the file lwarp.ist to process the index. This
lwarp.ist (*file*)   file is over-written by lwarp whenever a print version of the document is processed.

To use a custom *makeindex* style file:

1. Copy `lwarp.ist` to a new filename such as `projectname.ist`

2. Make changes to `projectname.ist`. Keep the lines which refer to `\hyperindexref`. These lines creates the hyperlinks for the HTML index. During print output `\hyperindexref` becomes a null function.

3. If changing

    ```
    delim_n -and- delim_r
    ```

    in `projectname.ist`, then in the document preamble redefine

    ```
    \IndexPageSeparator -and- \IndexRangeSeparator
    ```

    to match.

makeindexStyle (*Opt*)   4. In the document source use the `makeindexStyle` option for lwarp:
    ```
    \usepackage[
      . . . other options . . .
      makeindex,
      makeindexStyle=projectname.ist,
    ]{lwarp}
    ```
    Likewise, refer to the custom style file if using `\PrintIndexCmd`, `\HTMLIndexCmd`, or `\LatexmkIndexCmd`.

5. Recompile the print version, which causes lwarp to rewrite the `lwarpmk.conf` configuration file. This tells *lwarpmk* to use the custom `projectname.ist` file instead of `lwarp.ist`.

8.6.22 Using a custom *xindy* style file

xindy (*Prog*)
lwarp.xdy (*file*)
When using *xindy*, *lwarpmk* uses the file `lwarp.xdy` to process the index. This file is over-written by lwarp whenever a print version of the document is processed.

To use a custom *xindy* style file:

1. Copy `lwarp.xdy` to a new filename such as `projectname.xdy`

2. Make changes to `projectname.xdy`.

    Keep the lines which refer to `\hyperindexref`:

    ```
    (define-attributes (("hyperindexref")))
    (markup-locref :open "\hyperindexref{" :close "}")
    ...
    (markup-locref :open "\textit{\hyperindexref{" :close "}}" :attr "textit")
    ```

    These lines create the hyperlinks for the HTML index. During print output `\hyperindexref` becomes a null function.

    To create custom styles, refer to the lines for `\textbf` and `\textit`.

3. If changing any of

    ```
    markup-locref-list :sep
    markup-locclass-list :open
    markup-locclass-list :sep
    markup-crossref-layer-list :sep
    markup-range :sep
    ```

in `projectname.xdy`, then in the document preamble redefine

> `\IndexPageSeparator -and- \IndexRangeSeparator`

to match.

xindyStyle (*Opt*)   4. In the document source use the `xindyStyle` option for lwarp:

```
\usepackage[
  . . . other options . . .
  xindy,
  xindyStyle=projectname.xdy,
]{lwarp}
```

Likewise, refer to the custom style file if using `\PrintIndexCmd`, `\HTMLIndexCmd`, or `\LatexmkIndexCmd`.

5. Recompile the print version, which causes lwarp to rewrite the `lwarpmk.conf` configuration file. This tells *lwarpmk* to use the custom `projectname.xdy` file instead of `lwarp.xdy`.

8.6.23   Using a custom *xindex* style file

xindex (*Prog*)   To use a custom *xindex* style file:

⚠   filename   1. Copy `xindex-cfg.lua` to a new filename such as `xindex-projectname.lua`. The filename must start with `xindex-` and end with `.lua`.

2. Make changes to `xindex-projectname.lua`.

3. If changing

> `itemPageDelimiter -and- rangeSymbol`

in `xindex-projectname.lua`, then in the document preamble redefine

> `\IndexPageSeparator -and- \IndexRangeSeparator`

to match.

xindexConfig (*Opt*)   4. In the document source use the `xindexConfig` option for lwarp:

```
\usepackage[
  . . . other options . . .
  xindex,
  xindexConfig=projectname,   % (without xindex- or .lua)
]{lwarp}
```

Likewise, refer to the custom style file if using `\PrintIndexCmd`, `\HTMLIndexCmd`, or `\LatexmkIndexCmd`.

5. Recompile the print version, which causes lwarp to rewrite the `lwarpmk.conf` configuration file. This tells *lwarpmk* to use the custom `xindex-projectname.lua` file instead of the default `xindex-cfg.lua`.

8.6.24   Additional indexing limitations

xindy with hyperref   ⚠   *xindy* and hyperref may not work well together for print output with "see", "see also", reference ranges, or stylized index references. It may be necessary to turn off hyper-referencing for indexes:

```
\usepackage[hyperindex=false]{hyperref}
```

⚠   empty index   If an HTML index is empty, it may be necessary to add the following before lwarp is loaded:

```
\usepackage{morewrites}
\morewritessetup{allocate=10}
. . .
\usepackage{lwarp}
```

makeindex custom display styles   When using *makeindex*, custom display styles are possible:

```
\begin{warpprint}
\newcommand{\notesstyle}[1]{#1nn}
\end{warpprint}

\begin{warpHTML}
\makeatletter
\newcommand{\notesstyle}[1]{\LWR@doindexentry{#1} notes }
\makeatother
\end{warpHTML}
. . .
A sentence.\index{key|notesstyle}
```

xindy custom display styles   For custom styles with *xindy*, see lwarp.xdy for \textbf and \textit as examples.

8.6.25   Index positions, TOC, tocbibind

placement and TOC options   An index may be placed inline with other HTML text, or on its own HTML page:

makeidx (*Pkg*)   **Inline, with a manual TOC entry:**

A commonly-used method to introduce an index in a LATEX document:
```
\cleardoublepage
\phantomsection
\addcontentsline{toc}{section}{\indexname}% or chapter
\printindex
```

makeidx (*Pkg*)   **On its own HTML page, with a manual TOC entry:**
```
\begin{warpprint}
\cleardoublepage
\phantomsection
\addcontentsline{toc}{section}{\indexname}% or chapter
\end{warpprint}
\ForceHTMLPage
\ForceHTMLTOC
\printindex
```

tocbibind (*Pkg*)   **Inline, with an automatic TOC entry:**

The tocbibind package may be used to automatically place an entry in the TOC.

```
\usepackage[nottoc]{tocbibind}
...
\cleardoublepage
\phantomsection % to fix print-version index link
\printindex
```

tocbibind (*Pkg*) **On its own HTML page, with an automatic TOC entry:**

```
\usepackage[nottoc]{tocbibind}
...
\cleardoublepage
\phantomsection % to fix print-version index link
\ForceHTMLPage
\printindex
```

numindex (*Opt*) [tocbibind]   Use the tocbibind numindex option to generate a numbered index. Without this
numbered index section   option, the index heading has no number.

Other packages, such as imakeidx, may also have options for including the index
in the Table of Contents.

tocloft (*Pkg*)   If using tocloft with tocbibind, anonchap, fncychap, or other packages which
⚠ tocloft & other packages   change chapter title formatting, load tocloft with its titles option, which tells
tocloft to use standard LaTeX commands to create the titles, allowing other pack-
ages to work with it.

8.7 Math

8.7.1 Math in section names

⚠ math in section names   If using named HTML files, in section names use parentheses math \(x+y\) instead
of dollar math $x+y$. Parentheses math is removed from the file name. (Dollar
math works, but it generates complicated filenames.) Or, use a short name for the
TOC entry without the math, or use \texorpdfstring from the hyperref package:

```
\section[Simplified name]{Name with \(1+2=3\) math}
\section{Some math \texorpdfstring{$1+2=3$}{three}}
```

8.7.2 Math in custom environments

⚠ math in environments

To create an environment which places its contents inside math, instead of:

```
\newenvironment{mymathenv}{ \( starting math}{ending math \) }
```

use:

```
\NewDocumentEnvironment{mymathenv}{b}
  {
    \inlinemathother
    \( starting math #1 ending math \)
    \inlinemathnormal
  }
  {}
```

or:

```
\usepackage{environ}
\NewEnviron{mymathenv}{
  \inlinemathother
  \( starting math \BODY ending math \)
  \inlinemathnormal
}
```

For display math, use \[, \], \displaymathother, and \displaymathnormal.

8.7.3 Rendering tradeoffs

Math rendering   Math may be rendered as svg graphics or using the MathJax JavaScript display engine.

svg files   Rendering math as images creates a new svg file for each expression, except that an MD5 hash is used to combine identical duplicates of the same inline math expression into a single file, which must be converted to svg only once. Display math is still handled as individual files, since it may contain labels or references which are likely to change.

svg inline   The svg images are currently stored separately, but they could be encoded inline directly into the html document. This may reduce the number of files and potentially speed loading the images, but slows the display of the rest of the document before the images are loaded.

png files   Others LaTeX-to-html converters have used png files, sometimes pre-scaled for print resolution but displayed on-screen at a scaled down size. This allows high-quality print output at the expense of larger files, but svg files are the preferred approach for scalable graphics.

MathML   Conversion to MathML might be a better approach, among other things allowing a more compact representation of math than svg drawings. Problems with MathML include limited browser support and some issues with the fine control of the appearance of the result. Also see section 10 regarding epub output with MathJax.

8.7.4 svg option

svg math option   For svg math, math is rendered as usual by LaTeX into the initial pdf file using the current font[14], then is captured from the pdf and converted to svg graphics via a number of utility programs. The svg format is a scalable-vector web format, so math may be typeset by LaTeX with its fine control and precision, then displayed or printed at any size, depending on (sometimes broken) browser support. An html alt attribute carries the LaTeX code which generated the math, allowing copy/paste of the LaTeX math expression into other documents.

svg image font size   For svg display math and the lateximage environment, the size of the math and text used in the svg image may be adjusted by setting \LateximageFontSizeName to a font size name — *without the backslash*, which defaults to:

    \renewcommand{\LateximageFontSizeName}{normalsize}

For inline svg math, font size is instead controlled by \LateximageFontScale, which defaults to:

---

[14]See section 687 regarding fonts and fractions.

```
\newcommand*{\LateximageFontScale}{.75}
```

svg math copy/paste   For svg math, text copy/paste from the HTML `<alt>` tags lists the equation number or tag for single equations, along with the LATEX code for the math expression. For $\mathcal{AMS}$ environments with multiple numbers in the same environment, only the first and last is copy/pasted, as a range. No tags are listed inside a starred $\mathcal{AMS}$ environment, although the `\tag` macro will still appear inside the LATEX math expression.

⚠   svg math size, baseline   svg math sizing and baselines are improved if the **graphics** or **graphicx** package is loaded. An almost-invisible marker is placed at either end of the image to assist in cropping and computing the baseline. A warning is issued at the end of the compile if **graphics** or **graphicx** are not used.

⚠   svg math in TEX boxes   svg math does not work inside TEX boxes, since a `\newpage` is required before and after each image.

8.7.5   MATHJAX option

MATHJAX math option   The MATHJAX (`mathjax.org`) LATEX-math to HTML converter may be used to display
MathJax (*Prog*)   math.

When MATHJAX is enabled, math is rendered twice:

1. As regular LATEX PDF output placed inside an HTML comment, allowing equation numbering and cross referencing to be almost entirely under the control of LATEX, and

2. As detokenized printed LATEX commands placed directly into the HTML output for interpretation by the MATHJAX display scripts. An additional script is used to pre-set the equation number format and value according to the current LATEX values, and the MATHJAX equation numbering system is ignored in favor of the LATEX internal system, seamlessly integrating with the rest of the HTML output, including any math appearing in non-MATHJAX svg output.

8.7.6   MATHJAX rendering options
⚠   fonts

MATHJAX v3 may render using CHTML or SVG. svg display renders italic characters correctly. To select svg rendering, right-click on some math, and select

     Math Settings → Math Renderer → SVG

Wait a moment for the math to rerender.

8.7.7   Customizing MATHJAX

equation numbering   lwarp detects and adjusts MATHJAX equation numbering format for **article** and **book** style equations as well as **amsmath** `\numberwithin` for chapters, sections, and subsections. Custom equation number formats may be set as follows, for example:

```
\renewcommand*{\theequation}{\Alph{section}.\arabic{equation}}
\AtBeginDocument{
  \renewcommand*{\theMathJaxsection}{\Alph{section}.}
}
```

⚠ subequation The amsmath subequations environment is supported, but only with \alpha sube-
quation numbering.

global customizations MATHJAX does not have preexisting support every possible math function. Addi-
tional MATHJAX function definitions may be defined in the preamble. These will be
declared at the start of each HTML page, and thus will have a global effect across
all HTML pages.

Examples:

```
\begin{warpMathJax}
\CustomizeMathJax{
    \newcommand{\expval}[1]{\langle#1\rangle}
    \newcommand{\abs}[1]{\lvert#1\rvert}
}
\CustomizeMathJax{\newcommand{\arcsinh}{\text{arcsinh}}}
\CustomizeMathJax{\newcommand{\arccosh}{\text{arccosh}}}
\CustomizeMathJax{\newcommand{\NN}{\mathbb{N}}}
\end{warpMathJax}
```

⚠ slow compilation To avoid a slowdown in compile speed, use the warpMathJax environment to
prevent its contents from being processed in print or SVG math output. Also, place
each new definition inside its own \CustomizeMathJax. A warning to this effect is
issued if an overly-long definition is attempted.

lwarp already provides MATHJAX customizations for some packages.

siunitx When using siunitx, a similar process may be used to add custom units:

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\myunit}{\mathrm{WXYZ}}}
\CustomizeMathJax{\newcommand{\umyunit}{\mathrm{\micro\myunit}}}
\end{warpMathJax}
```

advanced control For more advanced control over dynamically creating custom definitions, see as
an example the lwarp definition for \DeclarePairedDelimiterX, in section 406,
mathtools.

local customizations For customizations local to the current HTML page only, macros may be defined
as follows:

```
\begin{warpMathJax}
\( \newcommand{\macroname}{. . . } \)
\( \newcommand{\anothername}{. . . } \)
\end{warpMathJax}
```

To maintain compile speed, use the warpMathJax environment, and use a separate
math environment for each definition.

\ifstar For MATHJAX, use \ifstar instead of \@ifstar:

```
\CustomizeMathJax{
  \def\myname{
    \ifstar\starredaction\unstarredaction
    % (Do not place anything after!)
  } }
```

\ifnextchar For MᴀᴛʜJᴀx, use \ifnextchar instead of \@ifnextchar:

> \CustomizeMathJax{\def\myname{\ifnextchar X \found\notfound}}

"X" may be a single ᴀsᴄɪɪ character, or a hex number inside braces, ex:

> \CustomizeMathJax{\def\myname{\ifnextchar{0x7B}\found\notfound}}

Use "(" or "{0x28}" for a left parenthesis, "{0x7B}" for a left brace, "{0x7D}" for a right brace, or "{0x5C}" for a backslash.

8.7.8 MᴀᴛʜJᴀx limitations

MᴀᴛʜJᴀx limitations   Limitations when using MᴀᴛʜJᴀx include:
MathJax (*Prog*)

⚠ \multicolumn, multirow
- MᴀᴛʜJᴀx does not support \multicolumn or multirow. These may be used in text tabulars or sᴠɢ math, but in MᴀᴛʜJᴀx math arrays they are emulated. \multicolumn only fills a single cell, resulting in a short row. \multirow simply prints its text on the first line.

⚠ footnotes
- Footnotes are emulated when used inside a MᴀᴛʜJᴀx expression. For an equation with a single footnote, the correct footnote number is used. For non-equations, \footnotename is used instead, since the actual number cannot be tracked. See section 8.5.4 regarding the use of footnotes with MᴀᴛʜJᴀx.

⚠ references
- Inside a MᴀᴛʜJᴀx expression, references to equations work within the same ʜᴛᴍʟ web page, but do not work when referring to an equation in a different ʜᴛᴍʟ web page. Outside of a MᴀᴛʜJᴀx expression, in the text body, references work as expected.

lateximage
- Math appearing inside a lateximage, and therefore also inside a Ti*k*Z or picture environment, is rendered as sᴠɢ math even if MᴀᴛʜJᴀx is used in the rest of the document.

siunitx
- For siunitx, see siunitx package, section 8.7.15.

physics
- For physics, see physics package, section 8.7.17.

tabbing
- A tabbing environment is emulated using an ʜᴛᴍʟ <pre>. While MᴀᴛʜJᴀx is enabled inside tabbing, the browser may not correctly render the horizontal alignment of the math and text following after on the same line.

\text
- MᴀᴛʜJᴀx includes the *textmacros* extension, which supports various macros which are commonly used inside \text, such as \textbf and text accents. Lwarp supports this extension.

⚠ Unicode
- If using ᴅᴠɪ LᴬTᴇX or ᴘᴅꜰ LᴬTᴇX, unicode input may not appear correctly in MᴀᴛʜJᴀx. Either use XƎLᴬTᴇX or LuaLᴬTᴇX, or replace Unicode special characters such as

  > \text{special character æ}

  with their special macros, such as

  > \text{special character \ae}

⚠ other macros and packages
- Many other math-related macros and packages are not directly supported by MᴀᴛʜJᴀx, including \ensuremath and occasionally-used macros such as \relax. While using MᴀᴛʜJᴀx, lwarp provides emulation for many of these

macros, as well as for footnotes and emulation for dozens of packages (see table 2). In many cases these emulations simply ignore the package in a source-compatible way. Others produce a result which represents the meaning, even if they don't look exact. Look up each package in this document for a description of the limitations of each.

8.7.9 Catcode changes

preamble macros with math

The math shift character $ is not set for HTML output until after the preamble. Macros defined in the preamble which contain $ must be enclosed between \StartDefiningMath and \StopDefiningMath to temporarily change to the HTML meaning of $:

> \StartDefiningMath
> \newcommand{. . . }
> \StopDefiningMath

As an alternative, use \(and \) instead of $, in which case \StartDefiningMath and \StopDefiningMath are not necessary.

If a package defines macros using $, it may be nessary to use \StartDefiningMath and \StopDefiningMath before and after loading the package.

8.7.10 Complicated inline math objects

\inlinemathnormal
\inlinemathother

changing contents
complicated alt tag

MATHJAX limitations

An inline math expression is usually converted to a reusable hashed SVG math image, or a MATHJAX expression. The hash or expression depends on the contents of the math expression. In most cases this math expression is static, such as $x+1$, so the image can be reused for multiples instances of the same expression. In some cases, the math expression includes a counter or other object which may change between uses. Another problem is complicated contents which do not expand well in an alt tag. Yet another problem is math packages which are only partially emulated in MATHJAX. The macro \inlinemathother may be used before a sequence of dynamic or complicated math expressions, and \inlinemathnormal after. Doing so tells lwarp to use unhashed SVG math images for those particular expressins, even if MATHJAX is otherwise in use. See section 46.

8.7.11 Complicated display math objects

\displaymathnormal

By default, or when selecting \displaymathnormal, MATHJAX math display environments print their contents as text into HTML for MATHJAX to interpret, and SVG display math environments render their contents as SVG images and use their contents as the alt tag of HTML output. To do so, the contents are loaded into a macro for reuse. In some cases, such as complicated TikZ pictures, compilation will fail.

\displaymathother
MATHJAX unsupported
complicated alt tag

When selecting \displaymathother, it is assumed that the contents are more complicated than "pure" math. An example is an elaborate TikZ picture, which will not render in MATHJAX and will not make sense as an HTML alt tag. In this mode, MATHJAX is turned off, math display environments become SVG images, even if MATHJAX is selected, and the HTML alt tags become simple messages. The contents are internally processed as an environment instead of a macro argument, so complicated objects such as TikZ pictures are more likely to compile successfully.

8.7.12   Theorems

cref reference format
undefined

If the print version does not use **cleveref**, place all \theoremstyle and \newtheorem declarations in the preamble inside \AtEndPreamble.[15] For some theorems, it may also be required to add inside \AtEndPreamble something such as:

```
\usepackage{etoolbox} % for \ifdef, \AtEndPreamble
\AtEndPreamble{ % if not using cleveref package
  \theoremstyle{definition}
  \newtheorem{dtheorem}{Definition}
  . . .
  \ifdef{\cref}{
    \crefname{Proof}{Proof}{Proofs}
  }{}
}
```

8.7.13   ntheorem package

ntheorem (*Pkg*)

Font control

This conversion is not total. Font control is via css, and the custom LaTeX font settings are ignored.

Equation numbering

ntheorem has a bug with equation numbering in $\mathcal{AMS}$ environments when the option thref is used. lwarp does not share this bug, so equations with \split, etc, are numbered correctly with lwarp's html output, but not with the print output. It is recommended to use **cleveref** instead of ntheorem's thref option.

8.7.14   mathtools package

mathtools (*Pkg*)

equation numbering

showonlyrefs is disabled, as it conflicts with **cleveref**, which is used by lwarp. Equation numbers may not match the print version.

italic correction

mathic is not emulated for html.

MathJax

If using MathJax:

- Recent changes may not yet be updated in the MathJax extension, which is used by lwarp.

- mathtools disallowspaces does not work for MathJax. Protect brackets which are not optional arguments, such as:
  ```
  \begin{gathered}{}
  [p]=1 . . .
  \end{gathered}
  ```

- showonlyrefs does not work in MathJax, and will result in a difference in equation numbering compared to the print version.

- alignat in MathJax requires math mode, but in LaTeX it doesn't. It may be required to use warpHTML and warpprint to isolate a version for each mode.

- \DeclarePairedDelimiter and related must be in the preamble before \begin{document}.

- \MakeAboxedCommand uses \Aboxed for all commands made.

---

[15]lwarp uses **cleveref** for the html conversion, and loads **cleveref** \AtEndPreamble, just before \AtBeginDocument. This is also before the .aux file is read.

8.7.15 siunitx package

siunitx (*Pkg*) siunitx is well supported by lwarp.

Limitations Some general limitations:

fractions Due to *pdftotext* limitations, fraction output is replaced by symbol output for per-mode and quotient-mode.

\cancel is not currently supported for siunitx v3.

Negative values are not automatically colored.

⚠ tabular Tabular S and s columns are rendered as simple c columns, although key settings will be set. If using scientific notation, table-format, table-align-uncertainty, drop-exponent, etc.. use \tablenum for each cell. This is especially required for ⚠ drop-exponent drop-exponent, without which the value will be shown incorrectly.

⚠ table-auto-round table-auto-round is ignored.

Math rendering Math may be rendered in several ways in the same document:

For math mode with svg display: The original siunitx code is used while generating the svg image.

For html text mode: lwarp uses siunitx code patched for html, and simplified units.

For math expressions while using MATHJAX: A limited emulation is used. Most functions work reasonably well, but many options cannot be emulated. The result usually looks fine, and otherwise is enough to get the meaning across.

Custom units siunitx allows customized units:

\DeclareSIUnit {⟨*name*⟩} {⟨*definition*⟩}

\DeclareSIUnit declares a version of the unit for the print version. This is also used when the unit is printed in svg math or a lateximage. It is also used for html if an html-specific version is not defined with \HTMLDeclareSIUnit.

> \DeclareSIUnit\myunit{\ensuremath{\text{m}_y}}

\HTMLDeclareSIUnit {⟨*name*⟩} {⟨*definition*⟩}

⚠ v3 only! Use this after the print unit has been defined. For siunitx v3, \HTMLDeclareSIUnit declares a simplified version of the unit for html, for example if the print-mode unit uses TEX boxes or \ensuremath:

> \HTMLDeclareSIUnit\myunit{\text{m}\textsubscript{\textit{y}}}

It is also possible to provide a custom unit for MATHJAX:

> \CustomizeMathJax{\newcommand{\myunit}{\text{m}_y}}

Predefined units   Most units work as-is with HTML. For the following units, lwarp has already set \HTMLDeclareSIUnit: \celsius, \arcminute, \arcsecond, \elementarycharge, \clight, \bohr, \electronmass, \hartree, \planckbar.

⚠   MathJax

Document modifications required for MATHJAX

⚠   \sisetup

- Place \sisetup in the preamble before \begin{document}. Changes made later may be ignored, especially with MATHJAX. The MATHJAX emulation also ignores most macro options.

⚠   complex numbers

- Complex numbers are displayed as entered, ignoring output-complex-root.

custom units

- Custom units may be added with \CustomizeMathJax. For example, from lwarp-common-mathjax-siunitx:

  \CustomizeMathJax{\newcommand{\hartree}{\mathit{E}_{\mathrm{h}}}}
  \CustomizeMathJax{\newcommand{\angstrom}{\mathrm{\unicode{x212B}}}}}

⚠   unit spacing

- Units work better using ~ between units instead of using periods.

⚠   \square, \cubic

- To square or cube compound units, enclose the following compound units in braces:

  \cubic{\centi\meter}

  Single units do not require braces.

- For \numlist, the argument is printed as text as-is, so use space between semicolons for improved readability.

⚠   Missing $ inserted

- If using parse-numbers = false, also use \num or \qty. siunitx=siunitx>Missing $ inserted.

Also see MATHJAX option, section 8.7.5.

8.7.16   units and nicefrac packages

units (*Pkg*)   units and nicefrac work with lwarp, but MATHJAX does not have an extension for
nicefrac (*Pkg*)   units or nicefrac. These packages do work with lwarp's option svgmath.

8.7.17   physics package

physics (*Pkg*)   physics works as-is for HTML with SVG math.

For MATHJAX, the MATHJAX v3 physics extension is used.

8.8   Graphics

graphics (*Pkg*)   Per table 9, image filenames may be specified either with or without an extension.
graphicx (*Pkg*)   If an extension is given it will be used as-is, for either print or HTML output. If
file extensions   no extension is given, a list of possible extensions is tried, which depends on
whether print or HTML is being generated. This allows a PDF file for print and a
SVG file for HTML, for example. If no extension is given, the automatic search will
⚠   case sensitive   only return lowercase extensions, even if the filename actually has an uppercase

Table 9: \includegraphics and file names

Print image file	HTML image file	Command to use
image.pdf[a]	image.svg[a]	\includegraphics{image}
image.eps[a]	image.svg[a]	\includegraphics{image}
image.jpg	—[b]	\includegraphics{image}
image.png	—[b]	\includegraphics{image}
image.JPG	—[b]	\includegraphics{image.JPG}[c]
image.PNG	—[b]	\includegraphics{image.PNG}[c]
image.jpg	image.gif	\includegraphics{image}

[a]: Must be a lowercase file extension.
[b]: The same file is used for print and HTML.
[c]: The uppercase extension must be specified.

extension, and lwarp cannot get around this problem, so image file extensions must be lowercase to be seen by the HTML browser with lwarp. For example, name the image file image.pdf instead of image.PDF, but refer to it in the source as image, without an extension. For images which may be used as-is with either print or HTML, such as JPG or PNG, you may use a capitalized extension if it is specified in the source, such as image.JPG.

\includegraphics file formats   For \includegraphics with .pdf or .eps files, the user must provide a .pdf or .eps image file for use in print mode, and also a .svg, .png, or .jpg version of the same image for use in HTML.

    \includegraphics{filename} % print:.pdf/.eps HTML:.svg, etc.

For print output, lwarp will automatically choose the .pdf or .eps format if available, or some other format otherwise. For HTML, one of the other formats is used instead.

If a .pdf or .eps image is referred to with its file extension, the extension will be changed to .svg for HTML:

    \includegraphics{filename.pdf} % uses .svg in html
    \includegraphics{filename.eps} % uses .svg in html

pdftocairo (*Prog*)
PDF to SVG   To convert a PDF image to SVG, use the utility *pdftocairo*:

    Enter ⇒ **pdftocairo -svg filename.pdf**

lwarpmk pdftosvg (*Prog*)   For a large number of images, use *lwarpmk*:

    Enter ⇒ **lwarpmk pdftosvg *.pdf**   **(or a list of filenames)**

lwarpmk epstopdf (*Prog*)
epstopdf (*Prog*)
epstopdf package   For EPS images converted to PDF using the package epstopdf, use

    Enter ⇒ **lwarpmk pdftosvg *.PDF**

to convert to SVG images.

DVI LⁱTEX When using DVI *latex*, it is necessary to convert EPS to PDF and then to SVG:

Enter ⇒ **lwarpmk epstopdf *.eps** **(or a list of filenames)**

Enter ⇒ **lwarpmk pdftosvg *.pdf** **(or a list of filenames)**

PNG and JPG For PNG or JPGwhile using *pdflatex*, *lualatex*, or *xelatex*, the same file may be used in both print or HTML versions, and may be used with a file extension, but will also be used without the file extension if it is the only file of its base name.

GIF GIF files may be used for HTML, but another format must also be provided for print output.

file extension priorities If a file extension is not used, for HTML the file extension priorities are: SVG, GIF, PNG, then JPG.

duplicate files A complication occurs if a file of the same name exists elsewhere in the TEX tree, such as a test image from some LⁱTEX package. TEX looks in the local document ⚠ image not displayed directory before considering the directories specified by \graphicspath, but the TEX tree is found as "local", so any file in the tree is found before the directories in \graphicspath. To use such an image, it must be copied to the document's directory to be used for HTML, and furthermore must be in the document's base directory instead of an images subdirectory.

⚠ graphics vs. graphicx If using the older graphics syntax, use both optional arguments for \includegraphics. A single optional parameter is interpreted as the newer graphicx syntax. Note that ⚠ viewport viewports are not supported by lwarp — the entire image will be shown.

units For \includegraphics, avoid px and % units for width and height, or enclose them inside warpHTML environments. For font-proportional image sizes, use ex or em. For fixed-sized images, use cm, mm, in, pt, or pc. Use the keys width=.5\linewidth, or similar for \textwidth or \textheight to give fixed-sized images proportional to a 6 by 9 inch text area. Do not use the scale option, since it is not well supported by HTML browsers.

options \includegraphics accepts width and height, origin, rotate and scale, plus new class and alt keys. (alt has recently been incorportated into graphicx itself.)

HTML class With HTML output, \includegraphics accepts an optional class=xyz keyval combination, and if this is given then the HTML output will include that class for the image. The class is ignored for print output.

HTML alt tags Likewise, the \includegraphics alt key adds an HTML alt tag to an image, and is ignored for print output. If not assigned, each image is given an alt tag according to \ImageAltText.

⚠ scale Avoid using the \includegraphics scale option. Change:

\includegraphics[scale=<xx>]{ . . . }

to:

\includegraphics[width=<yy>\linewidth]{ . . . }

\rotatebox \rotatebox accepts the optional origin key.

⚠ browser support \rotatebox, \scalebox, and \reflectbox depend on modern browser support. The CSS3 standard declares that when an object is transformed the whitespace

which they occupied is preserved, unlike LATEX, so expect some ugly results for
scaling and rotating.

8.8.1 tikz package

tikz (*Pkg*) If using display math with tikzpicture or \tikz, along with matrices with the &
⚠ displaymath and   character, the document must be modified as follows:
matrices

```
\usepackage{tikz}
\tikzset{every picture/.style={ampersand replacement=\&}}
```

and each instance of & in the tikz expression must be replaced with \&.

8.8.2 grffile package

grffile (*Pkg*) grffile is supported as-is. File types known to the browser are displayed, and
unknown file types are given a link. Each PDF image for print mode should be
⚠ matching PDF and SVG   accompanied by an SVG, PNG, or JPG version for HTML.

8.8.3 color package

color (*Pkg*) color is superceded by xcolor, and lwarp requires several of the features of xcolor.
When color is requested, xcolor is loaded as well.

8.8.4 xcolor package

xcolor (*Pkg*) \colorboxBlock and \fcolorboxBlock are provided for increased HTML compati-
\colorboxBlock and   bility, and they are identical to \colorbox and \fcolorbox in print mode. In HTML
\fcolorboxBlock   mode they place their contents into a <div> instead of a <span>. These <div>s are
set to display: inline-block so adjacent \colorboxBlocks appear side-by-side
in HTML, although text is placed before or after each.

Print-mode definitions for \colorboxBlock and \fcolorboxBlock are created by
lwarp's core if xcolor is loaded.

background: none \fcolorbox and \fcolorboxBlock allow a background color of none, in which case
only the frame is drawn, which can be useful for HTML.

color support Color definitions, models, and mixing are fully supported without any changes
required.

colored text and boxes \textcolor, \colorbox, and \fcolorbox are supported.

\color and \pagecolor \color and \pagecolor are ignored. Use CSS or \textcolor where possible.

8.8.5 epstopdf package

epstopdf (*Pkg*) Images with an .eps extension will be converted to .pdf. The HTML output uses
⚠ convert to .svg   the .svg version, so use

Enter ⇒   **lwarpmk pdftosvg <listofPDFfiles>**

to generate .svg versions.

8.8.6   pstricks package

pstricks (*Pkg*)   All pstricks content should be contained inside a pspicture environment.

⚠ use pspicture

8.8.7   pdftricks package

pdftricks (*Pkg*)   The pdftricks image files <jobname>-fig*.pdf must be converted to .svg, or else a

⚠ convert image files   missing file error will occur. The image files must also be converted again whenever they change. To convert the images:

Enter ⇒   **lwarpmk pdftosvg <jobname>-fig*.pdf**

8.8.8   psfrag package

psfrag (*Pkg*)   The psfrags environment is modified to use lateximage to encapsulate the image.

⚠ use psfrags   Always use a psfrags environment to contain any local \psfrag macros and the associated \includegraphics or \epsfig calls. Outside of a psfrags environment, psfrags adjustments will not be seen by lwarp.

⚠   Tip: Use a mono-spaced font for the tags in the EPS file.

8.8.9   pstool package

pstool (*Pkg*)   \graphicspath is ignored, and the file directory must be stated.

⚠ path and filename   The filename must not have a file extension.

Use

Enter ⇒   **lwarpmk html**

followed by

Enter ⇒   **lwarpmk limages**

.

8.8.10   asymptote package

asymptote (*Pkg*)   To compile:

```
pdflatex project.tex
asy project-*.asy
pdflatex project.tex

lwarpmk print
asy project-*.asy
lwarpmk print1
lwarpmk print1

lwarpmk html
asy project_html-*.asy
lwarpmk html1
lwarpmk html1
lwarpmk limages
```

8.8.11   overpic package

overpic (*Pkg*)

⚠ scaling

The macros \overpicfontsize and \overpicfontskip are used during HTML generation. These are sent to \fontsize to adjust the font size for scaling differences between the print and HTML versions of the document. Renew these macros before using the overpic and Overpic environments.

8.8.12   Multimedia packages

multimedia (*Pkg*)
movie15 (*Pkg*)
media9 (*Pkg*)

The packages multimedia, movie15, and media9 are supported.

HTML5 <audio> and <video> objects are created for .mp3 and .mp4 files.

HTML5 <embed> objects are created for http and ftp links.

\href links are created for other media types. (Unfortunately, there is not much overlap between the file types supported for print output and the file types supported by HTML5.)

For media9, a multimedia object is inserted for each addresource=, as well as each flashvars source= and src=. This may result in duplicate objects.

Undesired objects may be nullified by placing them inside \warpprintonly or the warpprint environment.

Each HTML multimedia object includes the poster text, except for <embed> objects. For movie15, the text option is supported to specify the poster text.

The width, height, and totalheight options are supported. The HTML object is scaled according to the display width, correctly compensating for either tall or wide viewports.

Other options are ignored.

media9 \addmediapath is supported. It is assumed that the same path structure will exist for the HTML document.

HTML5 media controls are always specified for each <audio> and <video> object.

media9 slideshows are not supported.

\hyperlinkmovie, \movieref, and \mediabutton are not supported.

3D objects are not supported.

If using a YOUTUBE™ video, use an "embedded" URL with …/embed/… instead of …/v/…

8.9 Tabbing

The tabbing environment works, except that SVG math and lateximages do not yet work inside the environment.

⚠ math in tabbing If math is used inside tabbing, place tabbing inside a lateximage environment, which will render the entire environment as a single SVG image.

8.10 Tabular

8.10.1 tabular environment

Tabular mostly works as expected, but pay special attention to the following, especially if working with environments, macros inside tabulars, multirows, siunitx S columns, or the packages multirow, longtable, supertabular, or xtab.

Defining macros and environments:

⚠ Misplaced alignment tab character &
- When defining environments or macros which include tabular and instances of the & character, it may be necessary to make & active before the environment or macro is defined, then restore & to its default catcode after, using the following commands. These are are ignored in print mode.

      \StartDefiningTabulars
      <define macros or environments using tabular and & here>
      \StopDefiningTabulars

⚠ floatrow This includes before and after defining any macro which used \ttabbox from floatrow.

⚠ tabular inside another environment
- When creating a new environment which contains a tabular environment, lwarp's emulation of the tabular does not automatically resume when the containing enviroment ends, resulting in corrupted HTML rows. To fix this, use \ResumeTabular as follows. This is ignored in print mode.

```
\StartDefiningTabulars   %   (&   is   used   in   a
definition)
\newenvironment{outerenvironment}
{
   \tabular{cc}
   left & right \\
}
{
   \TabularMacro\ResumeTabular
   left & right \\
   \endtabular
}
\StopDefiningTabulars
```

For developers:
- To automate the use of \StartDefiningTabulars and \EndDefiningTabulars, these macros may be embedded inside an HTML environment definition to automatically change the catcode of & before absorbing the arguments. Another environment may be embedded as well.

```
% Does the work after the catcode has been changed:
\newcommand*{\LWR@HTML@subsomename}[2]{%
   . . .
   \otherenvironmentname   [<args>]   {<args>}   %   for
example
}
% Change catcode before absorbing arguments:
\newcommand*{\LWR@HTML@somename{%
   \StartDefiningTabulars
   \LWR@HTML@subsomename
}
% Change catcode again at the end:
\newcommand*{\LWR@HTML@endsomename}{%
   . . .
   \endotherenvironmentname   % for example
   \StopDefiningTabulars
}
% Combine with the existing print definition:
\LWR@formattedenv{somename}
```

Cell contents:

⚠ macro in a table
- Using a custom macro inside a tabular data cell may result in an extra HTML data cell tag, corrupting the HTML table. To avoid this, use \TabularMacro just before the macro. This is ignored in print mode.

```
\TabularMacro\somemacro & more row contents \\
```

Column specifiers:

⚠ math
- Due to the way math is gathered for processing, column specifiers such as >{$}c<{$} do not work with lwarp. Instead, each cell must specify math mode individually.

@ and !
- Only one each of @ and ! is used at each column, and they are used in that order.

\multirow
- In \multirow cells, the print version may have extra instances of <, >, @, and ! cells on the second and later rows in the \multirow which do not appear in the HTML version.

⚠ \newcolumntype
- If \newcolumntype does not work for HTML, add a simplified column type using \HTMLnewcolumntype.

font and alignment

- lwarp detects each of the following, and sets HTML CSS appropriately:
    ```
    >{\centering\arraybackslash}
    >{\raggedright\arraybackslash}
    >{\raggedleft\arraybackslash}
    >{\itshape}
    >{\bfseries}
    >{\bfseries\itshape}
    ```
  These may be used with \newcolumntype, such as:
    ```
    \newcolumntype{P}[1]{>{\centering\arraybackslash}p{#1}}
    ```

Rules:

- Doubled \hlines, \midrules, and vertical rules are supported.

vertical rules

- Vertical rules next to either side of an @ or ! column are displayed on both sides of the column.

width and trim

- Width options are honored. Trim options are converted to rounded top corners. Trim corners are not rounded with @ or ! columns, and full-width rules ignore trim. When given an optional width, each cell is styled to create the custom border. Without an optional width, the entire row is given a class to assign the standard border.

combined rules

- If you wish to use \cmidrule followed by \bottomrule, it may be necessary to use:
    ```
    \cmidrule{2-3} \\[-2ex]
    \bottomrule
    ```
  The optional -2ex is ignored in HTML, but improves the visual formatting in the print output.

⚠ \warpprintonly
⚠ Misplaced \noalign

- For \toprule and \bottomrule, when combined with a warpprint or warpHTML environment, if a "Misplaced \noalign" error occurs, change
    ```
    This & That \endhead
    ```
  to
    ```
    \warpprintonly{This & That \endhead}
    ```
  and likewise with the other \end headings. Keep the \endfirsthead row unchanged, as it is still relevent to HTML output.

Other:

- tabularx ignores the width, but X columns do produce paragraph columns or multicolumns.

longtable headings

- For longtable, place headings and footings which do not apply to HTML inside \warpprintonly{}.

⚠ S columns

- For S columns (from the siunitx package), while producing print output, anything non-numeric must be placed inside { } braces, including commands such as \multirow. While producing HTML output, though, anything placed inside braces is not seen by lwarp's tabular handling algorithm. To resolve this problem, make a copy of the row, with one version for print output, containing the extra braces, and another version for HTML output, without the extra braces, such as:
    ```
    \warpprintonly{1 & 2 & {\multirow{2}{2cm}{Text}} & 3
    \\}
    \warpHTMLonly{1 & 2 & \multirow{2}{2cm}{Text} & 3 \\}
    ```

⚠ tabular inside a <span>

- In LaTeX, a tabular may be placed inside a minipage, but in HTML a <table> may not be inside a <span>. If this situation is detected, a warning is printed instructing the user to isolate the <span> using \warpprintonly or the warpprint environment.

8.10.2 multirow package

vposn
- Note that recent versions of multirow include a new optional vposn argument.

multirow cells
- For multirow, insert \mrowcell into any empty multi-row cells. This will be a null function for the print output, and is a placeholder for parsing the table for HTML output. An error is generated if this is missed.

```
... &   \multirow{2}{.5in}{text}   & ...

... &   \mrowcell                  & ...
```

colored cells
- The multirow documentation regarding colored cells recommends using a negative number of rows. This will not work with lwarp, so \warpprintonly and \warpHTMLonly must be used to make versions for print and HTML.

with \multicolumn
- See section 436.2 for \multicolumrow.

⚠ \multicolumn &
\multirow

lwarp does not support directly combining \multicolumn and \multirow. Use \multicolumnrow instead. To create a 2 column, 3 row cell:

```
\multicolumnrow{2}{c}[c]{3}[0]{1in}[0pt]{Text}
```

The two arguments for \multicolumn come first, followed by the five arguments for \multirow, many of which are optional, followed by the contents.

⚠ skipped cells

As per \multirow, skipped cells to the right of the \multicolumnrow statement are not included in the source code on the same line. On the following

⚠ empty cells

lines, \mcolrowcell must be used for each cell of each column and each row to be skipped. An error is generated if this is missed.

```
... &   \multicolumnrow{2}{c}[c]{3}[0]{1in}[0pt]{Text}   & ...

... &   \mcolrowcell          &          \mcolrowcell   & ...

... &   \mcolrowcell          &          \mcolrowcell   & ...
```

⚠ MathJax
- MATHJAX does not support multirow, so it is emulated to only print its text on the first row. \multirow works as expected in text tabulars or SVG math.

8.10.3 longtable package

longtable (*Pkg*) Use one of either \endhead or \endfirsthead for both print and HTML, and use a \warpprintonly macro to disable the other head phrase, and also the \endfoot and \endfirstfoot phrases. (See section 8.10.4 if using threeparttablex.)

```
\begin{longtable}{ [column specifiers] }
[ . . . ] \endfirsthead   % or \endhead, for print and HTML
\warpprintonly{           % not used in HTML
  [ . . . ] \endhead      % or \endfirsthead
  [ . . . ] \endfoot
  [ <lastfoot macros> ] \endlastfoot
}
. . .  table contents . . .
\warpHTMLonly{
  [ <lastfoot macros> ]   % HTML last footer, without \endfoot
                                            % or \endlastfoot.
}
\end{longtable}
```

⚠ Misplaced \noalign Use the \warpprintonly macro instead of the warpprint environment. Doing so helps avoid "Misplaced \noalign." when using \begin{warpprint}.

⚠ \kill \kill is ignored, place a \kill line inside

    \begin{warpprint} . . . \end{warpprint}

or place it inside \warpprintonly.

⚠ lateximage longtable is not supported inside a lateximage.

8.10.4 threeparttablex package

threeparttablex (*Pkg*) threeparttablex is used with longtable and booktabs as follows:

```
\begin{longtable}{ [column specifiers] }
[ . . . ] \endfirsthead   % or \endhead, for print and HTML
\warpprintonly{          % not used in HTML
  [ . . . ] \endhead     % or \endfirsthead
  [ . . . ] \endfoot
  \bottomrule \insertTableNotes \endlastfoot
}
. . . table contents . . .
\warpHTMLonly{   % HTML last footer
  \bottomrule
  \UseMinipageWidths     % optional
  \insertTableNotes
  \endlastfoot
}
\end{longtable}
```

table width The table notes are created using a \multicolumn. By default the width is not specified to the browser, so long table notes can cause the table to be spread out horizontally. For HTML output, lwarp guesses the width of the table depending on the number of columns, then restricts its guess to a min/max range. To use this guess for the width of the table notes, use \UseMinipageWidths before \insertTableNotes. The width is then specified, and in many cases the result is an improvement in overall table layout.

8.10.5 supertabular and xtab packages

supertabular (*Pkg*) For \tablefirsthead, etc., enclose them as follows:
xtab (*Pkg*)
⚠ Misplaced alignment   \StartDefiningTabulars
tab character &   \tablefirsthead
   . . .
   \StopDefiningTabulars

See section 8.10.1.

⚠ lateximage supertabular and xtab are not supported inside a lateximage.

8.10.6   colortbl package

colortbl (*Pkg*)   Only use \rowcolor and \cellcolor at the start of a row, in that order.

⚠ row/cell color   colortbl ignores the overhang arguments.

colored tables   \rowcolors is supported, except that the optional argument is ignored so far.

8.10.7   ctable package

⚠ Misplaced alignment   Use \StartDefiningTabulars before one or more \ctables, and \StopDefiningTabulars
tab character &   after. These change the meaning of the ampersand & character.

8.10.8   bigdelim package

bigdelim (*Pkg*)   \ldelim and \rdelim use \multirow, so \mrowcell must be used in the proper
⚠ use \mrowcell   number of empty cells in the same column below \ldelim or \rdelim, but not in
cells which are above or below the delimiter:

```
\begin{tabular}{lll}
<empty> & a & b \\
\ldelim{\{}{3}{.25in}[left ] & c & d \\
\mrowcell & e & f \\
\mrowcell & g & h \\
<empty> & i & j \\
\end{tabular}
```

<->	a	b
	c	d
left	e	f
	g	h
<->	i	j

For MATHJAX, limited emulation is provided which merely prints the delimter and
optional text in the first row.

8.11   Floats

8.11.1   Float contents alignment

⚠ figure & table   \centering, etc. are honored in a figure or table if they are the first command
alignment   inside the float:

```
\begin{table*}
\centering
\caption{A Table}
...
```

8.11.2   float, trivfloat, and/or algorithmicx together

float (*Pkg*)   If using \newfloat, trivfloat, and/or algorithmicx together, see section 646.1.
trivfloat (*Pkg*)

algorithmicx (*Pkg*)

⚠ package conflicts   ### 8.11.3   caption and subcaption packages

caption (*Pkg*)   Package options may cause problems with lwarp, especially if they include curley
subcaption (*Pkg*)   braces.

If selecting options with braces in \usepackage does not work:

```
\usepackage[font={it,small}]{caption}% does not work
```

… try instead selecting the package options before loading lwarp:

```
\PassOptionsToPackage{font={it,small}}{caption}
. . .
\usepackage{lwarp}
. . .
\usepackage{caption}
```

… or try setting package options after the package has been loaded:

```
\usepackage{caption}
\captionsetup{font={it,small}}
```

⚠ numbering   To ensure proper float numbering, set caption positions such as:

```
\captionsetup[figure]{position=bottom}
\captionsetup[subfigure]{position=bottom}
\captionsetup[table]{position=top}
\captionsetup[subtable]{position=top}
```

Similarly for longtable. These positions depend on where the user places the
\caption command inside each float.

8.11.4   subfig package

subfig (*Pkg*)

⚠ table numbering   To have correct sub table numbers:

```
\usepackage{caption}
\captionsetup[table]{position=top}
```

⚠ lof/lotdepth   At present, the package options for lofdepth and lotdepth are not working. These
counters must be set separately after the package has been loaded.

⚠ horizontal spacing   In the document source, use \hfill and \hspace* between subfigures to spread
them apart horizontally. The use of other forms of whitespace may cause para-
graph tags to be generated, resulting in subfigures appearing on the following lines
instead of all on a single line.

8.11.5 floatrow package

floatrow (*Pkg*) Use \StartDefiningTabulars and \StopDefiningTabulars before and after defining macros using \ttabbox with a tabular inside. See section 8.10.1.

⚠ Misplaced alignment
tab character &
⚠ subfig package

When combined with the subfig package, while inside a subfloatrow \ffigbox and \ttabbox must have the caption in the first of the two of the mandatory arguments.

⚠ \FBwidth, \FBheight

The emulation of floatrow does not support \FBwidth or \FBheight. These values are pre-set to .3\linewidth and 2in. Possible solutions include:

- Use fixed lengths. lwarp will scale the HTML lengths appropriately.

- Use warpprint and warpHTML environments to select appropriate values for each case.

- Inside a warpHTML environment, manually change \FBwidth or \FBheight before the \ffigbox or \ttabbox. Use \FBwidth or \FBheight normally afterwards; it will be used as expected in print output, and will use your custom-selected value in HTML output. This custom value will be used repeatedly, until it is manually changed to a new value.

8.11.6 keyfloat package

keyfloat (*Pkg*)
⚠ keywrap

If placing a \keyfig[H] inside a keywrap, use an absolute width for \keyfig, instead of lw-proportional widths. (The [H] option forces the use of a minipage, which internally adjusts for a virtual 6-inch wide minipage, which then corrupts the lw option.)

For wrapped figures, overhang and number of lines are ignored.

8.12 KOMA-SCRIPT classes

komascript (*Cls*) Many features are ignored during the HTML conversion. The goal is source-level compatibility.

\captionformat, \figureformat, and \tableformat are not yet emulated.

⚠ Not fully tested! Please send bug reports!

Some features have not yet been tested. Please contact the author with any bug reports.

8.13 MEMOIR class

memoir (*Cls*)
⚠ captions

lwarp uses caption, which causes a warning from memoir. This is normal. Adjust captions via caption, instead of memoir.

⚠ options clash

While emulating memoir, lwarp pre-loads a number of packages (section 702.1). This can cause an options clash when the user's document later loads the same packages with options. To fix this problem, specify the options before loading lwarp:

```
\documentclass{memoir}
. . .
\PassOptionsToPackage{options_list}{package_name}
. . .
\usepackage{lwarp}
. . .
\usepackage{package_name}
```

⚠ version numbers
memoir emulates a number of packages, and declares a version date for each which often does not match the date of the corresponding freestanding package. This can cause warnings about incorrect version numbers. Since lwarp is intended to support the freestanding packages, which are often newer than the date declared by memoir, it is hoped that memoir will update and change its emulated version numbers to match.

\label(bookmark){tag}
\label accepts an optional (bookmark) argument, but this is ignored in HTML.

⚠ comment
The comment environment is from the comment package, and thus requires that the \begin and \end each be on its own line:

```
\begin{comment}
This is a comment.
\end{comment}
```

\newcomment
Comments defined with \newcomment use memoir's defintions, and behave as expected, where the \begin and \end do have to each be on its own line.

⚠ verbatim footnotes
\verbfootnote is not supported.

⚠ \newfootnoteseries
\newfootnoteseries, etc. are not supported.

⚠ page notes
lwarp loads pagenote to perform memoir's pagenote functions, but there are minor differences in \pagenotesubhead and related macros.

page notes with cleveref
To add support for pagenotes with cleveref, add:

```
\crefname{pagenote}{page note}{page notes}
\Crefname{pagenote}{Page note}{Page notes}
```

page note \nameref
Note that for print mode, \nameref print the section name where the page notes are declared in the text, but for HTML it prints the name where the page notes are printed.

⚠ poems
Poem numbering is not supported.

⚠ verbatim
The verbatim environment does not yet support the memoir enhancements. It is currently recommended to load and use fancyvrb instead.

⚠ glossaries
The memoir glossary system is not yet supported by *lwarpmk*. The glossaries package may be used instead, but does require the glossary entries be changed from the memoir syntax to the glossaries syntax.

⚠ framewithtitle, titledframe
The custom frame commands in the memoir manual may be emulated by placing the original defintions in the preamble inside warpprint environments, and then providing an HTML equivalent:

```
\begin{warpHTML}
\newcommand{\FrameTitle}[2]{%
    \textbf{#2}
}
```

```
\newenvironment{framewithtitle}[2][\FrameFirst@Lab\ (cont.)]{%
    \begin{fminipage}{\linewidth}
    \textbf{#2}
    \begin{minipage}{\linewidth}
}
{\end{minipage}\end{fminipage}}

\newcommand{\TitleFrame}[2]{%
    \par
    \textbf{#1}\par
    \fboxBlock{#2}
}

\newenvironment{titledframe}[2][\FrameFirst@Lab\ (cont.)]{%
    \par
    \textbf{#2}
    \begin{fminipage}{\linewidth}
}
{\end{fminipage}}
\end{warpHTML}
```

8.14 International languages

⚠ section and file names   If using *pdflatex* with the setting \booltrue{FileSectionNames}, non-ASCII text in section names can result in corrupted HTML file names. *pdflatex* may be used if setting \boolfalse{FileSectionNames}, in which case HTML file numbers will be generated.

For correct HTML file names, use *xelatex*, *lualatex*, or dedicated document classes / engines.

(As of this writing, this warning is only relevent to the kotex package.)

8.15 Miscellaneous packages

8.15.1 verse and memoir

verse (*Pkg*)   When using verse or memoir, always place a \\ after each line.

memoir (*Cls*)
\attrib   The documentation for the verse and memoir packages suggest defining an \attrib command, which may already exist in current documents, but it will only work for print output. lwarp provides \attribution, which works for both print and HTML output. To combine the two so that \attrib is used for print and \attribution is used for HTML:

```
\begin{warpHTML}
\let\attrib\attribution
\end{warpHTML}
```

\vleftskip (*Len*)   These lengths are used by verse and memoir to control the left margin, and they
\vleftmargini (*Len*)   may already be set by the user for print output. New lengths \HTMLvleftskip and
\HTMLvleftskip (*Len*)   \HTMLleftmargini are provided to control the margins in HTML output. These
\HTMLleftmargini (*Len*)   new lengths may be set by the user before any verse environment, and persist

until they are manually changed again. One reason to change \HTMLleftmargini is if there is a wide \flagverse in use, such as the word "Chorus", in which case the value of \HTMLleftmargini should be set to a wide enough length to contain "Chorus". The default is wide enough for a stanza number.

⚠ spacing   Horizontal spacing relies on *pdftotext*'s ability to discern the layout (-layout option) of the text in the HTML-tagged PDF output. For some settings of \HTMLleftmargini or \HTMLleftskip the horizontal alignment may not work out exactly, in which

⚠ verse margin   case a label may be shifted by one space. During translation to HTML, the stanza numbers are kept out of the left margin, which would have caused *pdftotext* to shift everything over.

8.15.2 newclude package

newclude (*Pkg*)   newclude modifies \label in a non-adaptive way, so newclude must be loaded

⚠ loading   before lwarp is loaded:

```
\documentclass{article}
. . . <font setup>
\usepackage{newclude}
\usepackage[warpHTML]{lwarp}
. . .
```

8.15.3 babel package

babel (*Pkg*)   When French is used, the caption separator is changed to a dash. To restore it to a

⚠ \CaptionSeparator   colon, the following may be placed before lwarp is loaded:

```
\renewcommand*{\CaptionSeparator}{:~}
```

punctuation spaces   Also when French is used, lwarp creates fixed-width space around punctuation by patching \FBcolonspace, \FBthinspace, \FBguillspace, \FBmedkern, \FBthickkern, \FBtextellipsis, and the tilde. If the user's document also

⚠ customized spacing   changes these parameters, the user's changes should be placed inside a warpprint environment so that the user's changes do not affect the HTML output.

8.15.4 polyglossia package

polyglossia (*Pkg*)   lwarp uses cleveref, which has some limitations when using polyglossia, possibly resulting in the error

```
! Undefined control sequence. . . . \__hook begindocument
```

To test compatibility, add

```
\usepackage{cleveref}
```

near the end of the preamble (as the last package to be loaded), and try to compile the print version. It may be necessary to set

```
\setdefaultlanguage{english}
```

or some other language supported by cleveref, then select other languages using \setotherlanguages.

Once the print version works with cleveref and polyglossia, the HTML version should work as well using lwarp.

8.15.5 todonotes and luatodonotes packages

todonotes (*Pkg*) The documentation for todonotes and luatodonotes have an example with a todo
luatodonotes (*Pkg*) inside a caption. If this example does not work it will be necessary to move the
todo outside of the caption.

8.15.6 fixme

fixme (*Pkg*) External layouts (\fxloadlayouts) are not supported.

⚠ external layouts Customized layouts are overwritten by lwarp's versions \AtBeginDocument in order to provide the HTML conversion. If creating a new layout, see lwarp's changes to provide similar for the new layout, inside a warpHTML environment.

User control is provided for setting the HTML styling of the "faces". The defaults are as follows, and may be changed in the preamble after fixme is loaded:

```
\def\FXFaceInlineHTMLStyle{font-weight:bold}
\def\FXFaceEnvHTMLStyle{font-weight:bold}
\def\FXFaceSignatureHTMLStyle{font-style:italic}
\def\FXFaceTargetHTMLStyle{font-style:italic}
```

8.15.7 acro package

⚠ formats Define acronymn formats using \textbf instead of \bfseries etc.

8.15.8 chemfig package

If using \polymerdelim to add delimiters to a \chemfig, wrap both inside a single lateximage:

```
\begin{lateximage}[-chemfig-~\PackageDiagramAltText]
\chemfig{. . . }
\polymerdelim[. . . ]{. . . }
\end{lateximage}
```

8.15.9 chemformula package

⚠ chemformula with chemformula works best without MATHJAX. If MATHJAX is used, \displaymathother
MATHJAX must be used before array, and then \displaymathnormal may be used after. (The chemformula package adapts to array, but does not know about MATHJAX, and MATHJAX does not know about chemformula.)

While using MATHJAX, \displaymathother may also be used for other forms of display and inline math which contain chemformula expressions.

8.15.10 mhchem package

See section 417.

8.15.11 kotex package

kotex (*Pkg*) See section 8.14 regarding *pdflatex* and Korean section names.

⚠   Korean section names

9   Compiling using custom shell commands

lwarp and *lwarpmk* try to make it easy to process print and HTML compilation tasks
in most situations. Depending on the operating system, command-line options,
TEX engine, and lwarp options, the commands **lwarpmk print** and **lwarpmk html**
are automatically set up to correctly recompile the project. These actions may be
overridden using lwarp options, thus allowing the use of packages such as perltex
and pythontex.

9.1   Command options

PrintLatexCmd (*Opt*)
HTMLLatexCmd (*Opt*)

The lwarp options PrintLatexCmd and HTMLLatexCmd are used to set customized
commands to be executed by **lwarpmk print** and **lwarpmk html**.

PrintLatexCmd should be set to shell commands which take project.tex and
generate project.pdf.

HTMLLatexCmd should be set to take project_html.tex and generate project_html.pdf.
lwarpmk will then take project_html.pdf and automatically convert it and
generate project.html.

9.2   Literal character macros

The lwarp package options are parsed by TEX, and so some characters require
the use of a special macro to represent them. See table 10. \LWRopquote and
\LWRopseq may be used to increase operating-system portability. \jobname must
have _html appended for processing HTML. \space may be necessary between
other macros.

⚠   macro not found   To use these macros, either kvoptions-patch must be loaded before lwarp:

```
\usepackage{kvoptions-patch}
\usepackage[
    PrintLatexCmd={ ... } ,
    HTMLLatexCmd={ ... }
]{lwarp}
```

Table 10: Literal character macros

Character	Macro	Comment
%	\LWRpercent	
$	\LWRdollar	
&	\LWRamp	
%	\LWRhash	
\	\LWRbackslash	
' or "	\LWRopquote	Depends on the operating system.
& or &&	\LWRopseq	Depends on the operating system.
(space)	\space	Forces an extra space.
(jobname)	\jobname	Without file extension.

or \lwarpsetup must be used to set PrintLatexCmd and HTMLLatexCmd:

```
\usepackage[...]{lwarp}
\lwarpsetup{
  PrintLatexCmd=
      {
          latex tm \LWRopseq
          dvips -o tm-pics.ps tm.dvi \LWRopseq
          ps2pdf -dALLOWPSTRANSPARENCY tm-pics.ps \LWRopseq
          pdflatex tm.tex
      } ,
  HTMLLatexCmd=
      {
          latex tm_html \LWRopseq
          dvips -o tm_html-pics.ps tm_html.dvi \LWRopseq
          ps2pdf -dALLOWPSTRANSPARENCY tm_html-pics.ps \LWRopseq
          pdflatex tm_html.tex
      }
}
```

9.3 *latexmk*

latexmk (*Prog*) If *latexmk* is used for a project, it may be easiest to continue using it.

latexmk project.tex would create project.pdf as normal.

latexmk project_html.tex would create project_html.pdf, then

lwarpmk pdftohtml project_html.pdf would take project_html.pdf and convert it to project.html.

sagetex (*Pkg*) *latexmk* may simplify the use of packages such as sagetex.

9.4   perltex package

perltex (*Pkg*)   The lwarp package option settings to use perltex would be similar to:

```
\usepackage[
  . . .
  PrintLatexCmd={perltex -latex=pdflatex project.tex} ,
  HTMLLatexCmd={perltex -latex=pdflatex project_html.tex} ,
  . . .
]{lwarp}
```

⚠ "impure" math   Place perltex math expressions between \displaymathother and \displaymathnormal, or \inlinemathother and \inlinemathnormal. See section 8.7.11.

9.5   pythontex package

pythontex (*Pkg*)   An example using pythontex:

```
\usepackage[
  . . .
  PrintLatexCmd={
    pdflatex project.tex \LWRopseq
    pythontex project \LWRopseq
    pdflatex project.tex
  } ,
  HTMLLatexCmd={
    pdflatex project_html.tex \LWRopseq
    pythontex project_html \LWRopseq
    pdflatex project_html.tex
  } ,
  . . .
]{lwarp}
```

Another possibility is to use *latexmk*, placing the `latexmk . . .` commands in the PrintLatexCmd and HTMLLatexCmd options. While using these options, the lwarp option latexmk would not be used.

No attempt has yet been made to make pythontex robust with HTML output. Some

⚠ "impure" math   math objects must be surrounded by \displaymathother ... \displaymathnormal, or \inlinemathother ... \inlinemathnormal. Displays of code may have to be

⚠ HTML look-alike   enclosed inside a lateximage environment to prevent <, > and similar from being interpreted by the browser as HTML entities.

9.6   sympytex package

sympytex (*Pkg*)   For sympytex:

```
\usepackage[
  ...
  PrintLatexCmd={
    pdflatex project.tex \LWRopseq
    python project.sympy \LWRopseq
    pdflatex project.tex
  } ,
  HTMLLatexCmd={
    pdflatex project_html.tex \LWRopseq
    python project_html.sympy \LWRopseq
    pdflatex project_html.tex
  } ,
  ...
]{lwarp}
```

Also see the warnings for pythontex, above.

9.7   Other packages

rterface (*Pkg*)   Other packages such as rterface would be set up similar to pythontex, and the same warnings would apply.

9.8   *make* program

make (*Prog*)   To use lwarp with the *make* program, have the makefile take project.tex and generate the print version project.pdf, as normal. \usepackage{lwarp} must be used, and it generates lwarpmk.conf when the print version is created.

To generate HTML, first have project_html.tex be compiled to generate project_html.pdf. This must be in PDF format. Finally, have project_html.pdf be converted to HTML using **lwarpmk pdftohtml project_html.pdf**, and convert SVG math with **lwarpmk limages**.

9.9   UTF-8 locale

⚠   UTF-8 locale   *lwarpmk* uses the *texlua* program, which sets the "locale" to "C", including for external operating-system calls such as when executing **lwarpmk html**. In some cases, an external program called from the user's document may require the use of a UTF-8 "locale". For UNIX-related operating systems, it may be required to use lwarp's custom compilation options to add a locale change:

```
\usepackage{lwarp}[
  PrintLatexCmd={
    env LC_CTYPE=en_US.UTF-8
      xelatex -shell-escape project.tex
  }
  HTMLLatexCmd={
    env LC_CTYPE=en_US.UTF-8
      xelatex -shell-escape project_html.tex
  }
]
```

ditaa (*Pkg*) The only example seen so far where this is required is the ditaa package, where the locale change allows the use of UTF-8 with XeLATEX and ditaa. To use LuaLATEX instead, the locale change would have to be made inside the ditaa package where its calls the *ditaa* program.

10   EPUB conversion

lwarp does not produce EPUB documents, but it may be told to modify its HTML output to greatly assist in the conversion. An external program may then be used to finish the conversion to EPUB.

<meta> author   To assign the author's name for regular lwarp HTML files, and also for the EPUB, use \HTMLAuthor {⟨*name*⟩}. This assigns the name to the <meta> author element. It may be set empty, and it defaults to \theauthor.

A special boolean is provided to simplify the process of converting lwarp HTML output to EPUB:

> FormatEPUB

FormatEPUB (*bool*)   FormatEPUB changes HTML output for easy EPUB conversion via an
Default: false   external program. Removes per-file headers, footers, and nav. Adds
footnotes per chapter/section.

To help convert lwarp HTML output to EPUB, add

    \booltrue{FormatEPUB}

to the project's source preamble after \usepackage{lwarp}. The EPUB version of the document cannot co-exist with the regular HTML version, so

>   Enter ⇒   **lwarpmk cleanall**

>   Enter ⇒   **lwarpmk html**

>   Enter ⇒   **lwarpmk limages**

to recompile with the FormatEPUB boolean turned on. Several changes are then made to the HTML output:

- Headers, footers, and navigation are removed at file splits.

- Any accumulated footnotes are printed at the bottom of each section.

The resulting files will be ready to be loaded into an EPUB conversion program,
Calibre   such as the open-source program *Calibre* (https://calibre-ebook.com/).

The EPUB conversion program must know what order the files are included. For
⚠ search order   lwarp projects, set the EPUB conversion software to do a breadth-first search of the
files. For *Calibre*, this option is found in

Preferences → Plugins → File type plugins → HTML to Zip

Check the box **Add linked files in breadth first order**. Set the document encoding
⚠ encoding   as utf-8, which is what lwarp generates for HTML, even if the original printed
document uses some other encoding.

The EPUB-conversion program must also know where the section breaks are lo-
⚠ section breaks   cated. For a list of lwarp's section headings, see table 12. For example, an article
class document would break at \section, which is mapped to HTML heading level

<h4>, whereas a book class document would break at \chapter, which is HTML heading level <h3>. For *Calibre*, this option is found in

Preferences → Conversion (Common Options) → Structure Detection → Detect chapters at (XPath expression)

Select the "magic wand" to the right of this entry box, and set the first entry

Match HTML tags with tag name:

to "h4". (Or "h3" for document classes with \chapters.) The Detect chapters at field should then show

//h:h4   — or —   **//h:h3**

This option is also available on the main tool bar at the Convert books button.

Once these settings have been made, the lwarp-generated HTML files may be loaded by *Calibre*, and then converted to an EPUB.

MathJax support

MATHJAX may be used in EPUB documents. Some e-readers include MATHJAX, but any given reader may or may not have a recent version, and may or may not include extensions such as support for siunitx.

lwarp adds some modifications to MathML to support equations numbered by chapter. These modifications may not be compatible with the e-reader's version of MATHJAX, so lwarp requests that a known version be loaded instead. In some cases chapter numbering of equations still doesn't work.

Until math support in EPUB documents is improved, it is recommended to use SVG images instead of MATHJAX, especially for equations numbered by chapter, or where siunitx support is important.

---

11 Word-processor conversion

lwarp may be told to modify its HTML output to make it easier to import the HTML document into a word processor. At the time of this writing, it seems that LIBREOFFICE works best at preserving table layout, but it still has some limitations, such as an inability to automatically assign figure and table frames and captions according to user-selected HTML classes. lwarp provides some assistance in locating these frame boundaries, as shown below.

11.1 Activating word-processor conversion

A special boolean is provided to simplify the process of converting lwarp HTML output to EPUB:

FormatWP

FormatWP (*bool*) Changes HTML output for easier conversion by a word processor.
Default: false Removes headers and nav, prints footnotes per section, and also forces single-file output and turns off HTML debug comments. Additionally, honors the booleans WPMarkFloats, WPMarkMinipages, WPMarkTOC, and WPMarkLOFT.

To help modify lwarp HTML output for easier import to a word processor, add

    \booltrue{FormatWP}

formatting adjustments to the project's source preamble after lwarp is loaded. The following changes are then made to the HTML output:

- If using a class without chapters, \section and lower are shifted up in level for the HTML heading tags. The CSS has not been changed, so the section heading formats will not match the normal HTML output, but when imported to *LibreOffice Writer* the higher section headings will import as **Heading 1** for the title, **Heading 2** for \section, etc.

- Headers, footers, and navigation are removed at file splits.

- Any accumulated footnotes are printed at the bottom of each section.

- Forces single-file output.

- Turns off HTML debugging comments. These are comments appearing inside the HTML code, marking the opening/closing of sections and <div>s, but they are no longer useful when the document has been imported into a word processor.

- An additional <div> with an id encapsulates each float and minipage, which on import into *LibreOffice Writer* causes a thin frame to appear around the text block for each.

- Float captions are given an explicit italic formatting.

- Tabular rule borders are made explicit for *LibreOffice Writer*. LIBREOFFICE displays a light border around each cell while editing, even those which have

no border when printed, and lwarp also uses a light border for thin rules, so it will be best to judge the results using the print preview instead of while editing in LIBREOFFICE.

- \includegraphics and svg math width and height are made explicit for LIBREOFFICE.

- \hspace is approximated by a number of \quads, and rules are approximated by a number of underscores.

- Explicit HTML styles are given to:

    - \textsc, etc.
    - \underline, soul and ulem markup.
    - center, flushleft, flushright.
    - \marginpar, keyfloat, sidenotes, floatflt, and wrapfig.
    - fancybox \shadowbox, etc.
    - The LATEX and TEX logos.

- Honors several booleans:

    WPMarkFloats: Marks the begin and end of floats.

    WPMarkMinipages: Marks the begin and end of minipages.

    WPMarkTOC: Marks the location of the Table of Contents.

    WPMarkLOFT: Marks the locations of the List of Figures/Tables.

    WPMarkMath: Prints LATEX math instead of using images.

    WPTitleHeading: Adjusts title and section headings.

    Several of these may be used to add markers to the HTML text which help determine where to adjust the word processor document after import.

11.2 Additional modifications

WPMarkFloats

WPMarkFloats (*bool*)
Default: false

Adds

```
=== begin table ===
...
=== end ===
```

or

```
=== begin figure ===
...
=== end ===
```

around floats while formatting for word processors. This helps identify boundaries of floats to be manually converted to word-processor frames and captions.

WPMarkMinipages

WPMarkMinipages (*bool*)
Default: false

Adds

```
=== begin minipage ===
. . .
=== end minipage ===
```

around minipages while formatting for word processors. This helps identify boundaries of minipages to be manually converted to word-processor frames.

WPMarkTOC

WPMarkTOC (*bool*)
Default: true

While formatting for word processors, adds

```
=== table of contents ===
```

where the Table of Contents would have been. This helps identify where to insert the actual TOC.

If set false, *the actual toc is printed instead.*

WPMarkLOFT

WPMarkLOFT (*bool*)
Default: false

While formatting for word processors, adds

```
=== list of figures ===    and/or
=== list of tables ===
```

where each of these lists would have been. This helps identify where to insert the actual lists.

If set false, *the actual lists are printed instead.*

WPMarkMath

siunitx
WPMarkMath (*bool*)
Default: false
TeXMaths (*Prog*)

While formatting for word processors, prints math as LaTeX code instead of creating svg images or MATHJAX. This is useful for cut/paste into the *LibreOffice Writer TeXMaths* extension.

When using the siunitx package, enter

```
\usepackage{siunitx}
```

in the *TeXMaths* preamble. Equation numbering is problematic for $\mathcal{A}_{\mathcal{M}}\mathcal{S}$ math environments.

WPTitleHeading

WPTitleHeading (*bool*)
Default: false
section headings

While formatting for word processors, true sets the document title to <h1>, which is expected for HTML documents, but also causes the lower-level section headings to start at **Heading 2** when imported into LIBREOFFICE. Set to false to cause the title to be plain text, and the section headings to begin at **Heading 1**.

Table 11: Section HTML headings for word-processor conversion

	HTML headings*			
	With \chapter		Without \chapter	
	WPTitleHeading		WPTitleHeading	
Section	true	false	true	false
Title	\<h1\>	plain	\<h1\>	plain
\book	\<div\>	\<div\>	\<div\>	\<div\>
\part	\<h2\>	\<h1\>	\<h2\>	\<h1\>
\chapter	\<h3\>	\<h2\>	—	—
\section	\<h4\>	\<h3\>	\<h3\>	\<h2\>
\subsection	\<h5\>	\<h4\>	\<h4\>	\<h3\>
\paragraph	\<h6\>	\<h5\>	\<h5\>	\<h4\>
\subparagraph	\<span\>	\<h6\>	\<h6\>	\<h5\>

* For default depths when not FormatWP, see table 12 on page 206.

11.3 Recommendations

TOC, LOF, LOT For use with *LibreOffice Writer*, it is recommended to:

1. Set \booltrue{FormatWP}
2. Set \booltrue{WPMarkTOC} and \boolfalse{WPMarkLOFT}
3. Use lwarp to generate the HTML document.
4. Copy/paste from the HTML document into an empty *LibreOffice Writer* document.
5. Manually insert a LIBREOFFICE TOC in the LIBREOFFICE document.
6. Manually add frames around each float, adding a caption which is cut/pasted from each float's simulated caption.
7. Manually create cross references.

This process yields a document with an actual LIBREOFFICE Table of Contents, but a simulated List of Figures and List of Tables.

siunitx For siunitx, remember to adjust the preamble as mentioned above.

LO view border options LIBREOFFICE has options in the **View** menu to turn on/off the display of thin borders around table cells and text objects.

11.4 Limitations

Floats and captions are not explicitly converted to LIBREOFFICE floats with their own captions. Floats are surrounded by a thin frame in the LIBREOFFICE editor, and may be marked with WPMarkFloats, but are not given a proper LIBREOFFICE object frame. Captions are given an explicit italic formatting, but not a proper LIBREOFFICE paragraph style.

Cross references are not actual LIBREOFFICE linked cross references.

The List of Figures and List of Tables are not linked. The pasted pseudo LOF and LOT match the numbering of the LATEX and HTML versions.

Equation numbering is not automatic, but the equation numbers in SVG math will match the LATEX and HTML output. SVG math is recommended when using the $\mathcal{AMS}$ environments, which may have multiple numbered equations per object.

As of when last checked, LIBREOFFICE ignores the following:

- Minipage alignment.

- Tabular cell vertical alignment.

- Image rotation and scaling.

- Rounded border corners, which are also used by:

    - \textcircled
    - booktabs trim

- \hspace and rules, also used by algorithmic.

- Coloring of text decorations, used by soul and ulem.

- Overline text decoration, used by romanbar.

LIBREOFFICE also has limitations with frames and backgrounds:

- Multiple lines in an object are framed individually instead of as a whole.

- Nested frames are not handled correctly.

- Images inside boxes are not framed correctly.

- Spans with background colors and frames are not displayed correctly.

12   Modifying lwarp

locating something To quickly find the source for a package in `lwarp.dtx`, search for `*packagename`, such as `*siunitx`.

Likewise, to quickly find the source for a file in `lwarp.dtx`, search for `*filename`, such as `*lwarp.css`.

Purely text-based packages probably will work as-is when generating HTML.

Look to existing code for ideas on how to expand into new code.

image of TEX output An environment may be converted to a `lateximage` then displayed with an image of the resulting LATEX output. See section 95 for an example of the `picture` environment.

css classes To create a custom HTML block or inline css class, see section 54.10.

print/HTML macros To create print and HTML versions of the same macro or environment, see section 38.

⚠ TEX boxes Any TEX boxes must be undone, as svg math or `lateximages` require `\newpage`, which will not work in a TEX box.

12.1   Creating a development system

The following creates a local development system for lwarp on a TeXLive system in a UNIX-like environment. Doing so allows anything requesting lwarp to use the development version instead of whichever version is installed in TeXLive.

Create a development directory:

Place into this directory `lwarp.dtx` and `lwarp.ins`.

To create `lwarp.sty`, execute

Enter ⇒   **pdflatex lwarp.ins**

which creates `lwarp.sty` and several hundred additional `lwarp-*.sty` files for the various packages which are supported.

To create the initial documentation `lwarp.pdf`, execute

Enter ⇒   **pdflatex lwarp.dtx**

To make the development files visible to other projects:

Create the directory

/usr/local/texlive/texmf-local/tex/latex/local/lwarp

Inside this directory, create the file `update`, containing:

```
rm lwarp-*.sty
ln -s /path_to_dev_directory/lwarp*.sty .
ln -s /path_to_dev_directory/lwarp_baseline_marker.png .
ln -s /path_to_dev_directory/lwarp_baseline_marker.eps .
mktexlsr
```

Run ./update now, and whenever a new lwarp-* package is added.

To make the development version of *lwarpmk* visible to other projects:

```
cd /opt
ln -s /usr/local/texlive/texmf-local/bin/x86_64-linux texbin_local
cd texbin_local
ln -s ../../scripts/lwarp/lwarpmk.lua lwarpmk
cd /usr/local/texlive/texmf-local/scripts/
mkdir lwarp
cd lwarp
ln -s /path_to_dev_directory/lwarpmk.lua lwarpmk
```

Verify that the correct version is found with

Enter ⇒ **which lwarpmk**

To make the local versions visible to the shell:

Paths must be set by the shell startup, such as in .bashrc and .cshrc:

In .bashrc:

```
PATH=/opt/texbin_local:/opt/texbin:$PATH
```

In .cshrc:

```
setenv PATH ${HOME}/bin:/opt/texbin_local:/opt/texbin:${PATH}
```

To fully compile the lwarp documentation and indexes:

```
pdflatex lwarp.ins
pdflatex lwarp.dtx
pdflatex lwarp.dtx                          <if necessary>
makeindex -s gglo.ist -o lwarp.gls lwarp.glo   <indexes>
splitindex lwarp.idx – -s gind.ist
pdflatex lwarp.dtx
pdflatex lwarp.dtx                          <if necessary>
makeindex -s gglo.ist -o lwarp.gls lwarp.glo   <indexes>
splitindex lwarp.idx – -s gind.ist              <again>
pdflatex lwarp.dtx
pdflatex lwarp.dtx                          <if necessary>
```

(The multiple rounds of index processing are required to fully resolve the final Index of Indexes.)

To make it easier to update the documentation after a minor change, it is useful to create a command script called make_index, containing:

```
makeindex -s gglo.ist -o lwarp.gls lwarp.glo
splitindex lwarp.idx -- -s gind.ist
```

⚠ references Note that Index of Indexes and the cross-references to the indexes may not be correct until the above has been accomplished.

12.2 Modifying a package for lwarp

If a class loads additional packages, it will be required to modify the class for lwarp, since lwarp must be loaded before most other packages.

To work with lwarp, a class must first set up anything which replicates the functions of the basic LATEX classes, load any required fonts, then load lwarp, then finally load and adjust any other required packages.

When creating HTML, lwarp redefines the \usepackage and \RequirePackage macros such that it first looks to see if a lwarp-<packagename>.sty version exists. If so, the lwarp version is used instead. This modular system allows users to create their own versions of packages for lwarp to use for HTML, simply by creating a new package with a lwarp- prefix. If placed in the local directory along with the source code, it will be seen by that project alone. If placed alongside the other lwarp-packages where TEX can see it, then the user's new package will be seen by any documents using lwarp. (Remember mktexlsr or texhash.)

An lwarp-<packagename>.sty package is only used during HTML generation. Its purpose is to pretend to be the original package, while modify anything necessary to create a successful HTML conversion. For many packages it is sufficient to simply provide nullified macros, lengths, counters, etc. for anything which the original package does, while passing the raw text on to be typeset. See the pre-existing lwarp- packages for examples.

Anything the user might expect of the original package must be replaced or emulated by the new lwarp- package, including package options, user-adjustable counters, lengths, and booleans, and conditional behaviors. In many of these packages, most of the new definitions have a "local" prefix according to the package name, and @ characters inside the name, which hides these names from the user. In most cases these macros will not need to be emulated for HTML output. Only the "user-facing" macros need to be nullified or emulated.

Each lwarp-* package should first call either of:

> \LWR@ProvidesPackageDrop
> — *or* —
> \LWR@ProvidesPackagePass

If "Drop"ped, the original print-version package is ignored, and only the lwarp-version is used. Use this where the original print version is useless for HTML. If "Pass"ed, the original package is loaded first, with the user-supplied options, then the lwarp- version continues loading as well. See section 464 (ntheorem) for an example of selectively disabling user options for a package. Use this when HTML output only requires some modifications of the original package. For a case where the original package is usable without changes, there is no need to create a lwarp-version.

12.2.1 Adding a package to the lwarp.dtx file

When adding a package to lwarp.dtx for permanent inclusion in lwarp, provide the lwarp-<packagename> code in lwarp.dtx, add its entry into lwarp.ins, and also remember to add

> \LWR@loadafter{<packagename>}

to `lwarp.dtx` in section 22.1. This causes lwarp to stop with an error if packagename is loaded before lwarp. Finally, add an entry in table 2, Supported packages and features, and also the Updates section.

12.3   Modifying a class for lwarp

If a class loads additional packages, it will be required to modify the class for lwarp, since lwarp must be loaded before most other packages.

To work with lwarp, a class must first set up anything which replicates the functions of the basic LATEX classes, load any required fonts, then load lwarp, then finally load and adjust any other required packages.

12.4   Testing lwarp

Compiling `lwarp.ins` generates all the `*.sty` files for lwarp. It can be useful to create additional `*.ins` files to be able to recompile only the pieces which have changed.

compiling individual packages

core.ins (*file*)

For example, to be able to recompile the lwarp core alone, copy `lwarp.ins` to `core.ins`, then modify `core.ins` to only compile:

```
\generate{
\file{lwarp.sty}{\from{lwarp.dtx}{package}}
}
```

For individual packages, create `packagename.ins`, set to compile only:

```
\generate{
\file{lwarp-packagename.sty}{\from{lwarp.dtx}{packagename}}
}
```

When changes have been made, test the print output before testing the HTML. The print output compiles faster, and any errors in the printed version will be easier to figure out than the HTML version.

compiling css and other generated files

Remember that the configuration files are only rewritten when compiling the printed version of the document.

When changing the source to *lwarpmk* or a css file in `lwarp.dtx`:

1. Change the source in `lwarp.dtx`.

2. `pdflatex lwarp.ins` -or- `pdflatex core.ins`

3. If modifying *lwarpmk* the new version should now be active.

4. If modifying css files or other files generated by lwarp:

   (a) For the document, `lwarpmk print` to update the css files in the project.
   (b) Reload the HTML document to see the effect of the new css files.

5. If done testing, `pdflatex lwarp.dtx` to update the lwarp documentation.

Sometimes it is worth checking the <project>_html.pdf file, which is the PDF containing HTML tags. Also, <project>_html.html has the text conversion of these tags, before the file is split into individual HTML files.

It is also worth checking the browser's tools for verifying the correctness of HTML and CSS code.

12.5 Modifying *lwarpmk*

lwarpmk (*Prog*)
lwarpmk.lua (*file*)

In most installations, lwarpmk.lua is an executable file located somewhere the operating system knows about, and it is called by typing **lwarpmk** into a terminal.

A project-local copy of lwarpmk.lua may be generated, modified, and then used to compile documents:

1. Add the lwarpmk option to the lwarp package.

2. Recompile the printed version of the document. The lwarpmk option causes lwarp to create a local copy of lwarpmk.lua

3. The lwarpmk option may now be removed from the lwarp package.

4. Copy and rename lwarpmk.lua to a new file such as mymake.lua.

5. Modify mymake.lua as desired.

6. If necessary, make mymake.lua executable.

7. Use mymake.lua instead of lwarpmk.lua.

13 Troubleshooting

13.1 lwarp package error conditions and warnings

lwarp tests for a number of error conditions and prints appropriate warnings. The
following is a summary of these conditions.

13.1.1 Configuration file `lwarpmk.conf`

File does not exist: The configuration file must exist for lwarpmk.

Incorrect `Unix` /`Windows` selection: The operating system which was detected by
lwarp. So far only `Unix` and `Windows` are supported.

Incorrect delimiter characters. Older versions of *lwarpmk* used a different de-
limiter.

Source name is set to `lwarp`: lwarp has recently been recompiled in this direc-
tory, which overwrote the project's configuration files. This also occurs if
lwarpmk is executed in lwarp's source directory.

Incorrect operating system: The configuratio file was set for a different operating
system, perhaps due to sharing in a collaborative project.

Outdated configuration files: lwarp has been updated since this projects was last
compiled. If there appears to be a valid print command in the file, lwarpmk
displays this to instruct the user how to recompile the print version, which
then updates the configuration files.

The designated source file does not exist: For whatever reason…

Unknown engine: lwarp cannot determing which engine is being used. Sup-
ported are DVI LaTeX, PDF LaTeX, XƎLaTeX, LuaLaTeX, and upLaTeX.

13.1.2 Image generation with `lwarpmk limages`

"Wait a moment for the images to complete before reloading page.":
Images are generated by background tasks. If the document is reloaded
before these tasks are complete, some images may not yet be generated.
lwarpmk tries to wait for background tasks to complete before exiting.

HTML version does not exist: Images are extracted from the HTML version, which
must be compiled before images are generated.

`*-images.txt` does not exist: This file tells which images to extract from the
HTML file. If the file does not exist, it may be that no SVG math or `lateximages`
were used. If so, **`lwarpmk limages`** is not necessary.

Cross references are not correct: The document must have up-to-date cross ref-
erences to locate the images to extract. A number of conditions may cause
incorrect cross references.

"WARNING: Images will be incorrect.": An image reference was not found. Re-
compile.

`lwarpmk epstopdf *` or `lwarpmk pdftosvg *`: Errors if filenames are not found.

13.1.3 Default bitmapped font

lwarp requires the use of a vector font. If lwarp detects that the document uses the default COMPUTER MODERN font, and the cm-super package is not installed, it is assumed that the font is bitmapped. An error is generated, along with the recommendation to install cm-super or use lmodern.

13.1.4 Packages

Loaded before lwarp: Some packages and classes must be loaded before lwarp. These include input and font encoding, morewrites and newclude, and a number of CJK-related packages and classes.

Loaded after lwarp: Most packages which are modified by lwarp must be loaded after lwarp.

Loaded never: Some packages do not work with lwarp. An error is generated, along with a list of alternatives to consider.

Specific packages: Some packages enforce a specific load order vs. certain other packages.

Patching error: lwarp tries to patch some packages using xpatch. If the original package has been updated more recently than lwarp, a patch may not work. It may be necessary to use an older version of the package until lwarp is updated.

longtable: lwarp's longtable package issues detailed error messages regarding the use of the table headers and footers.

polyglossia: If used, an informative message is printed to instruct the user to be sure to set a language, without which an error will occur.

babel or polyglossia: An informative message is printed to note that not all langauges are supported by cleveref.

13.1.5 Compiling

SideTOCDepth < FileDepth: A warning is displayed if these counters are set such that the sideTOC will not be able to access all pages of the website.

Filenames: lwarp may generate file names from section names. While doing so, the filenames are simplified, and special characters and math are removed. If this process generates a duplicate filename, and warning is generated, describing the filename and which section name generated it, and a unique file number is appended to the file name. A warning is also issued if dollar-delimited math is used. Parenthesis-delimited math is recommended instead.

⚠ HTML corrupted **Multirow:** When \multirow or \multicolrow are used, \mrowcell or \mcolrowcell must be placed in the appropriate cells to avoid corrupted HTML output.

(width,height) missing a comma: \makebox and \framebox can accept a parenthesis-delimited width and height, which must be separated by a comma.

"Load graphicx or graphics for improved SVG math baselines.": SVG math sizing and baselines are improved if either of these packages are used.

"Load graphicx or graphics for improved XeTeX logo.": If these packages are loaded, the XƎLATEX logo can use the reversed "E".

"It is recommended to use [width=xx\linewidth] instead of [scale=yy] ": Browser support of scale does not have the same effect as in LATEX.

13.2   Using the lwarp package

The following address problems which may occur, and possible solutions to each.

Also see:
Section 7.11: Commands to be placed into the warpprint environment
Section 8: Special cases and limitations

⚠ HTML corrupted **Text is not converting correctly / corrupted HTML tags:**

- Font-related UTF-8 information must be embedded in the PDF file. See section 7.4 regarding bitmapped vs. vector fonts.
- See section 8.2.1 regarding HTML entities and the characters &, <, and >.

⚠ dotlessj **Dotless j (\j):** See section 7.4 regarding cmap, mmap.

Undefined HTML settings:

- See the warning regarding the placement of the HTML settings at section 7.6.

Tabular problems: See section 8.10.1.

Obscure error messages:

Print first: Be sure that a print version of the document compiles and that your document's LATEX code is correct, before attempting to generate an HTML version.

\end{warpHTML}, \end{warpprint}, \end{warpall}, \end{warpMathJax}:

⚠ warpHTML, warpprint,   Each of these must be without any other characters on the same line.
warpMathJax, warpall

"Runaway argument? File ended while scanning use of \next: Don't use warpHTML, warpprint, warpall, or warpMathJax inside itself.

Options clash: If using memoir, see section 8.13.

"Missing \begin{document}.": Some packages require that their options be specified before lwarp is loaded, or via the package's setup macro, especially if these options include the use of braces. See section 8.1.

"No room for a new \write.": Before \usepackage{lwarp}, add:

```
\usepackage{morewrites}
\morewritessetup{allocate=10}
```

"! TeX capacity exceeded, sorry [text input levels=15].": Packages were nested too many levels deep. Locate the file texmf.cnf for your distribution, and add the line

```
max_in_open = 30
```

"Missing $ inserted.": If using a filename or URL in a footnote or \item, escape underscores with _.

"Label(s) may have changed. Rerun to get cross-references right.":

This warning may repeat endlessly if a math expression is used in a caption. Simple math expressions such as $X=1$ may be replaced with

    \textit{X}\,=\,1

"Temporary page! LaTeX was unable to guess the total number of pages …":
Harmless. Recompile the document one more time.

"Leaders not followed by proper glue":
This can be caused by a missing l@<floattype> or l@<sectiontype>
definition. See lwarp's definitions for examples.

"Improper \prevdepth": lateximages and SVG math require \newpage, which
cannot work inside TEX boxes or \ensuremath. Anything using \newsavebox,
\newbox, lrbox, \savebox, \hbox, \vbox, \usebox, \sbox, etc., must be
modified to work without box commands.

If you find something using \ensuremath, have it temporarily set:

    \LetLtxMacro\@ensuredmath\LWR@origensuredmath

inside a group first.

LWR@texboxdepth   As a stop-gap measure, you may wish to try incrementing the counter
LWR@texboxdepth before the problematic macro, and then decrement-
ing it after. Doing so tells lwarp to avoid using a \newpage inside the
macro, which may avoid this error.

⚠ macros in section, table,   Also, custom macros which appear inside a section, figure, or table
figure names   name should be made robust since they appear inside the .toc, .lof,
or .lot files. Use \newrobustcmd or \robustify from etoolbox, xparse,
etc.

⚠ BibTeX   If using BibTeX, see section 8.6.9.

⚠ polyglossia   **"! Undefined control sequence. . . . __hook begindocument":**
See section 8.15.4 if using polyglossia.

⚠ custom macros for   **"\begin{equation} ended by \end{document}":** Do not use custom macros
environments   such as \beq and \eeq to replace

    \begin{equation}
    . . .
    \end{equation}

⚠ \LWR@formatted   **"Misplaced \omit":** If using \LWR@formatted to define new macros for
print and HTML modes, see section 38 regarding \LWR@expandableformatted.

"Token not allowed in a PDF string": This hyperref warning appears
while creating the print-mode document, not HTML. A low-level macro
is being used in a section name which appears in the PDF bookmarks.
hyperref removes this macro from the bookmark, and warns of doing
so. To avoid this warning, use \pdfstringdefDisableCommands in the
preamble to define simplified replacement macros for each, or use
\texorpdfstring in the \section or related macro to declare what to
use for the TEX text, v.s. the PDF bookmark. See the hyperref manual.

⚠ quote character   **"Command \textquoteright invalid in math mode":** This can occur when
the document source has math containing the slanted quote ' charac-
ter, instead of using the upright quote ' character.

⚠ "impure" math objects   **Complicated objects inside math:** Some objects, such as TikZ, may not
compile in lwarp's normal math emulation. Insert

> \displaymathother —*or*— \inlinemathother

before the math, and then

> \displaymathnormal —*or*— \inlinemathnormal

when displaying "normal" math. See section 8.7.11.

Slow compliation of math objects: Complicated math objects can also cause problems with alt tags, resulting in very slow compilation, large alt tags, and possible crashes. Use \inlinemathother ... \inlinemathnormal or \displaymathother ... \displaymathnormal around the math expression.

⚠ MATHJAX **Incorrect MATHJAX:** Some objects do not convert to MATHJAX. Use \displaymathother before these objects, then \displaymathnormal to return to "normal" display math. See section 8.7.11.

Missing sections: See section 7.6 regarding the FileDepth and SideTOCDepth counters, and the use of \tableofcontents in the home page.

Misnumbered footnotes from section headings: See section 8.5.4.

Missing HTML files:

- See the warning regarding changes to the HTML settings at section 7.6.
- Ensure that the filenames are unique after math and short words are removed. See FileSectionNames at section 7.6.

Missing / incorrect cross-references:

- Use lwarpmk again followed by lwarpmk html or lwarpmk print to compile the document one more time.

labels
⚠ label characters

- Labels with special characters may be a problem. It is best to stick with alpha-numeric, hyphen, underscore, and perhaps the colon (if not French).

\nameref
⚠ empty link

\nameref refers to the most recently-used section where the \label was defined. If no section has been defined before the \label, the link will be empty. Index entries also use \nameref and have the same limitation.

⚠ cleveref page numbers

- cleveref and varioref are supported, but printed page numbers do not map to HTML, so a section name or a text phrase are used for \cpageref and \cpagerefrange. This phrase includes \cpagerefFor, which defaults to "for".

Ex:
> \cpageref{tab:first,tab:second}
in html becomes:
> "pages **for** table 4.1 and **for** table 4.2"

See \cpagerefFor at page 753 to redefine the message which is printed for page number references.

BibTeX errors with \etalchar: See section 8.6.9.

Malformed URLs: Do not use the % character between arguments of \hyperref, etc., as this character is among those which is neutralized for inclusion in HTML URLS.

Em-dashes or En-dashes in listing captions and titles:

Use X∃LᴬTEX or LuaLᴬTEX.

Floats out of sequence:

> **Mixed "Here" and floating:** Floats [H]ere and regular floats may become
> out of order. \clearpage if necessary.
>
> **Caption setup:** With \captionsetup set the positions for the captions
> above or below to match their use in the source code.

Images are appearing in strange places:

- When images are added or removed, Enter **lwarpmk limages** to refresh
  the lateximage images.

svg images:

⚠ adding/removing   When a math expression, picture, or TikZ environment is added or removed,
the svg images must be re-created by entering **lwarpmk limages** to maintain
the proper image-file associations. Inline svg math may be hashed and thus
not need to be recreated, but display math and objects such as TikZ may
move to new image numbers when the document is changed.

recompile first   Before attempting to create the svg image files, *lwarpmk* verifies that the
HTML version of the document exists and has correct internal image refer-
ences.[16] If it is necessary to recompile the document's HTML version one
more time, *lwarpmk* usually will inform the user with an error message, but
there are some conditions which cannot be detected, so the user should
watch for the LaTeX recompile warnings.

⚠ HTML instead of images   If HTML appears where an svg image should be, recompile the document one
more time to get the page numbers back in sync, then remake the images
one more time.

Incorrect svg images will also occur if the document changes the page
⚠ page counter   counter:

    \setcounter{page}{<value>}

The page counter must *not* be adjusted by the user.

Expressing math as svg images has the advantage of representing the math
exactly as LaTeX would, but has the disadvantage of requiring an individual
⚠ Lots of files!   file for each math expression. For inline math, and some other objects,
lwarp uses an MD5 hash on its LaTeX source to combine multiple instances
of identical inline expressions into a single image file, but display math and
other environments such as picture and TikZ require one image file each.
For a document with a large amount of math, see section 5.5 to use MathJax
instead.

Plain-looking document:

- The document's css stylesheet may not be available, or may be linked
  incorrectly. Verify any \CSSFilename statements point to a valid css
  file.

⚠ HTML corrupted   **Broken fragments of HTML:**

- Check the PDF file used to create HTML to see if the tags overflowed the
  margin. (This is why such large page size and margins are used.)

Changes do not seem to be taking effect:

---

[16]This becomes important when dealing with a document containing thousands of images.

- Be sure to `lwarpmk clean`, recompile, then start by reloading the home page. You may have been looking at an older version of the document. If you changed a section name, you may have been looking at the file for the old name.
- See the warning regarding changes to the HTML settings at section 7.6.
- Verify that the proper CSS is actually being used.
- The browser may compensate for some subtle changes, such as automatically generating ligatures, reflowing text, etc.

Un-matched conditional compiles:

- Verify the proper begin/end of `warpprint`, `warpHTML`, and `warpall` environments.

13.2.1 Debug tracing output

`\tracinglwarp` When `\tracinglwarp` is used, lwarp will add extra tracing messages to the `.log` file. The last several messages may help track down errors.

Place `\tracinglwarp` just after `\usepackage{lwarp}` to activate tracing.

13.3 Compiling the `lwarp.dtx` file

`lwarp_tutorial.tex`: Copy or link `lwarp_tutorial.txt` from the TDS doc directory to the source directory, or wherever you wish to compile the documentation. This file is included verbatim in the documentation, but is in the doc directory so that it may be found by *texdoc* and copied by the user.

Illogical error messages caused by an out-of-sync `lwarp.sty` file:

1. Delete the `lwarp.sty` file.
2. Enter **`pdflatex lwarp.ins`** to generate a new `lwarp.sty` file.
3. Enter **`pdflatex lwarp.dtx`** to recompile the `lwarp.pdf` documentation.

Un-nested environments:

Be sure to properly nest:

- `\begin{macrocode}` and `\end{macrocode}`
- `\begin{macro}` and `\end{macro}`
- `\begin{environment}` and `\end{environment}`

14 Trademarks

- TEX is a trademark of American Mathematical Society.

- ADOBE® and ADOBE *Framemaker*® are either registered trademarks or trademarks of ADOBE SYSTEMS INCORPORATED in the United States and/or other countries.

- LINUX® is the registered trademark of Linus Torvalds in the U.S. and other countries.

- MAC OS® is a trademark of APPLE INC.

- MADCAP FLARE™ is the property of MADCAP SOFTWARE, INC.

- MATHJAX is copyright 2009 and later. The MATHJAX CONSORTIUM is a joint venture of the AMERICAN MATHEMATICAL SOCIETY (AMS) and the SOCIETY FOR INDUSTRIAL AND APPLIED MATHEMATICS (SIAM) to advance mathematical and scientific content on the web.

- MICROSOFT®, ENCARTA, MSN, and WINDOWS® are either registered trademarks or trademarks of MICROSOFT CORPORATION in the United States and/or other countries.

- UNIX® is a registered trademark of THE OPEN GROUP.

- YOUTUBE™ is trademark of GOOGLE LLC.

File 1 **`lwarp.sty`**

15 Implementation

This package is perhaps best described as a large collection of smaller individual technical challenges, in many cases solved through a number of ~~crude hacks~~ clever tricks. Reference sources are given for many of the solutions, and a quick internet search will provide additional possibilities.

Judgement calls were made, and are often commented. Improvements are possible. The author is open to ideas and suggestions.

Packages were patched for re-use where they provided significant functionality. Examples include xcolor with its color models and conversion to HTML color output, and siunitx which provides many number and unit-formatting options, almost all of which are available in pure-text form, and thus easily used by *pdftotext*.

Packages were emulated where their primary purpose was visual formatting which is not relevent to HTML output. For example, packages related to sectioning are already patched by numerous other packages, creating a difficult number of combinations to try to support, and yet in HTML output all of the formatting is thrown away, so these packages are merely emulated.

Packages with graphical output are allowed as-is, but must be nested inside a `lateximage` environment to preserve the graphics.

Testing has primarily been done with the Iceweasel/Firefox browser.

Table 12: Section depths and HTML headings

Section	LATEX depth	HTML headings *
Title of the entire website		`<h1>`
(none)	-5	new for this package
book	-2	`<div class = "book">`
part	-1	`<h2>`
chapter	0	`<h3>`
section	1	`<h4>`
subsection	2	`<h5>`
subsubsection	3	`<h6>`
paragraph	4	`<span class = "paragraph">`
subparagraph	5	`<span class = "subparagraph">`
listitem	7	new for this package, used for list items

* If `FormatWP` is true, section headings may be adjusted, depending on `WPTitleHeading`. See table 11 on page 190.

16 Section depths and HTML headings

Stacks are created to track depth inside the LATEX document structure. This depth is translated to HTML headings as shown in table 12. "Depth" here is not depth in the traditional computer-science stack-usage sense, but rather a representation of the nesting depth inside the LATEX document structure.

When starting a new section, the program first must close out any existing sections and lists of a deeper level to keep the HTML tags nested correctly.

Support for the memoir package will require the addition of a book level, which may push the HTML headings down a step, and also cause subsubsection to become a `<div>` due to a limit of six HTML headings.

It is possible to use HTML5 `<section>` and `<h1>` for all levels, but this may not be well-recognized by older browsers.

Fixed levels for parts and chapters allow the CSS to remain fixed as well.

17 Source code

This is where the documented source code for lwarp begins, continuing through the following sections all the way to the change log and index at the end of this document.

The following sections document the actual implementation of the lwarp package.

line numbers The small numbers at the left end of a line refer to line numbers in the lwarp.sty file.

subjects Blue-colored tags in the left margin aid in quickly identifying the subject of each paragraph. These are often the targets of index entries.

Prog Lwarp Black-colored tags in the left marign are used to identify programming objects such as files, packages, environments, booleans, and counters. Items without a tag index entries are command macros. Each of these also appears in the index as individual entries, and are also listed together under "files", "packages", "environments", "booleans", and "counters".

⚠ warnings Special warnings are marked with a warning icon.

for HTML output: Green-colored tags in the left margin show which sections of source code apply to **for PRINT output:** the generation of HTML, print, or both forms of output.
for HTML & PRINT:

— lwarp source code begins on the following page —

18 Required LaTeX 2ε format

This date is to ensure a recent enough version of lthooks, ltcmdhooks, ltpara, ltfilehook, ltshipout.

```
1 \NeedsTeXFormat{LaTeX2e}[2021/06/01]
```

19 Warn if using PDF tagging

```
2 \IfPackageLoadedTF{tagpdf-base}% *88*
3    {%
4        \PackageError{lwarp}
5        {%
6            PDF tagging is not supported yet.\MessageBreak%
7            Comment out \string\DocumentMetadata\space for HTML%
8        }
9        {Still in development!}
10   }
11   {}
```

20 Detecting the TEX engine — *pdflatex, lualatex, xelatex*

See: http://tex.stackexchange.com/a/47579.

Detects XƎTEX and LuaLaTeX:

```
12 \RequirePackage{iftex}[2019/11/07]
13 \RequirePackage{ifpdf}
14 \RequirePackage{ifptex}% in case TL2019 or earlier
15
16 \newif\ifxetexorluatex
17
18 \ifXeTeX
19    \xetexorluatextrue
20 \else
21    \ifLuaTeX
22        \xetexorluatextrue
23    \else
24        \xetexorluatexfalse
25    \fi
26 \fi
```

21 Early package requirements

etoolbox (*Pkg*)   Provides \ifbool and other functions.

```
27 \RequirePackage{etoolbox}[2020/10/05]%
```

Patch to fix copy of environment with a \par:

https://github.com/josephwright/etoolbox/issues/35

```
28 \long\def\etb@carsquare#1#2#3\@nil{#1#2}%
```

verifycommand (*Pkg*) Verify macros before patching.

```
29 \RequirePackage{verifycommand}
```

xpatch (*Pkg*) Patches macros with optional arguments.

```
30 \RequirePackage{xpatch}
```

ifplatform (*Pkg*) Provides \ifwindows to try to automatically detect WINDOWS OS.

```
31 \RequirePackage{ifplatform}% sense op-system platform
```

letltxmacro (*Pkg*)

```
32 \RequirePackage{letltxmacro}
```

22   Package load order

Several packages must never be used with lwarp, others should only be loaded before lwarp, and others should only be loaded after. The lwarp core checks most of these cases. In some lwarp-* packages, \LWR@loadbefore is used to trigger an error if they are loaded after lwarp, while additional code provides necessary patches for when they are loaded before.

Packages which must be loaded after lwarp are enforced by a large number of \LWR@loadafter statements, below. Some packages are emulated by memoir, and so these are tested by \LWR@notmemoirloadafter, which does not cause an error if memoir is used.

\LWR@checkloadfilename is used to check each filename to see if it must never be loaded, or must always be loaded before lwarp.

22.1   Tests of package load order

\LWR@loadafter {⟨*packagename*⟩}   Error if this package was loaded before lwarp.

```
33 \newcommand*{\LWR@loadafter}[1]{%
34 \IfPackageLoadedTF{#1}
35 {
36     \PackageError{lwarp}
37        {%
38            Package #1,\MessageBreak
39            or one which uses #1,\MessageBreak
40            must be loaded after Lwarp.\MessageBreak
41            Enter 'H' for possible solutions%
```

```
42          }
43          {%
44              Move ``\protect\usepackage{#1}'' after
45              ``\protect\usepackage{lwarp}''.\MessageBreak
46              Package #1 may also be loaded by something else,\MessageBreak
47              which must also be moved after Lwarp.%
48          }
49 }
50 {\relax}
51 }
```

\LWR@notmemoirloadafter {⟨*packagename*⟩}   Error if not memoir class and this package was loaded before
lwarp.

memoir emulates many packages, and pretends that they have already been
loaded.

```
52 \IfClassLoadedTF{memoir}
53 {\newcommand*{\LWR@notmemoirloadafter}[1]{}}
54 {\LetLtxMacro\LWR@notmemoirloadafter\LWR@loadafter}
```

\LWR@notltjloadafter {⟨*packagename*⟩}   Error if not a ltjs* class and this package was loaded before
lwarp.

```
55 \LetLtxMacro\LWR@notltjloadafter\LWR@loadafter
56
57 \IfClassLoadedTF{ltjarticle}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
58 \IfClassLoadedTF{ltjbook}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
59 \IfClassLoadedTF{ltjreport}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
60 \IfClassLoadedTF{ltjsarticle}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
61 \IfClassLoadedTF{ltjsbook}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
62 \IfClassLoadedTF{ltjsreport}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
63 \IfClassLoadedTF{ltjspf}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
64 \IfClassLoadedTF{ltjskiyou}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
65 \IfClassLoadedTF{ltjtarticle}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
66 \IfClassLoadedTF{ltjtbook}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
67 \IfClassLoadedTF{ltjtreport}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
```

\LWR@loadbefore {⟨*packagename*⟩}   Error if this package is loaded after lwarp.

```
68 \newcommand*{\LWR@loadbefore}[1]{%
69 \IfPackageLoadedTF{#1}
70 {\relax}
71 {
72     \PackageError{lwarp}
73     {%
74         Package #1 must be loaded before lwarp.\MessageBreak
75         Enter 'H' for possible solutions%
76     }
77     {Move ``\protect\usepackage{#1}'' before ``\protect\usepackage{lwarp}''.}
78 }
79 }
```

\LWR@checkloadbefore {⟨*packagename*⟩}

Given \LWR@tempone is the package name to compare to, if package names match, error if it is loaded after lwarp.

```
80 \newcommand*{\LWR@checkloadbefore}[1]{%
81     \ifdefstring{\LWR@tempone}{#1}{%
82         \LWR@loadbefore{#1}%
83     }{}%
84 }
```

\LWR@loadnever {⟨*badpackagename*⟩} {⟨*replacementpkgnames*⟩}

The first packages is not supported, so tell the user to use the second instead. Factored from \LWR@checkloadnever and \LWR@earlyloadnever.

```
85 \newcommand*{\LWR@loadnever}[2]{%
86 \PackageError{lwarp}
87 {%
88     Package #1 is not yet supported\MessageBreak
89     by lwarp's HTML conversion%
90     \ifblank{#2}{}{%
91         .\MessageBreak
92         Package(s)\MessageBreak
93         \space\space#2\MessageBreak
94         may be useful instead%
95     }%
96 }
97 {%
98     Package #1 might conflict with lwarp in some way,\MessageBreak
99     or is superceded by another package.%
100     \ifblank{#2}{}{%
101         \MessageBreak
102         For possible alternatives, see package(s) #2.%
103     }%
104 }
105 }
```

\LWR@afterloadnever {⟨*badpackagename*⟩} {⟨*replacementpkgnames*⟩}

Given: \LWR@tempone is set to the package name being tested against, if this package name is the bad packagename, suggest the replacements instead. This is used when loading packages after lwarp.

```
106 \newcommand*{\LWR@afterloadnever}[2]{%
107     \ifdefstring{\LWR@tempone}{#1}{%
108         \LWR@loadnever{#1}{#2}%
109     }{}%
110 }
```

\LWR@earlyloadnever {⟨*badpackagename*⟩} {⟨*replacementpkgname*⟩}

The first package is not supported, so tell the user to use the second instead. This version checks immediately for packages which may have been loaded before lwarp.

```
111 \newcommand*{\LWR@earlyloadnever}[2]{%
112     \IfPackageLoadedTF{#1}{%
113         \LWR@loadnever{#1}{#2}%
```

```
114    }{}%
115 }
```

`\LWR@earlyclassloadnever` {⟨*badclassname*⟩} {⟨*replacementclassname*⟩}

The first class is not supported, so tell the user to use the second instead. This version checks immediately for classes which may have been loaded before lwarp.

```
116 \newcommand*{\LWR@earlyclassloadnever}[2]{%
117 \IfClassLoadedTF{#1}{%
118 \PackageError{lwarp}
119 {%
120     Class #1 is not supported\MessageBreak
121     by lwarp's HTML conversion%
122     \ifblank{#2}{}{%
123         .\MessageBreak
124         #2 may be useful instead%
125     }%
126 }
127 {%
128     Class #1 might conflict with lwarp in some way,\MessageBreak
129     or is superceded by another class.%
130     \ifblank{#2}{}{%
131         \MessageBreak
132         For a possible alternative, see #2.%
133     }%
134 }
135 }{\relax}%
136 }
```

22.2   Error for disallowed packages and classes loaded before lwarp

`\LWR@checkloadnevers` Checks against a list of incompatible packages.

```
137 \newcommand*{\LWR@checkloadnevers}{
138 \LWR@checkloadnever{ae}{cm-super, lmodern}
139 \LWR@checkloadnever{aecompl}{cm-super, lmodern}
140 \LWR@checkloadnever{aecc}{cm-super, lmodern}
141 \LWR@checkloadnever{alg}{algorithm2e, algorithmicx}
142 \LWR@checkloadnever{algorithmic}{algorithm2e, algorithmicx}
143 \LWR@checkloadnever{bitfield}{bytefield}
```

bxcjkatype is based on CJK:

```
144 \LWR@checkloadnever{bxcjkjatype}{upLaTeX, bxjsarticle, ujarticle, utarticle}
```

```
145 \LWR@checkloadnever{caption2}{caption}
146 % \LWR@checkloadnever{ccaption}{caption}% might be preloaded by memoir
147 \LWR@checkloadnever{colortab}{colortbl}
148 \LWR@checkloadnever{csvtools}{datatool}
149 \LWR@checkloadnever{doublespace}{setspace}
150 \LWR@checkloadnever{fancyheadings}{fancyhdr}
151 \LWR@checkloadnever{fncylab}{cleveref}
152 \LWR@checkloadnever{formula}{siunitx}
153 \LWR@checkloadnever{glossary}{glossaries}
```

hangul is not in TeXLive, and is not tested:

```
154 \LWR@checkloadnever{hangul}{kotex, xetexko, luatexko}
```

```
155 \LWR@checkloadnever{hyper}{hyperref}
156 \LWR@checkloadnever{libgreek}{libertinust1math, newtx}
157 \LWR@checkloadnever{newthm}{ntheorem}
158 \LWR@checkloadnever{pdfcprot}{microtype}
159 \LWR@checkloadnever{picins}{floatflt, wrapfig, wrapfig2}
160 \LWR@checkloadnever{rplain}{fancyhdr}
161 \LWR@checkloadnever{si}{siunitx}
162 \LWR@checkloadnever{sistyle}{siunitx}
163 \LWR@checkloadnever{slashbox}{diagbox}
164 \LWR@checkloadnever{statex}{statex2}
165 \LWR@checkloadnever{t1enc}{fontenc, inputenc, inputenx}
166 \LWR@checkloadnever{ucs}{inputenc, inputencx}
167 \LWR@checkloadnever{wasysym}{textcomp, amssymb, amsfonts, mnsymbol, fdsymbol}
```

The following may one day be supported by lwarp:

```
168 % \LWR@checkloadnever{adjustbox}{}% req'd for menukeys
169 \LWR@checkloadnever{animate}{}
170 \LWR@checkloadnever{auto-pst-pdf}{}
171 \LWR@checkloadnever{auto-pst-pdf-lua}{}
172 \LWR@checkloadnever{algorithms}{}
173 \LWR@checkloadnever{arraycols}{}
174 \LWR@checkloadnever{bidi}{}
175 \LWR@checkloadnever{cals}{}
```

```
176 \LWR@checkloadnever{cellspace}{tabls}
```

```
177 \LWR@checkloadnever{cgloss4e}{}
178 \LWR@checkloadnever{collcell}{}
179 \LWR@checkloadnever{colophon}{}
180 \LWR@checkloadnever{cooltooltips}{}
181 \LWR@checkloadnever{covington}{}
182 \LWR@checkloadnever{crbox}{}
183 \LWR@checkloadnever{decision-table}{}
184 \LWR@checkloadnever{dvgloss}{}
185 \LWR@checkloadnever{ednotes}{}
186 \LWR@checkloadnever{edfnotes}{}
187 \LWR@checkloadnever{eledform}{}
188 \LWR@checkloadnever{eledmac}{}
189 \LWR@checkloadnever{embedfile}{}
190 \LWR@checkloadnever{endnotes-hy}{endnotes}
191 \LWR@checkloadnever{expex}{}
192 \LWR@checkloadnever{fancytooltips}{}
193 \LWR@checkloadnever{fixocgx}{}
194 \LWR@checkloadnever{flowfram}{}
195 \LWR@checkloadnever{gb4e}{}
196 \LWR@checkloadnever{gmverse}{}
197 \LWR@checkloadnever{graphbox}{}
198 \LWR@checkloadnever{graphicxbox}{}
199 \LWR@checkloadnever{hvfloat}{}
200 \LWR@checkloadnever{inline-images}{}
201 \LWR@checkloadnever{isorot}{rotating}
202 \LWR@checkloadnever{ledmac}{}
203 \LWR@checkloadnever{linguex}{}
```

```
204 \LWR@checkloadnever{longdiv}{}
205 \LWR@checkloadnever{longfigure}{}
206 \LWR@checkloadnever{longtabu}{}
207 \LWR@checkloadnever{mdwenv}{}
208 \LWR@checkloadnever{mdwlist}{}
209 \LWR@checkloadnever{mdwtab}{}
210 \LWR@checkloadnever{navigator}{}
211 \LWR@checkloadnever{nccpic}{}
212 \LWR@checkloadnever{nccsect}{}
213 \LWR@checkloadnever{newvbtm}{}
214 \LWR@checkloadnever{ocg-p}{}
215 \LWR@checkloadnever{ocgtools}{}
216 \LWR@checkloadnever{ocgx}{}
217 \LWR@checkloadnever{ocgx2}{}
218 \LWR@checkloadnever{parrun}{}
219 \LWR@checkloadnever{poemscol}{}
220 \LWR@checkloadnever{poetry}{}
221 \LWR@checkloadnever{program}{}
222 \LWR@checkloadnever{proofread}{}
223 \LWR@checkloadnever{pst-pdf}{}
224 \LWR@checkloadnever{refstyle}{}
225 \LWR@checkloadnever{robustindex}{}
226 \LWR@checkloadnever{robustglossary}{}
227 \LWR@checkloadnever{semioneside}{}
228 \LWR@checkloadnever{slemph}{}
229 \LWR@checkloadnever{snotez}{sidenotes}
230 \LWR@checkloadnever{spacingtricks}{}
231 \LWR@checkloadnever{sverb}{verbatim, fancyvrb}
232 \LWR@checkloadnever{syntax}{}
233 \LWR@checkloadnever{tablists}{}
234 \LWR@checkloadnever{tabto}{}
235 \LWR@checkloadnever{tabu}{}
236 \LWR@checkloadnever{tabularht}{}
237 \LWR@checkloadnever{tabularkv}{}
238 \LWR@checkloadnever{thumby}{}
239 \LWR@checkloadnever{titles}{}
240 \LWR@checkloadnever{typehtml}{}
241 \LWR@checkloadnever{unicode-bidi}{}
242 \LWR@checkloadnever{vcell}{}
243 \LWR@checkloadnever{xhfill}{}
244 }
```

\LWR@checkloadnever {⟨*badpackagename*⟩} {⟨*replacementpkgname*⟩}

The first package is not supported, so tell the user to use the second instead.

When lwarp is first loaded, this is set to \LWR@earlyloadnever to check for incompatible packages which were loaded before lwarp. After lwarp is loaded, this is changed to \LWR@afterloadnever to check for incompatible packages during \usepackage.

```
245 \LetLtxMacro\LWR@checkloadnever\LWR@earlyloadnever
```

Now check for incompatible packages which have been loaded before lwarp:

```
246 \LWR@checkloadnevers
```

The older CJK and CJKutf8 only work with xeCJK:

```
247 \IfPackageLoadedTF{xeCJK}{}{
248    \LWR@checkloadnever{CJK}{ctex, xeCJK}
249    \LWR@checkloadnever{CJKutf8}{ctex, xeCJK}
250 }
```

Some classes do not work with lwarp:

```
251 \LWR@earlyclassloadnever{beamer}{beamerarticle}
252 \LWR@earlyclassloadnever{jarticle}{ujarticle}
253 \LWR@earlyclassloadnever{jbook}{ujbook}
254 \LWR@earlyclassloadnever{jreport}{ujreport}
255 \LWR@earlyclassloadnever{tarticle}{utarticle}
256 \LWR@earlyclassloadnever{tbook}{utbook}
257 \LWR@earlyclassloadnever{treport}{utreport}
258 \LWR@earlyclassloadnever{novel}{}
259 \LWR@earlyclassloadnever{powerdot}{}
```

22.3   Enforcing package loading after lwarp

Packages which should only be loaded after lwarp are tested here to trip an error of they have already been loaded.

The following packages must be loaded after lwarp:

```
260 \LWR@loadafter{2in1}
261 \LWR@loadafter{2up}
262 \LWR@loadafter{a4}
263 \LWR@loadafter{a4wide}
264 \LWR@loadafter{a5comb}
265 \LWR@notmemoirloadafter{abstract}
266 \LWR@loadafter{academicons}
267 \LWR@loadafter{accents}
268 \LWR@loadafter{accessibility}
269 \LWR@loadafter{accsupp}
270 \LWR@loadafter{acro}
271 \LWR@loadafter{acronym}
272 \LWR@loadafter{adjmulticol}
273 \LWR@loadafter{addlines}
274 \LWR@loadafter{afterpage}
275 \LWR@loadafter{algorithm2e}
276 \LWR@loadafter{algorithmicx}
277 \LWR@loadafter{alltt}
278 \LWR@loadafter{amscdx}
279 % \LWR@loadafter{amsmath}% may be preloaded
280 % \LWR@loadafter{amsthm}% may be preloaded
281 \LWR@loadafter{anonchap}
282 \LWR@loadafter{anysize}
283 \LWR@notmemoirloadafter{appendix}
284 \LWR@loadafter{apxproof}
285 \LWR@loadafter{ar}
286 \LWR@loadafter{arabicfront}
287 \LWR@notmemoirloadafter{array}
288 \LWR@loadafter{arydshln}
289 \LWR@loadafter{asymptote}
290 % \LWR@loadafter{atbegshi}% now in LaTeX core, also used by morewrites
291 \LWR@loadafter{attachfile}
```

```
292 \LWR@loadafter{attachfile2}
293 \LWR@loadafter{authblk}
294 \LWR@loadafter{authoraftertitle}% Supported as-is, but must be loaded after.
295 \LWR@loadafter{autobreak}
296 \LWR@loadafter{autonum}
297 \LWR@loadafter{awesomebox}
298 \LWR@loadafter{axessibility}
299 \LWR@loadafter{axodraw2}
300 \LWR@loadafter{backnaur}
301 \LWR@loadafter{backref}
302 \LWR@loadafter{balance}
303 \LWR@loadafter{bbding}
304 \LWR@loadafter{beamerarticle}
305 \LWR@loadafter{bigdelim}
306 \LWR@loadafter{bigfoot}
307 \LWR@loadafter{bigstrut}
308 \LWR@loadafter{bitpattern}
309 \LWR@loadafter{blowup}
310 \LWR@loadafter{bm}
311 \LWR@loadafter{booklet}
312 \LWR@loadafter{bookmark}
313 \LWR@notmemoirloadafter{booktabs}
314 \LWR@loadafter{bophook}
315 \LWR@loadafter{bounddvi}
316 \LWR@loadafter{boxedminipage}
317 \LWR@loadafter{boxedminipage2e}
318 \LWR@loadafter{braket}
319 \LWR@loadafter{breakurl}
320 \LWR@loadafter{breqn}
321 \LWR@loadafter{bsheaders}
322 \LWR@loadafter{bussproofs}
323 \LWR@loadafter{bxpapersize}
324 \LWR@loadafter{bytefield}
325 \LWR@loadafter{ccicons}
326 \LWR@loadafter{cancel}
327 \LWR@loadafter{canoniclayout}
328 \LWR@loadafter{caption}
329 \LWR@loadafter{caption2}
330 \LWR@loadafter{caption3}
331 \LWR@loadafter{cases}
332 % catoptions is supported by the lwarp core
333 % \LWR@loadafter{ccaption}% may be preloaded by memoir
334 \LWR@loadafter{centerlastline}
335 % \LWR@loadafter{centernot}% may be preloaded by newtx
336 \LWR@loadafter{changebar}
337 \LWR@loadafter{changelayout}
338 \LWR@notmemoirloadafter{changepage}
339 \LWR@loadafter{changes}
340 \LWR@loadafter{chappg}
341 \LWR@loadafter{chapterbib}
342 \LWR@loadafter{chemfig}
343 \LWR@loadafter{chemformula}
344 \LWR@loadafter{chemgreek}
345 \LWR@loadafter{chemmacros}
346 \LWR@loadafter{chemnum}
347 \LWR@loadafter{chkfloat}
348 \LWR@notmemoirloadafter{chngpage}
349 \LWR@loadafter{cite}
350 \LWR@loadafter{citeref}
351 \LWR@loadafter{classicthesis}
```

```
352 \LWR@loadafter{cleveref}
353 % cmbright may be preloaded
354 \LWR@loadafter{cmdtrack}
355 \LWR@loadafter{colonequals}
356 \LWR@loadafter{color}
357 \LWR@loadafter{colortbl}
358 \LWR@loadafter{continue}
359 \LWR@loadafter{copyrightbox}
360 \LWR@notmemoirloadafter{crop}
361 % ctex must be loaded before lwarp
362 \LWR@loadafter{ctable}
363 \LWR@loadafter{cuted}
364 \LWR@loadafter{cutwin}
365 \LWR@loadafter{dblfloatfix}
366 \LWR@loadafter{dblfnote}
367 \LWR@notmemoirloadafter{dcolumn}
368 \LWR@loadafter{decimal}
369 \LWR@loadafter{decorule}
370 \LWR@loadafter{diagbox}
371 \LWR@loadafter{dingbat}
372 \LWR@loadafter{doipubmed}
373 \LWR@loadafter{DotArrow}
374 \LWR@loadafter{dotlessi}
375 \LWR@loadafter{dprogress}
376 \LWR@loadafter{draftcopy}
377 \LWR@loadafter{draftfigure}
378 \LWR@loadafter{draftwatermark}
379 \LWR@loadafter{drftcite}
380 \LWR@loadafter{easy-todo}
381 \LWR@loadafter{ebook}
382 \LWR@loadafter{econometrics}
383 \LWR@loadafter{ed}
384 \LWR@loadafter{ellipsis}
385 \LWR@loadafter{embrac}
386 \LWR@loadafter{emptypage}
387 \LWR@loadafter{endfloat}
388 \LWR@loadafter{endheads}
389 \LWR@loadafter{endnotes}
390 \LWR@loadafter{engtlc}
391 \LWR@loadafter{enotez}
392 \LWR@notmemoirloadafter{enumerate}
393 \LWR@loadafter{enumitem}
394 \LWR@notmemoirloadafter{epigraph}
395 \LWR@loadafter{epsf}
396 \LWR@loadafter{epsfig}
397 \LWR@loadafter{epstopdf}
398 \LWR@loadafter{epstopdf-base}
399 \LWR@loadafter{eqlist}
400 \LWR@loadafter{eqparbox}
401 \LWR@loadafter{errata}
402 \LWR@loadafter{eso-pic}
403 \LWR@loadafter{esvect}
404 \LWR@loadafter{etoc}
405 \LWR@loadafter{eurosym}
406 \LWR@loadafter{everypage}
407 % \LWR@loadafter{everyshi}% now in LaTeX core
408 \LWR@loadafter{extarrows}
409 \LWR@loadafter{extramarks}
410 \LWR@loadafter{fancybox}
411 \LWR@loadafter{fancyhdr}
```

```
412 \LWR@loadafter{fancypar}
413 \LWR@loadafter{fancyref}
414 \LWR@loadafter{fancytabs}
415 \LWR@loadafter{fancyvrb}
416 \LWR@loadafter{fbox}
417 \LWR@loadafter{fewerfloatpages}
418 \LWR@loadafter{figcaps}
419 \LWR@loadafter{figsize}
420 \LWR@loadafter{fitbox}
421 \LWR@loadafter{fix2col}
422 \LWR@loadafter{fixmath}
423 \LWR@loadafter{fixme}
424 \LWR@loadafter{fixmetodonotes}
425 \LWR@loadafter{flafter}
426 \LWR@loadafter{flippdf}
427 \LWR@loadafter{float}
428 \LWR@loadafter{floatflt}
429 \LWR@loadafter{floatpag}
430 \LWR@loadafter{floatrow}
431 \LWR@loadafter{fltrace}
432 \LWR@loadafter{flushend}
433 \LWR@loadafter{fnbreak}
434 \LWR@loadafter{fncychap}
435 \LWR@loadafter{fnlineno}
436 \LWR@loadafter{fnpara}
437 \LWR@loadafter{fnpos}
438 \LWR@loadafter{fontawesome}
439 \LWR@loadafter{fontawesome5}
440 % fontenc must be loaded before lwarp
441 % fontspec must be loaded before lwarp
442 \LWR@loadafter{footmisc}
443 \LWR@loadafter{footnote}
444 \LWR@loadafter{footnotebackref}
445 \LWR@loadafter{footnotehyper}
446 \LWR@loadafter{footnoterange}
447 \LWR@loadafter{footnpag}
448 \LWR@loadafter{foreign}
449 \LWR@loadafter{forest}
450 \LWR@loadafter{fouridx}
451 % fourier may be loaded before lwarp
452 \LWR@loadafter{framed}
453 \LWR@loadafter{froufrou}
454 \LWR@loadafter{ftcap}
455 \LWR@loadafter{ftnright}
456 \LWR@loadafter{fullminipage}
457 \LWR@loadafter{fullpage}
458 \LWR@loadafter{fullwidth}
459 \LWR@loadafter{fvextra}
460 \LWR@loadafter{fwlw}
461 \LWR@loadafter{gensymb}
462 \LWR@loadafter{gentombow}
463 % geometry is always loaded by lwarp, and lwarp-geometry is AtBeginDocument
464 \LWR@loadafter{ghsystem}
465 \LWR@loadafter{gindex}
466 \LWR@loadafter{glossaries}
467 \LWR@loadafter{gmeometric}
468 % \LWR@loadafter{graphics}% pre-loaded by xunicode
469 % \LWR@loadafter{graphicx}% pre-loaded by xunicode
470 \LWR@loadafter{gloss}
471 \LWR@loadafter{glossary}
```

```
472 \LWR@loadafter{grffile}
473 \LWR@loadafter{grid}
474 \LWR@loadafter{grid-system}
475 \LWR@loadafter{gridset}
476 \LWR@loadafter{hang}
477 \LWR@loadafter{hanging}
478 \LWR@loadafter{hepunits}
479 \LWR@loadafter{hhline}
480 \LWR@loadafter{hhtensor}
481 \LWR@loadafter{hypbmsec}
482 \LWR@loadafter{hypcap}
483 \LWR@loadafter{hypdestopt}
484 \LWR@loadafter{hypernat}
485 \LWR@loadafter{hyperref}
486 \LWR@loadafter{hyperxmp}
487 \LWR@loadafter{hyphenat}
488 \LWR@loadafter{idxlayout}
489 \LWR@loadafter{ifoddpage}
490 \LWR@loadafter{imakeidx}
491 \LWR@loadafter{impnattypo}
492 \LWR@notmemoirloadafter{index}
493 % inputenc must be loaded before lwarp
494 % inputenx must be loaded before lwarp
495 % inputtrc may be loaded before lwarp
496 \LWR@loadafter{intopdf}
497 \LWR@loadafter{isomath}
498 \LWR@loadafter{isotope}
499 \LWR@loadafter{jurabib}
500 \LWR@loadafter{karnaugh-map}
501 \LWR@loadafter{keyfloat}
502 \LWR@loadafter{keystroke}
503 % kpfonts may be loaded before lwarp
504 % kpfonts-otf may be loaded before lwarp
505 \LWR@loadafter{layaureo}
506 \LWR@loadafter{layout}
507 \LWR@loadafter{layouts}
508 \LWR@loadafter{leading}
509 \LWR@loadafter{leftidx}
510 \LWR@loadafter{letterspace}
511 \LWR@loadafter{lettrine}
512 % libertinust1math may be loaded before lwarp
513 \LWR@loadafter{lineno}
514 \LWR@loadafter{lips}
515 \LWR@loadafter{listings}
516 \LWR@loadafter{listliketab}
517 \LWR@loadafter{lltjp-siunitx}
518 \LWR@loadafter{lltjp-tascmac}
519 \LWR@loadafter{longtable}
520 \LWR@loadafter{lpic}
521 \LWR@loadafter{lscape}
522 \LWR@loadafter{ltablex}
523 \LWR@loadafter{ltcaption}
524 \LWR@loadafter{ltxgrid}
525 \LWR@loadafter{ltxtable}
526 \LWR@loadafter{lua-check-hyphen}
527 \LWR@loadafter{lua-visual-debug}
528 \LWR@loadafter{luacolor}
529 \LWR@loadafter{luamplib}
530 \LWR@loadafter{luatodonotes}
531 \LWR@loadafter{luavlna}
```

```
532 \LWR@loadafter{lyluatex}
533 \LWR@loadafter{magaz}
534 \LWR@notmemoirloadafter{makeidx}
535 \LWR@loadafter{manyfoot}
536 \LWR@loadafter{marginfit}
537 \LWR@loadafter{marginfix}
538 \LWR@loadafter{marginnote}
539 \LWR@loadafter{marvosym}
540 % mathalpha may be loaded before lwarp
541 \LWR@loadafter{mathastext}
542 \LWR@loadafter{mathcomp}
543 \LWR@loadafter{mathdesign}
544 \LWR@loadafter{mathdots}
545 \LWR@loadafter{mathfixs}
546 \LWR@loadafter{mathpazo}
547 \LWR@loadafter{mathptmx}
548 \LWR@loadafter{mathspec}
549 \LWR@loadafter{mathtools}
550 \LWR@loadafter{mattens}
551 \LWR@loadafter{maybemath}
552 \LWR@loadafter{mcaption}
553 \LWR@loadafter{mdframed}
554 \LWR@loadafter{mdwmath}
555 \LWR@loadafter{media9}
556 \LWR@loadafter{memhfixc}
557 \LWR@loadafter{menukeys}
558 \LWR@loadafter{metalogo}
559 \LWR@loadafter{metalogox}
560 \LWR@loadafter{mhchem}
561 \LWR@loadafter{microtype}
562 \LWR@loadafter{midfloat}
563 \LWR@loadafter{midpage}
564 \LWR@loadafter{minibox}
565 \LWR@loadafter{minitoc}
566 \LWR@loadafter{minted}
567 \LWR@loadafter{mismath}
568 \LWR@loadafter{mleftright}
569 % morefloats must be allowed early for print mode
570 \LWR@notmemoirloadafter{moreverb}
571 % morewrites must be loaded before lwarp
572 \LWR@notmemoirloadafter{movie15}
573 \LWR@notmemoirloadafter{mparhack}
574 \LWR@loadafter{multibib}
575 \LWR@loadafter{multicap}
576 %\LWR@loadafter{multicol}% loaded by ltxdoc
577 \LWR@loadafter{multicolrule}
578 \LWR@loadafter{multimedia}
579 \LWR@loadafter{multiobjective}
580 \LWR@loadafter{multirow}
581 \LWR@loadafter{multitoc}
582 \LWR@loadafter{musicography}
583 \LWR@loadafter{mwe}
584 \LWR@loadafter{nameauth}
585 \LWR@loadafter{natbib}
586 \LWR@notmemoirloadafter{nccfancyhdr}
587 \LWR@loadafter{nccfoots}
588 \LWR@loadafter{nccmath}
589 \LWR@notmemoirloadafter{needspace}
590 % newclude must be loaded before lwarp
591 % newpxmath may be preloaded
```

592 % newtxmath may be loaded before lwarp
593 % newtxsf may be loaded before lwarp
594 % newunicodechar must be loaded before lwarp
595 \LWR@notmemoirloadafter{nextpage}
596 \LWR@loadafter{nicefrac}
597 \LWR@loadafter{niceframe}
598 \LWR@loadafter{nicematrix}
599 \LWR@loadafter{noitcrul}
600 \LWR@loadafter{nolbreaks}
601 \LWR@loadafter{nomencl}
602 \LWR@loadafter{nonfloat}
603 \LWR@loadafter{nonumonpart}
604 \LWR@loadafter{nopageno}
605 \LWR@loadafter{notes}
606 \LWR@loadafter{notespages}
607 \LWR@loadafter{nowidow}
608 \LWR@loadafter{ntheorem}
609 \LWR@loadafter{octave}
610 \LWR@loadafter{orcidlink}
611 \LWR@loadafter{overpic}
612 \LWR@loadafter{pagegrid}
613 \LWR@notmemoirloadafter{pagenote}
614 \LWR@loadafter{pagesel}
615 \LWR@loadafter{paralist}
616 \LWR@loadafter{parallel}
617 \LWR@loadafter{parcolumns}
618 \LWR@loadafter{parnotes}
619 \LWR@notmemoirloadafter{parskip}
620 \LWR@loadafter{pbalance}
621 \LWR@loadafter{pbox}
622 \LWR@loadafter{pdfcol}
623 \LWR@loadafter{pdfcolfoot}
624 \LWR@loadafter{pdfcolmk}
625 \LWR@loadafter{pdfcolparallel}
626 \LWR@loadafter{pdfcolparcolumns}
627 \LWR@loadafter{pdfcomment}
628 \LWR@loadafter{pdfcrypt}
629 \LWR@loadafter{pdflscape}
630 \LWR@loadafter{pdfmarginpar}
631 \LWR@loadafter{pdfpages}
632 \LWR@loadafter{pdfprivacy}
633 \LWR@loadafter{pdfrender}
634 \LWR@loadafter{pdfsync}
635 \LWR@loadafter{pdftricks}
636 \LWR@loadafter{pdfx}
637 \LWR@loadafter{perpage}
638 \LWR@loadafter{pfnote}
639 \LWR@loadafter{phfqit}
640 \LWR@loadafter{physics}
641 \LWR@loadafter{physunits}
642 \LWR@loadafter{picinpar}
643 \LWR@loadafter{pifont}
644 \LWR@loadafter{pinlabel}
645 \LWR@loadafter{placeins}
646 \LWR@loadafter{plarray}
647 \LWR@loadafter{plarydshln}
648 \LWR@loadafter{plextarray}
649 \LWR@loadafter{plextarydshln}
650 \LWR@loadafter{plcolortbl}
651 \LWR@loadafter{plextdelarray}

```
652 \LWR@loadafter{plimsoll}
653 \LWR@loadafter{prelim2e}
654 \LWR@loadafter{prettyref}
655 \LWR@loadafter{preview}
656 \LWR@loadafter{psfrag}
657 \LWR@loadafter{psfragx}
658 \LWR@loadafter{pst-eps}
659 \LWR@loadafter{pstool}
660 \LWR@loadafter{pstricks}
661 % \LWR@loadafter{pxatbegshi}% may be used by morewrites
662 \LWR@loadafter{pxeveryshi}
663 % \LWR@loadafter{pxfonts}% may be loaded before lwarp
664 \LWR@loadafter{pxftnright}
665 \LWR@loadafter{pxjahyper}
666 \LWR@loadafter{quotchap}
667 \LWR@loadafter{quoting}
668 \LWR@loadafter{ragged2e}
669 \LWR@loadafter{refcheck}
670 \LWR@loadafter{register}
671 \LWR@loadafter{relsize}
672 \LWR@loadafter{repeatindex}
673 \LWR@loadafter{resizegather}
674 \LWR@loadafter{returntogrid}
675 \LWR@loadafter{rlepsf}
676 \LWR@loadafter{rmathbr}
677 \LWR@loadafter{rmpage}
678 \LWR@loadafter{romanbar}
679 \LWR@loadafter{romanbarpagenumber}
680 \LWR@loadafter{rotating}
681 \LWR@loadafter{rotfloat}
682 \LWR@loadafter{rviewport}
683 \LWR@loadafter{savetrees}
684 % scalefnt is loaded by babel-french
685 \LWR@loadafter{scalerel}
686 \LWR@loadafter{schemata}
687 \LWR@loadafter{scrextend}
688 \LWR@loadafter{scrhack}
689 \LWR@loadafter{scrlayer}
690 \LWR@loadafter{scrlayer-notecolumn}
691 \LWR@loadafter{scrlayer-scrpage}
692 \LWR@loadafter{scrpage2}
693 \LWR@loadafter{section}
694 \LWR@loadafter{sectionbreak}
695 \LWR@loadafter{sectsty}
696 \LWR@loadafter{selectp}
697 \LWR@loadafter{semantic-markup}
698 \LWR@notmemoirloadafter{setspace}
699 \LWR@loadafter{shadow}
700 \LWR@loadafter{shapepar}
701 \LWR@notmemoirloadafter{showidx}
702 \LWR@loadafter{showlabels}
703 \LWR@loadafter{showkeys}
704 \LWR@loadafter{showtags}
705 \LWR@loadafter{shuffle}
706 \LWR@loadafter{sidecap}
707 \LWR@loadafter{sidenotes}
708 \LWR@loadafter{simplebnf}
709 \LWR@loadafter{SIunits}
710 \LWR@loadafter{siunitx}
711 \LWR@loadafter{siunitx-v2}
```

```
712 \LWR@loadafter{skmath}
713 \LWR@loadafter{slantsc}
714 \LWR@loadafter{slashed}
715 \LWR@loadafter{soul}
716 \LWR@loadafter{soulpos}
717 \LWR@loadafter{soulutf8}
718 \LWR@loadafter{splitbib}
719 \LWR@loadafter{splitidx}
720 \LWR@loadafter{srcltx}
721 \LWR@loadafter{srctex}
722 \LWR@loadafter{stabular}
723 \LWR@loadafter{stackengine}
724 \LWR@loadafter{stackrel}
725 \LWR@loadafter{statex2}
726 \LWR@loadafter{statistics}
727 \LWR@loadafter{statmath}
728 \LWR@loadafter{steinmetz}
729 \LWR@notltjloadafter{stfloats}
730 \LWR@loadafter{struktex}
731 \LWR@loadafter{subcaption}
732 \LWR@loadafter{subfig}
733 \LWR@loadafter{subfigure}
734 \LWR@loadafter{subsupscripts}
735 \LWR@loadafter{supertabular}
736 \LWR@loadafter{svg}
737 \LWR@loadafter{swfigure}
738 \LWR@loadafter{sympytex}
739 \LWR@loadafter{syntonly}
740 \LWR@loadafter{t1inc}
741 \LWR@loadafter{tabfigures}
742 \LWR@loadafter{tabls}
743 \LWR@loadafter{tablefootnote}
744 \LWR@notmemoirloadafter{tabularx}
745 \LWR@loadafter{tabulary}
746 %\LWR@loadafter{tagpdf}% no longer true
747 %\LWR@loadafter{tagpdf-mc-code-generic}
748 %\LWR@loadafter{tagpdf-mc-code-lua}
749 \LWR@loadafter{tascmac}
750 \LWR@loadafter{tcolorbox}
751 \LWR@loadafter{tensor}
752 \LWR@loadafter{termcal}
753 \LWR@loadafter{textarea}
754 % \LWR@loadafter{textcomp}% maybe before lwarp with font packages
755 \LWR@loadafter{textfit}
756 \LWR@loadafter{textpos}
757 \LWR@loadafter{theorem}
758 \LWR@loadafter{thinsp}
759 \LWR@loadafter{thm-listof}
760 \LWR@loadafter{thm-restate}
761 \LWR@loadafter{thmbox}
762 \LWR@loadafter{thmtools}
763 \LWR@loadafter{threadcol}
764 \LWR@loadafter{threeparttable}
765 \LWR@loadafter{threeparttablex}
766 \LWR@loadafter{thumb}
767 \LWR@loadafter{thumbs}
768 \LWR@loadafter{tikz}
769 \LWR@loadafter{tikz-imagelabels}
770 \LWR@loadafter{titleps}
771 \LWR@loadafter{titlesec}
```

```
772 \LWR@loadafter{titletoc}
773 \LWR@notmemoirloadafter{titling}
774 % \LWR@loadafter{tocbasic}% preloaded by koma-script classes
775 \LWR@notmemoirloadafter{tocbibind}
776 \LWR@loadafter{tocdata}
777 \LWR@loadafter{tocenter}
778 \LWR@notmemoirloadafter{tocloft}
779 \LWR@loadafter{tocstyle}
780 \LWR@loadafter{todo}
781 \LWR@loadafter{todonotes}
782 \LWR@loadafter{topcapt}
783 \LWR@loadafter{tram}
784 \LWR@loadafter{transparent}
785 \LWR@loadafter{trimclip}
786 \LWR@loadafter{trivfloat}
787 \LWR@loadafter{truncate}
788 \LWR@loadafter{turnthepage}
789 \LWR@loadafter{twoup}
790 % \LWR@loadafter{txfonts}% may be loaded before lwarp
791 % txgreeks may be loaded before lwarp

792 % \LWR@loadafter{typearea}% preloaded by koma-script classes
793 \LWR@loadafter{typicons}
794 % \LWR@loadafter{ulem}% preloaded by ctexart and related classes
795 \LWR@loadafter{umoline}
796 \LWR@loadafter{underscore}
797 % unicode-math may be loaded before lwarp
798 \LWR@loadafter{units}
799 \LWR@loadafter{unitsdef}
800 \LWR@loadafter{upgreek}
801 \LWR@loadafter{upref}
802 \LWR@loadafter{url}
803 \LWR@loadafter{ushort}
804 \LWR@loadafter{uspace}
805 \LWR@loadafter{varioref}
806 \LWR@notmemoirloadafter{verse}
807 \LWR@loadafter{versonotes}
808 \LWR@loadafter{vertbars}
809 \LWR@loadafter{vmargin}
810 \LWR@loadafter{vowel}
811 \LWR@loadafter{vpe}
812 \LWR@loadafter{vwcol}
813 \LWR@loadafter{wallpaper}
814 \LWR@loadafter{watermark}
815 \LWR@loadafter{widetable}
816 \LWR@loadafter{widows-and-orphans}
817 \LWR@loadafter{witharrows}
818 \LWR@loadafter{wrapfig}
819 \LWR@loadafter{wrapfig2}
820 \LWR@loadafter{xbmks}
821 \LWR@loadafter{xcolor}
822 \LWR@loadafter{xechangebar}
823 \LWR@loadafter{xellipsis}
824 % xetexko must be loaded before lwarp
825 \LWR@loadafter{xevlna}
826 \LWR@loadafter{xfakebold}
827 \LWR@loadafter{xfrac}
828 \LWR@loadafter{xltabular}
829 \LWR@loadafter{xltxtra}
830 \LWR@loadafter{xmpincl}
```

```
831 \LWR@loadafter{xpiano}
832 \LWR@loadafter{xpinyin}
833 \LWR@loadafter{xr}
834 \LWR@loadafter{xr-hyper}
835 \LWR@loadafter{xtab}
836 % xunicode must be loaded before lwarp
837 \LWR@loadafter{xurl}
838 \LWR@loadafter{xy}
839 \LWR@loadafter{zwpagelayout}
```

23   MD5 hashing

The MD5 hash is used for lateximage filenames for svg math.

```
840 \newcommand{\LWR@mdfive}[1]{%
841     \PackageError{lwarp}
842         {No MD5 macro was found}
843         {%
844             Lwarp must find the macros \protect\pdfmdfivesum\space
845             or \protect\mdfivesum.%
846         }
847 }
```

The default for PDF LATEX, DVI LATEX, upLATEX, etc:

```
848 \ifdef{\pdfmdfivesum}
849     {\let\LWR@mdfive\pdfmdfivesum}
850     {}
```

For LuaLATEX:

```
851 \ifLuaTeX
852 \RequirePackage{pdftexcmds}
853 \let\LWR@mdfive\pdf@mdfivesum
854 \fi
```

For XƎLATEX:

```
855 \ifXeTeX
856 \@ifundefined{pdffivesum}{}
857     {\let\LWR@mdfive\pdfmdfivesum}
858 \@ifundefined{mdfivesum}{}
859     {\let\LWR@mdfive\mdfivesum}
860 \fi
```

24   PDF LATEX T1 and UTF-8 encoding

When using PDF LATEX, lwarp requires T1 font encoding, and recommends UTF-8 input encoding.

If some other input encoding is already defined, lwarp will try to use it instead, and hope for the best.

XƎLATEX and LuaLATEX are both UTF-8 by nature.

`\LWR@pdfencoding` Sets T1, and also utf8 if not already set.

```
861 \newcommand*{\LWR@pdfencoding}{%
862     \RequirePackage[T1]{fontenc}
863
864     \IfPackageLoadedTF{inputenc}{}{
865         \IfPackageLoadedTF{inputenx}{}{
866             \RequirePackage[utf8]{inputenc}
867         }
868     }
869 }
```

```
870 \ifPDFTeX% pdflatex or dvi latex
871     \LWR@pdfencoding
872 \fi
873
874 \ifpTeX
875     \LWR@pdfencoding
876 \fi
```

25   Unicode input characters

If using *pdflatex*, convert a minimal set of Unicode characters. Additional characters may be defined by the user, as needed.

A commonly-used multiply symbol is declared to be `\texttimes`.

The first arguments of `\newunicodechar` below are text ligatures in the source code, even though they are not printed in the following listing.

```
877 \ifpTeX
878 \else
879 \RequirePackage{newunicodechar}
880
881 \newunicodechar{×}{\texttimes}
882
883 \ifPDFTeX% pdflatex or dvi latex
884 \newunicodechar{ff}{ff}% Here, the first arguments are ligatures.
885 \newunicodechar{fi}{fi}
886 \newunicodechar{fl}{fl}
887 \newunicodechar{ffi}{ffi}
888 \newunicodechar{ffl}{ffl}
889 \newunicodechar{–}{---}
890 \newunicodechar{-}{--}
891 \fi
892
893 \fi
```

26   Avoid a bitmapped font

If DVI or PDF LATEX, and if the default Computer Modern is the selected font family, ensure that cm-super or lmodern is used to provide a vector font.

```
894 \ifxetexorluatex
```

```
895 \else
896     \ifdefstring{\f@family}{cmr}{
897         \IfFileExists{type1ec.sty}% found in cm-super
898         {}
899         {% cm-super not installed
900             \IfFileExists{lmodern.sty}{
901                 \PackageInfo{lwarp}{cm-super not installed, loading lmodern}
902                 \RequirePackage{lmodern}
903             }{
904                 \PackageError{lwarp}
905                 {%
906                     Lwarp requires a vector font.\MessageBreak
907                  Install and load cm-super, lmodern, or another\MessageBreak
908                     Type-1 vector font before loading lwarp.\MessageBreak
909                     Enter 'H' for possible solutions%
910                 }
911                 {%
912                     Install cm-super or lmodern.\MessageBreak
913                     If lmodern, load it before lwarp:\MessageBreak
914                         \space\space\protect\usepackage{lmodern}\MessageBreak
915                         \space\space\protect\usepackage{lwarp}%
916                 }
917             }
918         }% cm-super not installed
919     }{}% f@family
920 \fi
```

27 Upright quotes

In PDF TEX, preserve upright quotes in verbatim text. upquote also loads textcomp.

```
921 \ifPDFTeX
922 \RequirePackage{upquote}
923 \fi
924
925 \ifpTeX
926     \RequirePackage{upquote}
927 \fi
```

28 Avoid bad font combinations

For XƎLᴬTEX and LuaLᴬTEX, certain font combinations cause problems with lwarp.

libertinus-otf has special handling for \textquotedbl. Search for \LWR@orig@textquotedbl.

```
928 \ifxetexorluatex
929     \AtBeginDocument{
930         \IfPackageLoadedTF{kpfonts}{
931             \PackageError{lwarp}
932             {%
933                 When using XeLaTeX or LuaLaTeX,\MessageBreak
934                 use kpfonts-otf instead of kpfonts%
935             }
936             {%
```

```
937                    Replace: \protect\usepackage{kpfonts}\MessageBreak
938                    with: \protect\usepackage{kpfonts-otf}
939            }
940        }{}
941    }
942 \fi
```

29 Miscellaneous tools

29.1 Variables

```
943 \newlength{\LWR@templengthone}
944 \newlength{\LWR@templengthtwo}
945 \newlength{\LWR@templengththree}
946 \newcounter{LWR@tempcountone}
```

29.2 Lengths and units

\LWR@providelength {⟨\lengthname⟩} Provides the length if it isn't defined yet.

Used to provide source compatibility for lengths which will be ignored, but might or might not be already provided by other packages.

```
947 \newcommand*\LWR@providelength[1]{%
948     \ifdeflength{#1}{}{\newlength{#1}}%
949 }
```

\LWR@convertto {⟨*dest unit*⟩} {⟨*length*⟩}

Prints a length in the given units, without printing the unit itself.

```
950 \newcommand*{\LWR@convertto}[2]{\strip@pt\dimexpr #2*65536/\number\dimexpr 1#1}
```

\LWR@printpercentlength {⟨*smaller*⟩} {⟨*larger*⟩}

Prints a percent ratio of the two lengths.

```
951 \newcommand*{\LWR@printpercentlength}[2]{%
952     \setcounter{LWR@tempcountone}{100*\ratio{#1}{#2}}%
953     \arabic{LWR@tempcountone}%
954 }
```

29.3 Counters

\defaddtocounter {⟨*name*⟩} {⟨*value*⟩}

Locally add to a counter.

```
955 \providecommand*{\defaddtocounter}[2]{%
956     \defcounter{#1}{\value{#1}+#2}%
957 }
```

29.4   Patching macros

\LWR@patcherror {⟨*packagename*⟩} {⟨*macroname*⟩}

Prints an error if could not patch a macro.

```
958 \newcommand*{\LWR@patcherror}[2]{%
959     \PackageError{lwarp}%
960         {%
961             Unable to patch package #1,\MessageBreak
962             macro \LWRbackslash #2.\MessageBreak
963             Lwarp or #1 may need to be updated%
964         }%
965         {Please contact the maintainer of the Lwarp package.}%
966 }
```

29.5   Copying macros

\csNewCommandCopycs {⟨*dest csname*⟩} {⟨*source csname*⟩}

Given a cs-name for each, copies a macro to a new definition.

```
967 \providecommand*{\csNewCommandCopycs}[2]{%
968     \expandafter\NewCommandCopy\csname#1\expandafter\endcsname%
969         \csname#2\endcsname%
970 }
```

\NewEnvironmentCopy {⟨*dest*⟩} {⟨*source*⟩}

Copies an environment to a new definition.

```
971 \providecommand*{\NewEnvironmentCopy}[2]{%
972     \csNewCommandCopycs{#1}{#2}%
973     \csNewCommandCopycs{end#1}{end#2}%
974 }
```

29.6   Chinese text isolation

\LWR@isolate {⟨*text*⟩} Isolates Chinese characters from the surrounding text. This is required
to avoid extra spaces on either side of the Chinese characters, especially when
written to a file.

```
975 \newcommand{\LWR@isolate}[1]{#1}%
976
977 \IfPackageLoadedTF{ctexpatch}{
978     \renewcommand{\LWR@isolate}[1]{\null#1\null}%
979 }{}
980
981 \IfPackageLoadedTF{xeCJK}{
982     \renewcommand{\LWR@isolate}[1]{\null#1\null}%
983 }{}
```

\LWR@disablepinyin Disable xpinyin during file, sideᴛᴏᴄ, and footnote generation. Set by xpinyin.

```
984 \newcommand*{\LWR@disablepinyin}{}
```

29.7   Inserting vertical space

\LWR@forceemptyline Extra vertical space in the HTML output. Use after \LWR@stoppars.

```
985 \newcommand*{\LWR@forceemptyline}{%
986     \LWR@origrule{0pt}{1\baselineskip}%
987     \LWR@orignewline%
988 }
```

29.8   Argument selection

\LWR@thirdofthree {⟨*first*⟩} {⟨*second*⟩} {⟨*third*⟩}

\LWR@fourthoffour {⟨*first*⟩} {⟨*second*⟩} {⟨*third*⟩} {⟨*fourth*⟩}

\LWR@firstoffive {⟨*first*⟩} {⟨*second*⟩} {⟨*third*⟩} {⟨*fourth*⟩} {⟨*fifth*⟩}

\LWR@secondoffive {⟨*first*⟩} {⟨*second*⟩} {⟨*third*⟩} {⟨*fourth*⟩} {⟨*fifth*⟩}

\LWR@thirdoffive {⟨*first*⟩} {⟨*second*⟩} {⟨*third*⟩} {⟨*fourth*⟩} {⟨*fifth*⟩}

\LWR@fourthoffive {⟨*first*⟩} {⟨*second*⟩} {⟨*third*⟩} {⟨*fourth*⟩} {⟨*fifth*⟩}

\LWR@fifthoffive {⟨*first*⟩} {⟨*second*⟩} {⟨*third*⟩} {⟨*fourth*⟩} {⟨*fifth*⟩}

Expands to the nth of the five arguments. Used for extra cross referencing.

```
989 \long\def\LWR@thirdofthree#1#2#3{#3}%
990 \long\def\LWR@fourthoffour#1#2#3#4{#4}%
991
992 \long\def\LWR@firstoffive#1#2#3#4#5{#1}
993 \long\def\LWR@secondoffive#1#2#3#4#5{#2}
994 \long\def\LWR@thirdoffive#1#2#3#4#5{#3}
995 \long\def\LWR@fourthoffive#1#2#3#4#5{#4}
996 \long\def\LWR@fifthoffive#1#2#3#4#5{#5}
```

\LWR@edeffirstoffive {⟨*first*⟩} {⟨*second*⟩} {⟨*third*⟩} {⟨*fourth*⟩} {⟨*fifth*⟩} \edefs to the first of five arguments. Used for back referencing.

```
997 \long\def\LWR@edeffirstoffive#1#2#3#4#5{%
998     \edef\@tempa{#1}%
999 }%
```

29.9   Inside boxes

Greater than zero if currently inside a TEX box, thus should not use \LWR@orignewpage. See section 13.2.

```
1000 \newcounter{LWR@texboxdepth}
1001 \setcounter{LWR@texboxdepth}{0}
```

\LWR@maybe@orignewpage   Only do \LWR@orignewpage if not inside a TEX box. Avoids nested paragraph tags.

```
1002 \newcommand*{\LWR@maybe@orignewpage}{%
1003     \LWR@traceinfo{LWR@maybe@orignewpage}%
1004     \ifnumgreater{\value{LWR@texboxdepth}}{0}%
1005         {}%
1006         {\LWR@orignewpage}%
1007     \LWR@traceinfo{LWR@maybe@orignewpage done}%
1008 }
```

29.10   Global boxes

\LWR@gsavebox   {⟨*macroname*⟩} {⟨*contents*⟩}

From https://tex.stackexchange.com/questions/288702/
savebox-forgets-its-content-across-columns-inside-align

```
1009 \DeclareRobustCommand\LWR@gsavebox[1]{%
1010   \@ifnextchar(%
1011     {\LWR@@gsavepicbox#1}{\@ifnextchar[{\LWR@@gsavebox#1}{\LWR@gsbox#1}}}%
1012 \long\def\LWR@gsbox#1#2{\global\setbox#1\hbox{%
1013   \color@setgroup#2\color@endgroup}}
1014 \def\LWR@@gsavebox#1[#2]{%
1015   \@ifnextchar [{\LWR@@igsavebox#1[#2]}{\LWR@@igsavebox#1[#2][c]}}
1016 \long\def\LWR@@igsavebox#1[#2][#3]#4{%
1017   \LWR@gsbox#1{\@imakebox[#2][#3]{#4}}}
1018 \def\LWR@@gsavepicbox#1(#2,#3){%
1019   \@ifnextchar[%
1020     {\LWR@@igsavepicbox#1(#2,#3)}{\LWR@@igsavepicbox#1(#2,#3)[]}}
1021 \long\def\LWR@@igsavepicbox#1(#2,#3)[#4]#5{%
1022   \LWR@gsbox#1{\@imakepicbox(#2,#3)[#4]{#5}}}
```

LWR@glrbox (*env.*)   {⟨*macroname*⟩}

```
1023 \def\LWR@glrbox#1{%
1024   \edef\reserved@a{%
1025     \endgroup
1026     \global\setbox#1\hbox{%
1027       \begingroup\aftergroup}%
1028       \def\noexpand\@currenvir{\@currenvir}%
1029       \def\noexpand\@currenvline{\on@line}}%
1030   \reserved@a
1031   \@endpefalse
1032   \color@setgroup
1033     \ignorespaces}
1034 \let\LWR@endglrbox\LWR@endlrbox
```

29.11   Converting a macro name to a cs name

\macrotocsname   {⟨*macro name with backslash*⟩}

Results in the macro name without the leading backslash.

Ref: https://tex.stackexchange.com/questions/42318/
removing-a-backslash-from-a-character-sequence

```
1035 \newcommand*{\macrotocsname}[1]{%
1036   \ifcat\relax\noexpand#1%
1037     \expandafter\expandafter\expandafter\@gobble\expandafter\string
1038   \fi
1039   #1%
1040 }
```

29.12   Title case

\LWRtexttitlecase

```
1041 \ExplSyntaxOn
1042 \newcommand*{\LWRtexttitlecase}[1]{%
1043     \text_titlecase:n{#1}%
1044 }
1045 \ExplSyntaxOff
```

29.13   LetLtxMacrocs

\LWR@LetLtxMacrocs {⟨*newcsname*⟩} {⟨*oldcsname*⟩}

\LetLtxMacro with cs names.

```
1046 \newcommand*{\LWR@LetLtxMacrocs}[2]{%
1047     \expandafter\LetLtxMacro\csname #1\expandafter\endcsname%
1048     \csname#2\endcsname%
1049 }
```

29.14   Absorbing a star

\LWR@absorbstar {⟨*csname*⟩}

Modifies a macro to aborb a star. Used for cleveref, since hyperref is emulated, so
the starred macros are not created by cleveref.

```
1050 \newcommand*{\LWR@absorbstar}[1]{%
1051     \LWR@LetLtxMacrocs{LWR@origns@#1}{#1}%
1052     \csdef{#1}{\@ifstar{\csuse{LWR@origns@#1}}{\csuse{LWR@origns@#1}}}
1053     \expandafter\robustify\csname #1\endcsname
1054 }
```

30   Operating-System portability

Unix (*Prog*)   lwarp tries to detect which operating system is being used. UNIX / MAC OS / LINUX
Mac OS (*Prog*)   is the default (collectively referred to as "UNIX" in the configuration files), and
Linux (*Prog*)   MS-WINDOWS is supported as well.

MS-Windows (*Prog*) If MS-WINDOWS is not correctly detected, use the lwarp option OSWindows.

Windows (*Prog*)   When detected or specified, the operating-system path separator used by lwarp
OSWindows (*Opt*)   is modified, and the boolean usingOSWindows is set true. This boolean may be
                     tested by the user for later use.

30.1 Literal characters

Literal characters to be used in `PrintLatexCmd` and `HTMLLatexCmd`. These are defined without @ to easily allow their inclusion in the user's document.

The literal % character:

```
1055 \let\LWRpercent\@percentchar
```

The literal $ character:

```
1056 \catcode`\$=12
1057 \def\LWRdollar{$}
1058 \def\LWRdollar{$}% syntax highlighting
1059 \catcode`\$=3
```

The literal & character:

```
1060 \catcode`\&=12
1061 \def\LWRamp{&}
1062 \catcode`\&=4
```

The literal \ character. The ampersand is temporarily set to the escape character during the definition of the backslash macro.

```
1063 \catcode`\&=0
1064 &catcode`&\=12
1065 &def&LWRbackslash{\}
1066 &catcode`&\=0
1067 \catcode`\&=4
```

The literal { character. The ampersand is temporarily set to the begin group character during the definition of the leftbrace macro.

```
1068 \catcode`\&=1
1069 \catcode`\{=12
1070 \def\LWRleftbrace&{}
1071 \catcode`\{=1
1072 \catcode`\&=4
```

The literal } character. The ampersand is temporarily set to the end group character during the definition of the leftbrace macro.

```
1073 \catcode`\&=2
1074 \catcode`\}=12
1075 \def\LWRrightbrace{}&
1076 \catcode`\}=2
1077 \catcode`\&=4
```

The literal # character:

```
1078 \catcode`\#=12
1079 \def\LWRhash{#}
1080 \catcode`\#=6
```

\LWRopquote   The operating system's quote mark, UNIX default. For WINDOWS, see \LWR@setOSWindows, below.

1081 \def\LWRopquote{'}

\LWRopseq The operating system's sequential execution command, Unix default. For Windows, see \LWR@setOSWindows, below.

1082 \def\LWRopseq{\space\LWRamp\LWRamp\space\space}

30.2 Common portability code

usingOSWindows (*bool*) Set if the OSWindows option is used, or if Windows is automatically detected.

1083 \newbool{usingOSWindows}
1084 \boolfalse{usingOSWindows}

30.3 Unix, Linux, and Mac OS

\OSPathSymbol Symbol used to separate directories in a path.

1085 \newcommand*{\OSPathSymbol}{/}

30.4 MS-Windows

For MS-Windows:

\LWR@setOSWindows Set defaults for the MS-Windows operating system. lwarp attempts to auto-detect the operatings system, and the OSWindows option may also be used to force MS-Windows compatibility.

1086 \newcommand*{\LWR@setOSWindows}
1087 {
1088 \booltrue{usingOSWindows}
1089 \renewcommand*{\OSPathSymbol}{\@backslashchar}
1090 \def\LWRopquote{"}
1091 \def\LWRopseq{\space\LWRamp\space\space}
1092 }

Test for windows during compile. The user may also specify OSWindows package option in case this test fails.

1093 \ifwindows
1094 \LWR@setOSWindows
1095 \fi

31 Package options

kvoptions (*Pkg*) Allows key/value package options.

1096 \RequirePackage{kvoptions}
1097 \SetupKeyvalOptions{family=LWR,prefix=LWR@}

\lwarpsetup A user interface to set the keys:

```
1098 \newcommand{\lwarpsetup}[1]{\setkeys{LWR}{#1}}
```

warpingprint (*bool*)

warpingHTML (*bool*)

mathjax (*bool*)   Set to true/false depending on the package option selections for print/HTML/EPUB
                     output and mathsvg/mathjax.

LWR@origmathjax (*bool*)

LWR@origmathjax remembers the original setting to be restored by \displaymathnormal.

```
1099 \newbool{warpingprint}
1100 \newbool{warpingHTML}
1101 \newbool{mathjax}
1102 \newbool{LWR@origmathjax}
```

defaults The default is print output, and SVG math if the user chose HTML output.

```
1103 \booltrue{warpingprint}%
1104 \boolfalse{warpingHTML}%
1105 \boolfalse{mathjax}%
```

warpdisable (*Opt*) If the warpdisable option is given, both boolean warpingprint and boolean
                     warpingHTML are false, and may be used for \ifbool tests. This option may be
                     used to disable almost all of lwarp, for testing purposes.

```
1106 \DeclareVoidOption{warpdisable}{%
1107     \PackageInfo{lwarp}{Using option 'warpdisable'}
1108     \boolfalse{warpingprint}%
1109     \boolfalse{warpingHTML}%
1110 }
```

warpprint (*Opt*) If the warpprint option is given, boolean warpingprint is true and boolean
                   warpingHTML is false, and may be used for \ifbool tests.

```
1111 \DeclareVoidOption{warpprint}{%
1112     \PackageInfo{lwarp}{Using option 'warpprint'}
1113     \booltrue{warpingprint}%
1114     \boolfalse{warpingHTML}%
1115 }
```

warpHTML (*Opt*) Anything in the warpHTML environment will be generated for HTML output only.

warpHTML (*Opt*) If the warpHTML option is given, boolean warpingHTML is true and boolean warpingprint
                  is false, and may be used for \ifbool tests.

```
1116 \DeclareVoidOption{warpHTML}{%
1117     \PackageInfo{lwarp}{Using option 'warpHTML'}%
1118     \booltrue{warpingHTML}%
1119     \boolfalse{warpingprint}%
1120 }
```

mathsvg (*Opt*) Option mathsvg selects SVG math display: If the mathsvg option is given, boolean
                 mathjax is false, and may be used for \ifbool tests.

```
1121 \DeclareVoidOption{mathsvg}{%
1122     \PackageInfo{lwarp}{Using option 'mathsvg'}
```

```
1123        \boolfalse{mathjax}%
1124        \boolfalse{LWR@origmathjax}%

1125 }
```

mathjax (*Opt*)   Option mathjax selects MATHJAX math display: If the mathjax option is given,
                   boolean mathjax is true, may be used for \ifbool tests.

```
1126 \DeclareVoidOption{mathjax}{%
1127        \PackageInfo{lwarp}{Using option 'mathjax'}
1128        \booltrue{mathjax}%
1129        \booltrue{LWR@origmathjax}%

1130 }
```

BaseJobname (*Opt*)   Option BaseJobname sets the \BaseJobname for this document.
Default: \jobname
                       This is the \jobname of the printed version, even if currently compiling the HTML
                       version. I.e. this is the \jobname without _html appended. This is used to set
                       \HomeHTMLFilename if the user did not provide one.

```
1131 \DeclareStringOption[\jobname]{BaseJobname}
```

ImagesDirectory (*Opt*)   Option ImagesDirectory sets the name of the directory to use for the lateximage
Default: \jobname-images   images.

```
1132 \DeclareStringOption[\BaseJobname-images]{ImagesDirectory}
```

ImagesName (*Opt*)   Option ImagesName sets the prefix to use for the lateximage images.
Default: image-
```
1133 \DeclareStringOption[image-]{ImagesName}
```

makeindexStyle (*Opt*)   Selects a custom .ist file. A customized file should be based on lwarp.ist. See
Default: lwarp.ist   section 8.6.21.

```
1134 \DeclareStringOption[lwarp.ist]{makeindexStyle}
```

xindyStyle (*Opt*)   Selects a custom .xdy file. A customized file should be based on lwarp.xdy. See
Default: lwarp.xdy   section 8.6.22.

```
1135 \DeclareStringOption[lwarp.xdy]{xindyStyle}
```

xindyLanguage (*Opt*)   Sets the *xindy* language to be assigned in *lwarpmk*'s configuration files. This is then
Default: english   used by *lwarpmk* while processing the index and glossary.

```
1136 \DeclareStringOption[english]{xindyLanguage}
```

xindyCodepage (*Opt*)   Sets the *xindy* codepage to be assigned in *lwarpmk*'s configuration files. This is
Default: utf8   then used by *lwarpmk* while processing the index.

```
1137 \DeclareStringOption[utf8]{xindyCodepage}
```

xindexConfig (*Opt*)   Selects a custom xindex-*.lua file. A customized file should be based on
Default: <empty>   xindex-cfg.lua. See section 8.6.23.

```
1138 \DeclareStringOption[]{xindexConfig}
```

pdftotextEnc (*Opt*)   The option pdftotextEnc sets the encoding used by *pdftotext*. This is passed to
Default: UTF-8   *pdftotext* using its -enc option, and is used when converting LATEX PDF output with
HTML tags into a plain-text file with HTML tags.

```
1139 \DeclareStringOption[UTF-8]{pdftotextEnc}
```

lwarpmk (*Opt*)   Tells lwarp to generate a local copy of *lwarpmk* called lwarpmk.lua. Useful for
archiving for future use. This file may be made executable and acts just like
lwarpmk.

If lwarpmk option, creates a local copy of lwarpmk.lua:

```
1140 \newbool{LWR@creatinglwarpmk}
1141 \boolfalse{LWR@creatinglwarpmk}
1142
1143 \DeclareVoidOption{lwarpmk}{
1144     \PackageInfo{lwarp}{Using option 'lwarpmk'}
1145     \booltrue{LWR@creatinglwarpmk}
1146 }
```

OSWindows (*Opt*)   Tells lwarp to use MS-WINDOWS compatibility. Auto-detection of the operating sys-
tem is attempted, and this option is only necessary if the auto-detection fails. See
the automatically-generated lwarpmk.conf file to find out whether the operating
system was detected correctly.

```
1147 \DeclareVoidOption{OSWindows}{
1148     \PackageInfo{lwarp}{Using option 'OSWindows'}
1149     \LWR@setOSWindows
1150 }
```

HomeHTMLFilename (*Opt*)   The filename of the homepage. The default is the jobname. This option is stored
Default: \BaseJobname   into \LWR@HomeHTMLFilename, and later transferred into \HomeHTMLFilename for
internal use.

```
1151 \DeclareStringOption[]{HomeHTMLFilename}
```

HTMLFilename (*Opt*)   The filename prefix of web pages after the homepage. The default is empty, no
Default: <empty>   prefix. This option is stored into \LWR@HTMLFilename, and later transferred into
\HTMLFilename for internal use.

```
1152 \DeclareStringOption[]{HTMLFilename}
```

PrintLatexCmd (*Opt*)   The shell commands to use to compile the print document.
Default: <automatic>
```
1153 \DeclareStringOption[]{PrintLatexCmd}
```

HTMLLatexCmd (*Opt*)   The shell commands to use to compile the HTML document.
Default: <automatic>
```
1154 \DeclareStringOption[]{HTMLLatexCmd}
```

PrintIndexCmd (*Opt*)   The shell commands to use to compile the print indexes.
Default: <empty>
```
1155 \DeclareStringOption[]{PrintIndexCmd}
```

HTMLIndexCmd (*Opt*) The shell commands to use to compile the HTML indexes.
Default: <empty>

```
1156 \DeclareStringOption[]{HTMLIndexCmd}
```

LatexmkIndexCmd (*Opt*) The shell commands to by used by *latexmk* to compile the print indexes. Un-
Default: <empty> like PrintIndexCmd and HTMLIndexCmd, LatexmkIndexCmd does not include the
filename, which will be provided by *latexmk*.

```
1157 \DeclareStringOption[]{LatexmkIndexCmd}
```

makeindex (*Opt*) Tells lwarp to use *makeindex* for index generation. When lwarpmk.conf and
*.lwarpmkconf are generated, PrintIndexCmd and HTMLIndexCmd will be set for
makeindex with a single index file.

```
1158 \DeclareBoolOption[false]{makeindex}
```

xindy (*Opt*) Tells lwarp to use *xindy* for index generation. When lwarpmk.conf and *.lwarpmkconf
are generated, PrintIndexCmd and HTMLIndexCmd will be set for *xindy* with a single
index file.

```
1159 \DeclareBoolOption[false]{xindy}
```

xindex (*Opt*) Tells lwarp to use *xindex* for index generation. When lwarpmk.conf and *.lwarpmkconf
are generated, PrintIndexCmd and HTMLIndexCmd will be set for *xindex* with a single
index file.

```
1160 \DeclareBoolOption[false]{xindex}
```

IndexRef (*Opt*) Tells lwarp how to display the index entries in HTMLoutput. See section 7.5.
Default: cref
```
1161 \DeclareStringOption[cref]{IndexRef}
```

GlossaryCmd (*Opt*) The shell command to use to compile the glossary. The print or HTML version of
Default: makeglossaries the glossary filename will be appended to this command.

```
1162 \DeclareStringOption[makeglossaries]{GlossaryCmd}
```

latexmk (*Opt*) Option latexmk tells *lwarpmk* to use *latexmk* when compiling documents.

```
1163 \DeclareBoolOption[false]{latexmk}
```

dvips (*Opt*) Option dvips tells *lwarpmk* to use *dvips* when compiling DVI *latex* documents.

```
1164 \DeclareBoolOption[false]{dvips}
```

dvipdfm (*Opt*) Option dvipdfm tells *lwarpmk* to use *dvipdfm* when compiling DVI *latex* documents.

```
1165 \DeclareBoolOption[false]{dvipdfm}
```

dvipdfmx (*Opt*) Option dvipdfmx tells *lwarpmk* to use *dvipdfmx* when compiling DVI *latex* docu-
ments.

```
1166 \DeclareBoolOption[false]{dvipdfmx}
```

Execute options   Execute the package options, with the defaults which have been set just above:

```
1167 \ProcessKeyvalOptions*\relax
```

31.1   Additional options support

Assign the \BaseJobname if the user hasn't provided one:

```
1168 \providecommand*{\BaseJobname}{\LWR@BaseJobname}
```

Defaults unless already over-ridden by the user:

```
1169 \ifcsempty{LWR@HomeHTMLFilename}{
1170     \newcommand*{\HomeHTMLFilename}{\BaseJobname}
1171 }{
1172     \csedef{HomeHTMLFilename}{\LWR@HomeHTMLFilename}
1173 }
1174
1175 \csedef{HTMLFilename}{\LWR@HTMLFilename}
```

Special handling for underscores in labels and filenames.

\LWR@sanitized   The sanitized version of what was given to \LWR@sanitize. Characters are set to their detokenized versions. Required for underscores in labels and filenames.

```
1176 \newcommand*{\LWR@sanitized}{}
```

\LWR@sanitize {⟨*text*⟩}

Sanitizes the text and returns the result in \LWR@sanitized.

```
1177 \newcommand*{\LWR@sanitize}[1]{%
1178 \edef\LWR@sanitized{#1}%
1179 \edef\LWR@sanitized{\detokenize\expandafter{\LWR@sanitized}}%
1180 }
```

Sanitize some string options to neutralize underscores.

```
1181 \LWR@sanitize{\LWR@BaseJobname}
1182 \edef\LWR@BaseJobname{\LWR@sanitized}
1183
1184 \LWR@sanitize{\LWR@ImagesDirectory}
1185 \edef\LWR@ImagesDirectory{\LWR@sanitized}
1186
1187 \LWR@sanitize{\LWR@ImagesName}
1188 \edef\LWR@ImagesName{\LWR@sanitized}
```

\LWR@PrintIndexCmd and \LWR@HTMLIndexCmd are tested to see if they are empty. If so, they are set to a reasonable defaults for a single index using *makeindex*, then possibly set to defaults for *xindy* if the lwarp xindy option was selected, then likewise for *xindex* if the xindex option was selected.

```
1189 \ifdefempty{\LWR@PrintIndexCmd}{
1190     \renewcommand{\LWR@PrintIndexCmd}{%
```

```
1191        makeindex  -s \LWR@makeindexStyle \space  \jobname.idx%
1192    }
1193    \ifbool{LWR@xindy}{
1194        \renewcommand{\LWR@PrintIndexCmd}{%
1195            xindy
1196            -M \LWR@xindyStyle \space
1197            -L \LWR@xindyLanguage \space
1198            -C \LWR@xindyCodepage \space
1199            \jobname.idx%
1200        }
1201    }{}
1202    \ifbool{LWR@xindex}{
1203        \ifdefvoid{\LWR@xindexConfig}{
1204            \renewcommand{\LWR@PrintIndexCmd}{%
1205                xindex
1206                \jobname.idx%
1207            }
1208        }{
1209            \renewcommand{\LWR@PrintIndexCmd}{%
1210                xindex
1211                -c \LWR@xindexConfig \space
1212                \jobname.idx%
1213            }
1214        }
1215    }{}
1216 }{}
1217
1218 \ifdefempty{\LWR@HTMLIndexCmd}{
1219    \renewcommand{\LWR@HTMLIndexCmd}{%
1220        makeindex  -s \LWR@makeindexStyle \space  \jobname_html.idx%
1221    }
1222    \ifbool{LWR@xindy}{
1223        \renewcommand{\LWR@HTMLIndexCmd}{%
1224            xindy
1225            -M \LWR@xindyStyle \space
1226            -L \LWR@xindyLanguage \space
1227            -C \LWR@xindyCodepage \space
1228            \jobname_html.idx%
1229        }
1230    }{}
1231    \ifbool{LWR@xindex}{
1232        \ifdefvoid{\LWR@xindexConfig}{
1233            \renewcommand{\LWR@HTMLIndexCmd}{%
1234                xindex
1235                \jobname_html.idx%
1236            }
1237        }{
1238            \renewcommand{\LWR@HTMLIndexCmd}{%
1239                xindex
1240                -c \LWR@xindexConfig \space
1241                \jobname_html.idx%
1242            }
1243        }
1244    }{}
1245 }{}
1246
1247 \ifdefempty{\LWR@LatexmkIndexCmd}{
1248    \renewcommand{\LWR@LatexmkIndexCmd}{%
1249        makeindex  -s \LWR@makeindexStyle%
1250    }
```

```
1251  \ifbool{LWR@xindy}{
1252      \renewcommand{\LWR@LatexmkIndexCmd}{%
1253          xindy
1254          -M \LWR@xindyStyle \space
1255          -L \LWR@xindyLanguage \space
1256          -C \LWR@xindyCodepage%
1257      }
1258  }{}
1259  \ifbool{LWR@xindex}{
1260      \ifdefvoid{\LWR@xindexConfig}{
1261          \renewcommand{\LWR@LatexmkIndexCmd}{%
1262              xindex
1263          }
1264      }{
1265          \renewcommand{\LWR@LatexmkIndexCmd}{%
1266              xindex
1267              -c \LWR@xindexConfig
1268          }
1269      }
1270  }{}
1271  }{}
```

31.2   Conditional compilation

\warpprintonly {⟨*contents*⟩}

>   Only process the contents if producing printed output.

```
1272 \newcommand{\warpprintonly}[1]{\ifbool{warpingprint}{#1}{}}
```

\warpHTMLonly {⟨*contents*⟩}

>   Only process the contents if producing HTML output.

```
1273 \newcommand{\warpHTMLonly}[1]{\ifbool{warpingHTML}{#1}{}}
```

comment (*Pkg*)   Provides conditional code blocks.

>   Attempts to use versions or verbatim fail in some cases, and do not provide much
>   of a speed benefit even when they do work.

```
1274 \RequirePackage{comment}
```

\LWR@includecomment {⟨*env name*⟩} {⟨*partial filename*⟩}

\LWR@excludecomment {⟨*env name*⟩} {⟨*partial filename*⟩}

>   Use many comment cut files to avoid collision in case the user uses the comment
>   package. Each filename is "comment_#2.cut". Based on the comment package.

```
1275 \def\LWR@includecomment
1276 #1#2{\message{Lwarp: Including comment '#1'}%
1277     \csarg\def{After#1Comment}{%
1278         \CloseAndInputCutFile%
1279         \csundef{LWR@#1commentused}%
```

```
1280        }
1281    \csarg\def{#1}{%
1282        \endgroup
1283        \ifcsdef{LWR@#1commentused}{
1284            \PackageError{lwarp}%
1285                {Nested #1 environment}%
1286                {%
1287                    Environment #1 cannot be nested.\MessageBreak
1288                    This can happen when a package is loaded
1289                    from inside a\MessageBreak
1290                    #1 environment.%
1291                }%
1292        }{\relax}
1293        \csdef{LWR@#1commentused}{}
1294        \message{Including '#1' comment.}%
1295        \def\CommentCutFile{comment_#2.cut}
1296        \SetUpCutFile
1297        \ProcessComment{#1}
1298    }%
1299    \CommentEndDef{#1}
1300 }
1301
1302 \def\LWR@excludecomment
1303 #1#2{\message{Lwarp: Excluding comment '#1'}%
1304     \csarg\def{#1}{
1305        \endgroup
1306        \message{Excluding '#1' comment.}%
1307        \begingroup
1308          \def\CommentCutFile{comment_#2.cut}
1309           \def\ProcessCutFile{}%
1310          \def\ThisComment####1{}%
1311           \ProcessComment{#1}
1312     }%
1313     \csarg\def{After#1Comment}{\CloseAndInputCutFile \endgroup}
1314     \CommentEndDef{#1}}
```

warpall (*env.*) Anything in the warpall environment will be generated for print or HTML outputs.

```
1315 \LWR@includecomment{warpall}{all}
```

warpHTML (*env.*) For HTML output:

```
1316 \ifbool{warpingHTML}
1317     {\LWR@includecomment{warpHTML}{html}}
1318     {\LWR@excludecomment{warpHTML}{html}}
```

warpprint (*env.*) Anything in the warpprint environment will be generated for print output only.

```
1319 \ifbool{warpingprint}
1320     {\LWR@includecomment{warpprint}{print}}
1321     {\LWR@excludecomment{warpprint}{print}}
```

If warpdisable, turn off both print and HTML output:

```
1322 \ifboolexpr{bool {warpingprint} or bool {warpingHTML}}
1323     {}
1324     {
```

```
1325        \LWR@excludecomment{warpHTML}{html}
1326        \LWR@excludecomment{warpprint}{print}
1327        \LWR@excludecomment{warpMathJax}{mathjax}
1328    }
```

warpMathJax (*env.*) Only if MathJax is being used along with HTML.

```
1329 \begin{warpprint}
1330 \LWR@excludecomment{warpMathJax}{mathjax}
1331 \end{warpprint}
1332
1333 \begin{warpHTML}
1334 \ifbool{mathjax}
1335    {\LWR@includecomment{warpMathJax}{mathjax}}
1336    {\LWR@excludecomment{warpMathJax}{mathjax}}
1337 \end{warpHTML}
```

warpsvg (*env.*) Only if SVG math is being used along with HTML, or in print mode.

```
1338 \begin{warpprint}
1339 \LWR@includecomment{warpsvg}{mathsvg}
1340 \end{warpprint}
1341
1342 \begin{warpHTML}
1343 \ifbool{mathjax}
1344    {\LWR@excludecomment{warpsvg}{mathsvg}}
1345    {\LWR@includecomment{warpsvg}{mathsvg}}
1346 \end{warpHTML}
```

LWRcreatelwarpmk (*env.*) Optionally generate a local copy of *lwarpmk*. Default to no.

```
1347 \ifbool{LWR@creatinglwarpmk}
1348    {\LWR@includecomment{LWRcreatelwarpmk}{lwarpmk}}
1349    {\LWR@excludecomment{LWRcreatelwarpmk}{lwarpmk}}
```

32   Required packages

These packages are automatically loaded by lwarp when generating HTML output. Some of them are also automatically loaded when generating print output, but some are not.

for HTML output: `1350 \begin{warpHTML}`

fontspec (*Pkg*) Load fontspec if necessary:

```
1351 \ifxetexorluatex
1352 \IfPackageLoadedTF{fontspec}{}{
1353    \usepackage[no-math]{fontspec}
1354 }
```

The monospaced font is used for HTML tags, so turn off its TeX ligatures and common ligatures:

```
1355 \defaultfontfeatures[\rmfamily]{Ligatures={NoCommon,TeX}}
```

```
1356 \defaultfontfeatures[\sffamily]{Ligatures={NoCommon,TeX}}
1357 \defaultfontfeatures[\ttfamily]{Ligatures=NoCommon}
1358 \else
```

pdflatex only: Only pre-loaded if *pdflatex* is being used.

microtype (*Pkg*)

ligatures   Older browsers don't display ligatures. Turn off letter ligatures, keeping LaTeX dash
and quote ligatures, which may fail on older browers but at least won't corrupt
written words.

```
1359 \RequirePackage{microtype}
1360
1361 \microtypesetup{
1362     protrusion=false,
1363     expansion=false,
1364     tracking=false,
1365     kerning=false,
1366     spacing=false}
1367 %     \begin{macrocode}
1368 %
1369 % Disable ligatures for typewriter fonts.
1370 % The comma was causing issues with \MathJax\ and \cs{,} followed by a comma.
1371 % Ligatures for f, q, t, etc used to be disabled for non-typewriter fonts, but
1372 % are now allowed.
1373 % \changes{v0.89}{2020/08/01}{Disable typewriter ligatures.}
1374 % ^^A \DisableLigatures[{,},f,q,t,T,Q]{encoding = *,family = *}% previous
1375 %     \begin{macrocode}
1376 \DisableLigatures{encoding = *,family = tt*}

1377 \fi

1378 \end{warpHTML}
```

geometry (*Pkg*)   Tactics to avoid unwanted page breaks and margin overflow:

- Uses a very long and wide page to minimize page breaks and margin over-
  flow.

- Uses a scriptsize font.

- Uses extra space at the margin to avoid HTML tag overflow off the page.

- Forces a new PDF page before some environments.

- Forces line break between major pieces of long tags.

for HTML output: `1379 \begin{warpHTML}`

If **geometry** has not yet been loaded, use the preexising page and text sizes to be
preserved for later reuse. These will be replaced by **lwarp** \AtBeginDocument with
a very large page size to reduce HTML tag overflow off the page.

```
1380 \IfPackageLoadedTF{geometry}
1381 {}{
1382     \RequirePackage[
1383         reset,
```

```
1384          paperwidth=\paperwidth,
1385          paperheight=\paperheight,
1386          textwidth=\textwidth,
1387          textheight=\textheight,
1388          left=\oddsidemargin,
1389          top=\topmargin,
1390          marginparsep=\marginparsep,
1391          marginparwidth=\marginparwidth,
1392     ]{geometry}
1393 }
```

Remember the original definitions for later reuse. If the geometry package is loaded by the user, lwarp-geometry will nullify the user-level originals.

```
1394 \LetLtxMacro\LWR@origgeometry\geometry
1395 \LetLtxMacro\LWR@orignewgeometry\newgeometry
1396 \LetLtxMacro\LWR@origrestoregeometry\restoregeometry
1397 \LetLtxMacro\LWR@origsavegeometry\savegeometry
1398 \LetLtxMacro\LWR@origloadgeometry\loadgeometry
```

LWR@allowanothergeometry
(*bool*)

geometry may be loaded by the user before lwarp, after lwarp, or not at all. If before lwarp, it will have already been loaded by now and its page layout has already been saved. If geometry is loaded after lwarp, its layout will be set at that time and the user macros nullified. \AtEndPreamble this layout will be saved. If the user never loads geometry, lwarp-geometry will be loaded \AtBeginDocument, but it should not change the page layout set here. This is controlled by the boolean LWR@allowanothergeometry. Geometry may be adjusted throughout the preamble until \AtEndPreamble, when this boolean is set false.

```
1399 \newbool{LWR@allowanothergeometry}
1400 \booltrue{LWR@allowanothergeometry}
```

Use \AtEndPreamble to avoid class and option conflict by changing settings after other packages load, instead of using geometry package options:

```
1401 \AtEndPreamble{
```

Whatever geometry choices the user has made in the preamble, either before or after lwarp was loaded, are now saved for possible temporary reuse, such as by lyluatex.

See the lwarp-geometry section for what happens if geometry is loaded after lwarp.

```
1402 \LWR@origsavegeometry{LWR@usergeometry}
```

The user's paper size is saved for later reuse, such as by the pdfpages or parallel packages.

```
1403 \newlength{\LWR@userspaperwidth}
1404 \setlength{\LWR@userspaperwidth}{\paperwidth}
1405
1406 \newlength{\LWR@userspaperheight}
1407 \setlength{\LWR@userspaperheight}{\paperheight}
1408
1409 \newlength{\LWR@usersmarginparwidth}
1410 \setlength{\LWR@usersmarginparwidth}{\marginparwidth}
1411
1412 \newlength{\LWR@userstextwidth}
```

```
1413 \setlength{\LWR@userstextwidth}{\textwidth}
1414
1415 \newlength{\LWR@userstextheight}
1416 \setlength{\LWR@userstextwidth}{\textheight}
```

For lwarp, use a very large page and margins to help avoid letting HTML tags run off the edge:

```
1417 \LWR@origgeometry{
1418     reset,
1419     paperheight=190in,
1420     paperwidth=20in,
1421     left=2in,
1422     right=6in,
1423     top=1in,
1424     bottom=1in,
1425     heightrounded,%
1426 }
```

The lwarp page geometry is saved for future restore:

```
1427 \LWR@origsavegeometry{LWR@lwarpgeometry}
```

No longer adjust the page layout when lwarp-geometry is loaded \AtBeginDocument:

```
1428 \boolfalse{LWR@allowanothergeometry}%
```

ltjsbook and other classes can print vertically, and require these to be reset by lwarp:

```
1429 \setlength{\textheight}{0.8\paperheight}
1430 \setlength{\textwidth}{0.7\paperwidth}
1431
1432 \@twosidefalse
1433 \@mparswitchfalse
1434 }% \AtEndPreamble
1435
1436 \end{warpHTML}
```

for HTML & PRINT: `1437 \begin{warpall}`

xparse (*Pkg*)

LaTeX3 command argument parsing

```
1438 \IfFormatAtLeastTF{2020-10-01}{}{\RequirePackage{xparse}}
```

calc (*Pkg*)

```
1439 \RequirePackage{calc}
```

```
1440 \end{warpall}
```

for HTML output: `1441 \begin{warpHTML}`

expl3 (*Pkg*)

LaTeX3 programming

1442 `\RequirePackage{expl3}`

gettitlestring (*Pkg*)

Used to emulate `\nameref`.

1443 `\RequirePackage{gettitlestring}`
1444
1445
1446 `\end{warpHTML}`

for HTML & PRINT: 1447 `\begin{warpall}`

filecontents (*Pkg*)

Used to write helper files while creating the print version.

Recent versions of LaTeX (as of Fall 2019) now include the functionality of the filecontents package, but with a new optional argument used to specify whether to force the overwriting of an existing file. If an older LaTeX kernel is used, the original filecontents package is used, but it is patched to throw away the new optional argument.

```
1448 \@ifundefined{filec@ntents@opt}{% older kernel, discard optional args
1449
1450     \RequirePackage{filecontents}
1451
1452     \LetLtxMacro\LWR@orig@filec@ntents\filec@ntents
1453
1454     \IfPackageAtLeastTF{filecontents}{2011/10/08}
1455     {
```

For a newer version of the filecontents package, simply discard the optional argument.

```
1456         \renewcommand*{\filec@ntents}[1][]{\LWR@orig@filec@ntents}
1457     }
1458     {% patch older package for morewrites
```

For an older version of filecontents, discard the optional argument, and also patch to work with morewrites, per https://tex.stackexchange.com/questions/312830/does-morewrites-not-support-filecontents-and-can-i-write-body-of-environment-us/312910

```
1459         \newwrite\fcwrite
1460         \renewcommand*{\filec@ntents}[1][]{%
1461             \def\chardef##1\write{\let\reserved@c\fcwrite}%
1462             \LWR@orig@filec@ntents%
1463         }
1464     }
1465
1466 }% older kernel
1467 {% newer kernel
```

For a newer kernel with a `filecontents` environment which accepts the optional `overwrite` argument, use the environment as-is.

```
1468 }% newer kernel, filecontents env accepts optional args, do not load package
```

```
1469 \end{warpall}
```

for HTML output: `1470 \begin{warpHTML}`

xifthen (*Pkg*)

```
1471 \RequirePackage{xifthen}
```

verbatim (*Pkg*)

```
1472 \RequirePackage{verbatim}
```

refcount (*Pkg*)

Provides \setcounterref, \setcounterpageref, etc.

```
1473 \RequirePackage{refcount}
```

newfloat (*Pkg*)

```
1474 \RequirePackage{newfloat}
```

```
1475 \end{warpHTML}
```

for HTML & PRINT: `1476 \begin{warpall}`

xstring (*Pkg*)   There was a short-term bug in xstring regarding \IfInteger which affected lwarp's
⚠ index   index generation. The updated version is requested here.

```
1477 \RequirePackage{xstring}[2019/02/01]
```

environ (*Pkg*)   Used to encapsulate math environments for re-use in HTML <alt> text.

```
1478 \RequirePackage{environ}
```

```
1479 \end{warpall}
```

for HTML output: `1480 \begin{warpHTML}`

printlen (*Pkg*)   Used to convert lengths for image width/height options.

```
1481 \RequirePackage{printlen}
```

\LWR@printlength {⟨*length*⟩}

Prints a length using a locally-controlled unit and space. Rounding is used unless the length is small.

```
1482 \newrobustcmd*{\LWR@printlength}[1]{%
1483     \begingroup%
```

```
1484        \uselengthunit{PT}%
1485        \renewcommand*{\unitspace}{}%
1486        \ifdimless{#1}{10pt}{%
1487            \printlength{#1}%
1488        }{%
1489            \rndprintlength{#1}%
1490        }%
1491        \endgroup%
1492 }
```

```
1493 \end{warpHTML}
```

33   Loading packages

\RequirePackage and \usepackage are modified to error-check for certain packages, and for HTML they load the lwarp- version if it exists.

for HTML & PRINT: 1494 \begin{warpall}

Remember the original \RequirePackage:

```
1495 \LetLtxMacro\LWR@origRequirePackage\RequirePackage
1496 \LetLtxMacro\LWR@origRequirePackageWithOptions\RequirePackageWithOptions
```

\LWR@requirepackagenames   Stores the list of required package names.

```
1497 \newcommand*{\LWR@requirepackagenames}{}
```

\LWR@parsedrequirepackagenames   Stores the parsed list of required package names after spaces are removed and lwarp- is prepended.

```
1498 \newcommand*{\LWR@parsedrequirepackagenames}{}
```

\LWR@nullifycomment   Remove the preexisting comment environment. Certain packages define it for their own use.

```
1499 \newcommand*{\LWR@nullifycomment}{%
1500     \PackageInfo{lwarp}%
1501       {Nullifying the comment environment before loading \LWR@strresulttwo,}%
1502     \let\comment\relax%
1503     \let\endcomment\relax%
1504 }
```

\LWR@findword [⟨*1: separator*⟩] {⟨*2: list*⟩} {⟨*3: index*⟩} [⟨*4: destination*⟩]

Note that argument 4 is passed directly to \StrBetween.

```
1505 \newcommand*\LWR@findword[3][,]{%
1506     \StrBetween[#3,\numexpr#3+1]{#1#2#1}{#1}{#1}%
1507 }
```

\LWR@checkloadnever {⟨*bad package name*⟩} {⟨*replacement package names*⟩}

From now on, check for incompatible packages loaded via \usepackage, instead of packages loaded before lwarp:

```
1508 \LetLtxMacro\LWR@checkloadnever\LWR@afterloadnever
```

\LWR@checkloadfilename {⟨*filename*⟩} Checks if this filename should be loaded after lwarp, or never at all.

```
1509 \newcommand*{\LWR@checkloadfilename}[1]{%
```

Remember the package name to compare with, to be used by \LWR@checkloadnever and \LWR@checkloadbefore.

```
1510    \edef\LWR@tempone{#1}%
```

Check against the list of packages which should never be loaded:

```
1511    \LWR@checkloadnevers
```

The following should only be loaded before lwarp:

```
1512    \LWR@checkloadbefore{ctex}
1513    \LWR@checkloadbefore{fontspec}
1514    \LWR@checkloadbefore{inputenc}
1515    \LWR@checkloadbefore{inputenx}
1516    \LWR@checkloadbefore{nfssext-cfr}
1517    \LWR@checkloadbefore{fontaxes}
1518    \LWR@checkloadbefore{kotex}
1519    \LWR@checkloadbefore{kpfonts}% textcomp option clash
1520    \LWR@checkloadbefore{luatexja}
1521    \LWR@checkloadbefore{luatexja-fontspec}
1522    \LWR@checkloadbefore{luatexko}
1523    \LWR@checkloadbefore{morewrites}
1524    \LWR@checkloadbefore{newclude}
1525    \LWR@checkloadbefore{newunicodechar}
1526    \LWR@checkloadbefore{plext}
1527    \LWR@checkloadbefore{xeCJK}
1528    \LWR@checkloadbefore{xetexko}
1529    \LWR@checkloadbefore{zxjatype}
1530 }
```

\LWR@lookforpackagename {⟨*index*⟩}

If HTML, and if this is an lwarp-supported package name, re-direct it to the lwarp version by renaming it lwarp- followed by the original name.

Looks index deep into the list of package names, \LWR@requirepackagenames, and builds \LWR@parsedrequirepackagenames which is the modified list of names.

```
1531 \newcommand*{\LWR@lookforpackagename}[1]{%
```

Find the index'th package name from the list:

```
1532 \LWR@findword{\LWR@requirepackagenames}{#1}[\LWR@strresult]%
```

Remove blanks. The original name with blanks is in LWR@strresult and the final name with no blanks goes into LWR@strresulttwo.

```
1533 \StrSubstitute[100]{\LWR@strresult}{ }{}[\LWR@strresulttwo]%
```

See if the package name was found:

```
1534 \IfStrEq{\LWR@strresulttwo}{}%
1535 {}% no filename
1536 {% yes filename was found
```

Possible adjustments before loading the package. Maybe nullify the comment environment if the new package will be redefining it for a new purpose.

```
1537     \ifdefstring{\LWR@strresulttwo}{easyReview}{\LWR@nullifycomment}{}%
1538     \ifdefstring{\LWR@strresulttwo}{changes}{\LWR@nullifycomment}{}%
```

If HTML, check if the package should be loaded before lwarp, or never at all:

```
1539     \ifbool{warpingHTML}{\LWR@checkloadfilename{\LWR@strresulttwo}}{}%
```

If HTML, and if found, and if an lwarp-equivalent name exists, use lwarp-* instead.

```
1540     \ifboolexpr{
1541         bool{warpingHTML} and
1542         test{\IfFileExists{lwarp-\LWR@strresulttwo.sty}}
1543     }%
1544     {% lwarp-* file found
1545         \ifdefvoid{\LWR@parsedrequirepackagenames}{%
1546             \edef\LWR@parsedrequirepackagenames{lwarp-\LWR@strresulttwo}%
1547         }{%
1548             \edef\LWR@parsedrequirepackagenames{%
1549                 \LWR@parsedrequirepackagenames,lwarp-\LWR@strresulttwo%
1550             }%
1551         }%
1552     }%
1553     {% lwarp-* file not found
```

Otherwise, use the current package name.

```
1554         \ifdefvoid{\LWR@parsedrequirepackagenames}{%
1555             \edef\LWR@parsedrequirepackagenames{\LWR@strresulttwo}%
1556         }{%
1557             \edef\LWR@parsedrequirepackagenames{%
1558                 \LWR@parsedrequirepackagenames,\LWR@strresulttwo%
1559             }%
1560         }%
1561     }% no lwarp-* file
1562 }% yes filename
1563 }
```

\RequirePackage [⟨*1: options*⟩] {⟨*2: package names*⟩} [⟨*3: version*⟩]

For each of many package names in a comma-separated list, if an lwarp version of a package exists, select it instead of the LaTeX version.

```
1564 \RenewDocumentCommand{\RequirePackage}{o m o}{%
```

Redirect up to twenty names:[17]

---

[17]This was originally nine names, but then I came across a package which used twelve…

```
1565 \renewcommand*{\LWR@requirepackagenames}{#2}%
1566 \renewcommand*{\LWR@parsedrequirepackagenames}{}%
1567 \LWR@lookforpackagename{1}%
1568 \LWR@lookforpackagename{2}%
1569 \LWR@lookforpackagename{3}%
1570 \LWR@lookforpackagename{4}%
1571 \LWR@lookforpackagename{5}%
1572 \LWR@lookforpackagename{6}%
1573 \LWR@lookforpackagename{7}%
1574 \LWR@lookforpackagename{8}%
1575 \LWR@lookforpackagename{9}%
1576 \LWR@lookforpackagename{10}%
1577 \LWR@lookforpackagename{11}%
1578 \LWR@lookforpackagename{12}%
1579 \LWR@lookforpackagename{13}%
1580 \LWR@lookforpackagename{14}%
1581 \LWR@lookforpackagename{15}%
1582 \LWR@lookforpackagename{16}%
1583 \LWR@lookforpackagename{17}%
1584 \LWR@lookforpackagename{18}%
1585 \LWR@lookforpackagename{19}%
1586 \LWR@lookforpackagename{20}%
```

Error if braces are used in optional argument. This can cause an error, so tell how to avoid.

```
1587 \IfSubStr{\detokenize\expandafter{#1}}{\LWRleftbrace}%
1588    {%
1589        \PackageError{lwarp}{%
1590            You used:\MessageBreak
1591            \protect\usepackage[#1]{#2}\MessageBreak
1592            Braces in the package options will fail with Lwarp.\MessageBreak
1593            Instead, use:\MessageBreak
1594            \protect\PassOptionsToPackage{#1}{#2}\MessageBreak
1595            \protect\usepackage{#2}\MessageBreak
1596            near the line number given below.\MessageBreak
1597            Enter 'h' for more info%
1598        }%
1599        {%
1600          See the Lwarp manual troubleshooting index entry for\MessageBreak
1601            ``package, options with braces''%
1602        }%
1603    }%
1604    {}% no brace
```

\RequirePackage depending on the options and version:

```
1605 \IfValueTF{#1}%
1606 {% options given
```

The LaTeX3 key/value handler does not appear to expand the option argument, so it is pre-expanded here. This was a problem for mathspec, which passed options to fontspec.

```
1607    \edef\LWR@packageoptions{#1}%
1608    \IfValueTF{#3}% version given?
1609        {%
1610            \expandafter\LWR@origRequirePackage%
1611                \expandafter[\LWR@packageoptions]%
```

```
1612                    {\LWR@parsedrequirepackagenames}[#3]%
1613            }%
1614            {%
1615                \expandafter\LWR@origRequirePackage%
1616                    \expandafter[\LWR@packageoptions]%
1617                    {\LWR@parsedrequirepackagenames}%
1618            }%
1619 }%
1620 {% no options given
1621     \IfValueTF{#3}% version given?
1622        {\LWR@origRequirePackage{\LWR@parsedrequirepackagenames}[#3]}%
1623        {\LWR@origRequirePackage{\LWR@parsedrequirepackagenames}}%
1624 }%
1625 }
1626 \LetLtxMacro\usepackage\RequirePackage
1627 \@onlypreamble\RequirePackage
1628 \@onlypreamble\usepackage


1629 \end{warpall}
```

`1630 \begin{warpHTML}`

\LWR@ProvidesPackagePass {⟨*pkgname*⟩} [⟨*version*⟩]

Uses the original package, including options.

```
1631 \NewDocumentCommand{\LWR@ProvidesPackagePass}{m o}{
1632     \PackageInfo{lwarp}{%
1633        Using package `#1',\MessageBreak
1634        and adding lwarp modifications, including options,\MessageBreak%
1635     }%
1636     \IfValueTF{#2}%
1637        {\ProvidesPackage{lwarp-#1}[#2]}%
1638        {\ProvidesPackage{lwarp-#1}}%
1639     \DeclareOption*{%
1640        \PassOptionsToPackage{\CurrentOption}{#1}%
1641     }%
1642     \ProcessOptions\relax%
```

If using catoptions, an error occurs if a package is loaded with an option then loaded again with no options. lwarp does this if a package is preloaded then later patched. To avoid an error while using catoptions, if a package has already been loaded, it is loaded again with its original options.

```
1643     \IfPackageLoadedTF{#1}{%
1644        \edef\LWR@tempone{\csuse{opt@#1.sty}}%
1645        \IfValueTF{#2}%
1646            {%
1647                \expandafter\LWR@origRequirePackage%
1648                    \expandafter[\LWR@tempone]{#1}[#2]%
1649            }%
1650            {%
1651                \expandafter\LWR@origRequirePackage%
1652                    \expandafter[\LWR@tempone]{#1}%
1653            }%
1654     }{%
1655        \IfValueTF{#2}%
1656            {\LWR@origRequirePackage{#1}[#2]}%
1657            {\LWR@origRequirePackage{#1}}%
```

```
1658    }%
```

In some cases, the following seems to be required to avoid an "unknown option" error, such as when loading xcolor with options.

```
1659    \DeclareOption*{}%
1660    \ProcessOptions\relax%
1661 }
```

\LWR@ProvidesPackageDropA {⟨*name*⟩} {⟨*date or -NoValue-*⟩}

Declares the package. Factored for reuse.

```
1662 \newcommand*{\LWR@ProvidesPackageDropA}[2]{%
1663    \PackageInfo{lwarp}{%
1664        Replacing package `#1' with the lwarp version,\MessageBreak
1665        and discarding options,%
1666    }%
1667    \IfValueTF{#2}
1668    {\ProvidesPackage{lwarp-#1}[#2]}
1669    {\ProvidesPackage{lwarp-#1}}
1670 }
```

\LWR@ProvidesPackageDropB Nullifies then processes the options.

Seems to be required when options contain curly braces, which were causing "Missing \begin{document}".

```
1671 \newcommand*{\LWR@ProvidesPackageDropB}{%
1672 % \ProcessOptions\relax% original LaTeX code
1673 \let\ds@\@empty%          from the original \ProcessOptions
1674 \edef\@curroptions{}%     lwarp modification to \ProcessOptions
1675 \@process@ptions\relax%   from the original \ProcessOptions
1676 }
```

\LWR@ProvidesPackageDrop {⟨*pkgname*⟩} [⟨*version*⟩]

Ignores the original package and uses lwarp's version instead. Drops/discards all options.

```
1677 \NewDocumentCommand{\LWR@ProvidesPackageDrop}{m o}{
```

Declare the package:

```
1678 \LWR@ProvidesPackageDropA{#1}{#2}
```

Ignore all options:

```
1679 \DeclareOption*{}
```

Process the options:

```
1680 \LWR@ProvidesPackageDropB
1681 }
```

```
1682 \end{warpHTML}
```

34   File handles

Defines file handles for writes.

1683 `\begin{warpall}`

`\LWR@quickfile` For quick temporary use only. This is reused in several places.

1684 `\newwrite\LWR@quickfile%`

1685 `\end{warpall}`

1686 `\begin{warpHTML}`

`\LWR@lateximagesfile` For `<project>-images.txt`:

1687 `\newwrite\LWR@lateximagesfile`

1688 `\end{warpHTML}`

35   Include a file

During HTML output, `\include{<filename>}` causes the following to occur:

1. lwarp creates `<filename>_html_inc.tex` whose contents are:

   `\input <filename>.tex`

2. `<filename>_html_inc.tex` is then `\included` instead of `<filename>.tex`.

3. `<filename>_html_inc.aux` is automatically generated and used by LATEX.

1689 `\begin{warpHTML}`

`\@include` {⟨*filename*⟩} Modified to load `_html_inc` files.

(Below, `\clearpage` caused missing text, and was changed to `\newpage`.)

```
1690 \def\@include#1 {%
1691 \immediate\openout\LWR@quickfile #1_html_inc.tex% lwarp
1692 \immediate\write\LWR@quickfile{\string\input{#1.tex}}% lwarp
1693 \immediate\closeout\LWR@quickfile% lwarp
1694 \LWR@maybe@orignewpage% changed from clearpage
1695 \if@filesw
1696     \immediate\write\@mainaux{\string\@input{#1_html_inc.aux}}% changed
1697 \fi
1698 \@tempswatrue
1699 \if@partsw
1700     \@tempswafalse
1701     \edef\reserved@b{#1}%
1702     \@for\reserved@a:=\@partlist\do
1703     {\ifx\reserved@a\reserved@b\@tempswatrue\fi}%
```

```
1704 \fi
1705 \if@tempswa
1706     \let\@auxout\@partaux
1707     \if@filesw
1708         \immediate\openout\@partaux #1_html_inc.aux % changed
1709         \immediate\write\@partaux{\relax}%
1710     \fi
1711     \@input@{#1_html_inc.tex}% changed
1712     \LWR@maybe@orignewpage% changed from clearpage
1713     \@writeckpt{#1}%
1714     \if@filesw
1715         \immediate\closeout\@partaux
1716     \fi
1717 \else
1718     \deadcycles\z@
1719     \@nameuse{cp@#1}%
1720 \fi
1721 \let\@auxout\@mainaux%
1722 }

1723 \end{warpHTML}
```

36   Copying a file

1724 \begin{warpHTML}

\LWR@copyfile {⟨*source filename*⟩} {⟨*destination filename*⟩}

Used to copy the .toc file to .sidetoc to re-print the toc in the sideToc navigation pane.

```
1725 \newwrite\LWR@copyoutfile   % open the file to write to
1726 \newread\LWR@copyinfile     % open the file to read from
1727
1728 \newcommand*\LWR@copyfile}[2]{%
1729     \LWR@traceinfo{LWR@copyfile: copying #1 to #2}
1730
1731     \immediate\openout\LWR@copyoutfile=#2
1732     \openin\LWR@copyinfile=#1
1733     \begingroup\endlinechar=-1
1734     \makeatletter
1735
1736     \LWR@traceinfo{LWR@copyfile: about to loop}
1737
1738     \loop\unless\ifeof\LWR@copyinfile
1739         \LWR@traceinfo{LWR@copyfile: one line}
1740       \read\LWR@copyinfile to\LWR@fileline % Read one line and store it into \LWR@fileline
1741 %   \LWR@fileline\par                  % print the content into the pdf
1742 % print the content:
1743       \immediate\write\LWR@copyoutfile{\unexpanded\expandafter{\LWR@fileline}}%
1744     \repeat
1745     \immediate\closeout\LWR@copyoutfile
1746     \LWR@traceinfo{LWR@copyfile: done}
1747     \endgroup
1748 }

1749 \end{warpHTML}
```

37 Debugging messages

HTML comments   To have the HTML output include additional HTML comments, such as which `<div>` is closing, use

> `\booltrue{HTMLDebugComments}`

debugging information   To have debug information written to the log, use

> `\tracinglwarp`

for HTML & PRINT: 1750 `\begin{warpall}`

LWR@tracinglwarp (*bool*)   True if tracing is turned on.

         1751 `\newbool{LWR@tracinglwarp}`

`\tracinglwarp`   Turns on the debug tracing messages.

         1752 `\newcommand{\tracinglwarp}{\booltrue{LWR@tracinglwarp}}`

`\LWR@traceinfo` {⟨*text*⟩}   If tracing is turned on, writes the text to the `.log` file.

```
1753 \newcommand{\LWR@traceinfo}[1]{%
1754     \ifbool{LWR@tracinglwarp}%
1755     {%
1756         \typeout{*** lwarp: #1}%
1757     }%
1758     {}%
1759 }
```

HTMLDebugComments (*bool*)   Add comments in HTML about closing `<div>`s, sections, etc.
Default: `false`
```
1760 \newbool{HTMLDebugComments}
1761 \boolfalse{HTMLDebugComments}
```

If `\tracinglwarp`, show where preamble hooks occur:

```
1762 \AfterEndPreamble{
1763 \LWR@traceinfo{AfterEndPreamble}
1764 }
1765
1766 \AtBeginDocument{
1767 \LWR@traceinfo{AtBeginDocument}
1768 }
1769 \end{warpall}
```

38 Defining print and HTML versions of macros and environments

The following refers to defining objects inside lwarp, and may also be of some use for package authors to adapt their packages for lwarp. The following is not for the user's document.

Many macros and environments must be provided as both print and HTML versions.

While generating the print version of a document, the original macros as defined by LaTeX and its packages are used as-is.

While generating the HTML version of a document, the original macro or environment is redefined to call a new HTML version or a copy of the original print version. The new HTML versions of macros and environments are used most of the time. Copies of the print versions are used inside a lateximage environment, which draws and remembers an image of the printed output, and also several other places.

The general structure for providing print and HTML versions of a macro or environment is as follows:

For a preexisting macro: An HTML version is provided with a special name, inside a warpHTML environment, then \LWR@formatted is used to redefine and patch various macros:

```
\begin{warpHTML}
\newcommand{\LWR@HTML@name}{...}

\LWR@formatted{name}
\end{warpHTML}
```

\LWR@formatted{name} copies the original print version to a new name \LWR@print@<name>, then redefines \name to use either the print or HTML version depending on which mode lwarp is using.

For a preexisiting environment: The process is similar. Note the use of \LWR@formattedenv instead of \LWR@formatted.

```
\begin{warpHTML}
\newenvironment{LWR@HTML@name}{...}{..}

\LWR@formattedenv{name}
\end{warpHTML}
```

For a new macro or environment: The print version is defined inside warpall, so that it can also be seen and modified by during HTML outut.

```
\begin{warpall}
\newcommand{\name}{...}% The print version.
\end{warpall}

\begin{warpHTML}
\newcommand{\LWR@HTML@name}{...}

\LWR@formatted{name}
\end{warpHTML}
```

Similar for an environment, using \formattedenv.

In general, \LWR@formatted or \LWR@formattedenv are placed inside a warpHTML environment, and while producing an HTML document they do the following:

- Macros are modified:

  1. The pre-existing print version \name is saved as \LWR@print@<name>, unless \LWR@print@<name> is already defined.

  2. The original \name is redefined to call either the print or HTML version depending on which format is in use at the moment, as set by \LWR@formatting, which is defined as either "print" or "HTML".

- When lwarp is producing a print document, the original definitions are used, as well as any new definitions defined in warpall above.

- When lwarp is generating HTML output, \LWR@formatting is set to "HTML", and \name is directed to \LWR@HTML@<name>. For an environment, \endname is directed to \endLWR@HTML@<name>.

- When lwarp is generating HTML output but enters a lateximage environment, or for some other reason needs to draw images using the original print defintions, \LWR@formatting is changed to "print" and \name is then redirected to \LWR@print@<name>, which was the original \name.

- Since the new \name does not process any arguments, they are processed by \LWR@print@name or \LWR@HTML@name.

Expandable versions are also provided as well. These usually are necessary for anything which could appear inside a tabular, without which a "Misplaced \omit"

⚠ Misplaced \omit error   error may occur.

>   \LWR@expandableformatted
>   \LWR@expandableformattedenv

(Older versions of lwarp used \LetLtxMacro for everything, but this could fail when using macros defined by xparse. This older system is still in use for many definitions.)

Print or disabled versions:

for HTML & PRINT: 1770 \begin{warpall}

1771 \newcommand*{\LWR@formatted}[1]{}
1772 \newcommand*{\LWR@expandableformatted}[1]{}
1773 \newcommand*{\LWR@formattedenv}[1]{}
1774 \newcommand*{\LWR@expandableformattedenv}[1]{}

1775 \end{warpall}

for HTML output: HTML versions:

1776 \begin{warpHTML}

\LWR@formatting   Remembers if selected print/HTML formatting.

Used while \LWR@restoreorigformatting, such as in an lateximage. May be set to either "print" or "HTML".

1777 \newcommand*{\LWR@formatting}{HTML}

`\LWR@formatted@checkname` {⟨*name*⟩}

> Verify that a print and HTML version exist.

```
1778 \newcommand*{\LWR@formatted@checkname}[1]{%
1779     \ifcsundef{#1}{%
1780         \ifcsundef{LWR@print@#1}{%
1781             \PackageError{lwarp}
1782             {%
1783                 \LWRbackslash#1 or \protect\LWR@print@#1\MessageBreak
1784                 must be defined before using \protect\LWR@formatted, etc%
1785             }
1786             {Perhaps #1 is misspelled.}
1787         }{\relax}%
1788     }{\relax}%
1789     \ifcsundef{LWR@HTML@#1}{%
1790         \PackageError{lwarp}
1791         {%
1792             \protect\LWR@HTML@#1 must be defined
1793             before using \protect\LWR@formatted, etc%
1794         }
1795         {Perhaps #1 is misspelled.}
1796     }{\relax}%
1797 }
```

`\LWR@formatted@checkendname` {⟨*name*⟩}

```
1798 \newcommand*{\LWR@formatted@checkendname}[1]{%
1799     \ifcsundef{end#1}{%
1800         \ifcsundef{endLWR@print@#1}{%
1801             \PackageError{lwarp}
1802             {%
1803                 \protect\end#1 or \protect\endLWR@print@#1\MessageBreak
1804                 must be defined before using \protect\LWR@formatted, etc%
1805             }
1806             {Perhaps #1 is misspelled.}
1807         }{\relax}%
1808     }{\relax}%
1809     \ifcsundef{endLWR@HTML@#1}{%
1810         \PackageError{lwarp}
1811         {%
1812             \protect\endLWR@HTML@#1 must be defined
1813             before using \protect\LWR@formatted, etc%
1814         }
1815         {Perhaps #1 is misspelled.}
1816     }{\relax}%
1817 }
```

`\LWR@formatted` {⟨*macroname*⟩}   No backslash in the macro name.

> If not yet defined, defines `\LWR@print@<name>` as the original print-mode `\<name>`.
> Also redefines `\<name>` to use `\LWR@<format>@<name>`, where <format> is set by
> `\LWR@formatting`, and is print or HTML.

```
1818 \renewcommand*{\LWR@formatted}[1]{%
1819     \LWR@formatted@checkname{#1}%
1820     \ifcsundef{LWR@print@#1}{%
1821         \csNewCommandCopycs{LWR@print@#1}{#1}%
```

```
1822    }{}%
1823    \ifcsundef{#1}{%
1824        \expandafter\newrobustcmd\csname #1\endcsname{%
1825            \@nameuse{LWR@\LWR@formatting @#1}%
1826        }%
1827    }{%
1828        \expandafter\renewrobustcmd\csname #1\endcsname{%
1829            \@nameuse{LWR@\LWR@formatting @#1}%
1830        }%
1831    }%
1832 }
```

\LWR@expandableformatted {⟨*macroname*⟩}   No backslash in the macro name.

An expandable version of \LWR@formatted.

```
1833 \renewcommand*{\LWR@expandableformatted}[1]{%
1834    \LWR@formatted@checkname{#1}%
1835    \ifcsundef{LWR@print@#1}{%
1836        \csNewCommandCopycs{LWR@print@#1}{#1}%
1837    }{}%
1838    \ifcsundef{#1}{%
1839        \expandafter\newcommand\csname #1\endcsname{%
1840            \@nameuse{LWR@\LWR@formatting @#1}%
1841        }%
1842    }{%
1843        \expandafter\renewcommand\csname #1\endcsname{%
1844            \@nameuse{LWR@\LWR@formatting @#1}%
1845        }%
1846    }%
1847 }
```

\LWR@formattedenv {⟨*environmentname*⟩}

If not yet defined, defines the environment LWR@print@<name> as the original
print-mode <name>. Also redefines the environment <name> to use environment
LWR@<format>@<name>, where <format> is set by \LWR@formatting, and is print
or HTML.

```
1848 \renewcommand*{\LWR@formattedenv}[1]{%
1849    \LWR@formatted@checkname{#1}%
1850    \LWR@formatted@checkendname{#1}%
1851    \ifcsundef{LWR@print@#1}{%
1852        \NewEnvironmentCopy{LWR@print@#1}{#1}%
1853    }{}%
1854    \DeclareDocumentEnvironment{#1}{}%
1855    {%
1856        \@nameuse{LWR@\LWR@formatting @#1}%
1857    }%
1858    {%
1859        \@nameuse{endLWR@\LWR@formatting @#1}%
1860    }%
1861 }
```

\LWR@expandableformattedenv {⟨*environmentname*⟩}

An expandable version of LWR@formattedenv.

```
1862 \renewcommand*{\LWR@expandableformattedenv}[1]{%
1863     \LWR@formatted@checkname{#1}%
1864     \LWR@formatted@checkendname{#1}%
1865     \ifcsundef{LWR@print@#1}{%
1866         \NewEnvironmentCopy{LWR@print@#1}{#1}%
1867     }{}%
1868     \DeclareExpandableDocumentEnvironment{#1}{}%
1869     {%
1870         \@nameuse{LWR@\LWR@formatting @#1}%
1871     }%
1872     {%
1873         \@nameuse{endLWR@\LWR@formatting @#1}%
1874     }%
1875 }

1876 \end{warpHTML}
```

39 HTML-conversion output modifications

These booleans modify the HTML output in various ways to improve conversion to EPUB or word processor imports.

for HTML & PRINT:
```
1877 \begin{warpall}
```

39.1 User-level controls

FormatEPUB (*bool*)   Changes HTML output for easy EPUB conversion via an external program. Removes
Default: false   per-file headers, footers, and nav. Adds footnotes per chapter/section.

```
1878 \newbool{FormatEPUB}
1879 \boolfalse{FormatEPUB}
```

FormatWP (*bool*)   Changes HTML output for easier conversion by a word processor. Removes headers
Default: false   and nav, prints footnotes per section, and also forces single-file output and turns
                        off HTML debug comments.

```
1880 \newbool{FormatWP}
1881 \boolfalse{FormatWP}
```

WPMarkFloats (*bool*)   Adds
Default: false
```
=== begin table ===

. . .

=== end ===
```
or
```
=== begin figure ===

. . .

=== end ===
```
around floats while formatting for word processors. This helps identify boundaries
of floats to be manually converted to word-processor frames and captions. [18]

---

[18]Perhaps some day word processors will have HTML import options for identifying <figure> and caption tags for figures and tables.

```
1882 \newbool{WPMarkFloats}
1883 \boolfalse{WPMarkFloats}
```

WPMarkMinipages (*bool*)   Adds
Default: false

```
=== begin minipage ===
. . .
=== end minipage ===
```

around minipages while formatting for word processors. This helps identify boundaries of minipages to be manually converted to word-processor frames.

```
1884 \newbool{WPMarkMinipages}
1885 \boolfalse{WPMarkMinipages}
```

WPMarkTOC (*bool*)   While formatting for word processors, adds
Default: true

```
=== table of contents ===
```

where the Table of Contents would have been. This helps identify where to insert the actual TOC.

If set false*, the actual toc is printed instead.*

```
1886 \newbool{WPMarkTOC}
1887 \booltrue{WPMarkTOC}
```

WPMarkLOFT (*bool*)   While formatting for word processors, adds
Default: false

```
=== list of figures ===    and/or
=== list of tables ===
```

where each of these lists would have been. This helps identify where to insert the actual lists.

If set false*, the actual lists are printed instead.*

```
1888 \newbool{WPMarkLOFT}
1889 \boolfalse{WPMarkLOFT}
```

WPMarkMath (*bool*)   While formatting for word processors, prints math as LaTeX code instead of creat-
Default: false   ing svg images or MATHJAX. This is useful for cut/paste into the *LibreOffice Writer TeXMaths*
extension.

```
1890 \newbool{WPMarkMath}
1891 \boolfalse{WPMarkMath}
```

WPTitleHeading (*bool*)   While formatting for word processors, true sets the document title to <h1>, which
Default: false   is expected for HTML documents, but also causes the lower-level section headings
to start at **Heading 2** when imported into LIBREOFFICE. Set to false to cause the
title to be plain text, and the section headings to begin at **Heading 1**.

See table 11 on page 190.

```
1892 \newbool{WPTitleHeading}
1893 \boolfalse{WPTitleHeading}
```

```
1894 \end{warpall}
```

39.2   Heading adjustments

If formatting the HTML for a word processor, adjust heading levels.

If `WPTitleHeading` is true, adjust so that `part` is **Heading 1**.

If `WPTitleHeading` is false, use `<h1>` for the title, and set `part` to **Heading 2**.

1895 `\begin{warpHTML}`

1896 `\AtBeginDocument{`
1897 `\ifbool{FormatWP}{`
1898 `\@ifundefined{chapter}{`
1899 `\ifbool{WPTitleHeading}{% part and section starting at h2`
1900 `\renewcommand*{\LWR@tagtitle}{h1}`
1901 `\renewcommand*{\LWR@tagtitleend}{/h1}`
1902 `\renewcommand*{\LWR@tagpart}{h2}`
1903 `\renewcommand*{\LWR@tagpartend}{/h2}`
1904 `\renewcommand*{\LWR@tagsection}{h3}`
1905 `\renewcommand*{\LWR@tagsectionend}{/h3}`
1906 `\renewcommand*{\LWR@tagsubsection}{h4}`
1907 `\renewcommand*{\LWR@tagsubsectionend}{/h4}`
1908 `\renewcommand*{\LWR@tagsubsubsection}{h5}`
1909 `\renewcommand*{\LWR@tagsubsubsectionend}{/h5}`
1910 `\renewcommand*{\LWR@tagparagraph}{h6}`
1911 `\renewcommand*{\LWR@tagparagraphend}{/h6}`
1912 `\renewcommand*{\LWR@tagsubparagraph}{%`
1913 `    span class=\textquotedbl{}subparagraph\textquotedbl%`
1914 `}`
1915 `\renewcommand*{\LWR@tagsubparagraphend}{/span}`
1916 `}% WPTitleHeading`
1917 `{% not WPTitleHeading, part and section starting at h1`
1918 `\renewcommand*{\LWR@tagtitle}{div class=\textquotedbl{}title\textquotedbl}`
1919 `\renewcommand*{\LWR@tagtitleend}{/div}`
1920 `\renewcommand*{\LWR@tagpart}{h1}`
1921 `\renewcommand*{\LWR@tagpartend}{/h1}`
1922 `\renewcommand*{\LWR@tagsection}{h2}`
1923 `\renewcommand*{\LWR@tagsectionend}{/h2}`
1924 `\renewcommand*{\LWR@tagsubsection}{h3}`
1925 `\renewcommand*{\LWR@tagsubsectionend}{/h3}`
1926 `\renewcommand*{\LWR@tagsubsubsection}{h4}`
1927 `\renewcommand*{\LWR@tagsubsubsectionend}{/h4}`
1928 `\renewcommand*{\LWR@tagparagraph}{h5}`
1929 `\renewcommand*{\LWR@tagparagraphend}{/h5}`
1930 `\renewcommand*{\LWR@tagsubparagraph}{h6}`
1931 `\renewcommand*{\LWR@tagsubparagraphend}{/h6}`
1932 `}% not WPTitleHeading`
1933 `}% chapter undefined`
1934 `{% chapter defined`
1935 `\ifbool{WPTitleHeading}{}`
1936 `{% not WPTitleHeading, part and chapter starting at h1`
1937 `\renewcommand*{\LWR@tagtitle}{div class=\textquotedbl{}title\textquotedbl}`
1938 `\renewcommand*{\LWR@tagtitleend}{/div}`
1939 `\renewcommand*{\LWR@tagpart}{h1}`
1940 `\renewcommand*{\LWR@tagpartend}{/h1}`
1941 `\renewcommand*{\LWR@tagchapter}{h2}`
1942 `\renewcommand*{\LWR@tagchapterend}{/h2}`
1943 `\renewcommand*{\LWR@tagsection}{h3}`
1944 `\renewcommand*{\LWR@tagsectionend}{/h3}`
1945 `\renewcommand*{\LWR@tagsubsection}{h4}`

```
1946 \renewcommand*{\LWR@tagsubsectionend}{/h4}
1947 \renewcommand*{\LWR@tagsubsubsection}{h5}
1948 \renewcommand*{\LWR@tagsubsubsectionend}{/h5}
1949 \renewcommand*{\LWR@tagparagraph}{h6}
1950 \renewcommand*{\LWR@tagparagraphend}{/h6}
1951 \renewcommand*{\LWR@tagsubparagraph}{span class=\textquotedbl{}subparagraph\textquotedbl}
1952 \renewcommand*{\LWR@tagsubparagraphend}{/span}
1953 }% not WPTitleHeading
1954 }% chapter defined
1955 }{}% FormatWP
1956 }% AtBeginDocument
```

```
1957 \end{warpHTML}
```

40   Remembering original formatting macros

`1958 \begin{warpHTML}`

Remember original definitions of formatting commands. Will be changed to HTML commands for most uses. Will be temporarily restored to original meaning inside any lateximage environment and inside a tabbing environment. Also nullify unused commands.

Some packages redefine \#, which is used to generate HTML, so the original must be remembered here.

```
1959 \chardef\LWR@origpound=`\#
```

```
1960 \let\LWR@origcomma\,
1961 \LetLtxMacro\LWR@origtilde~
1962 \LetLtxMacro\LWR@orignobreakspace\nobreakspace
1963 \let\LWR@orighfil\hfil
1964 \let\LWR@orighss\hss
1965 \let\LWR@origllap\llap
1966 \let\LWR@origrlap\rlap
1967 \let\LWR@orighfilneg\hfilneg
1968 \let\LWR@orighspace\hspace
1969
1970 \let\LWR@origrule\rule
1971
1972 \let\LWR@origmedskip\medskip
1973 \let\LWR@origbigskip\bigskip
```

libertinus-otf has too much kerning for \textquotedbl, causing an extra space.

```
1974 \LetLtxMacro\LWR@orig@@textquotedbl\textquotedbl
1975 \LetLtxMacro\LWR@orig@textquotedbl\LWR@orig@@textquotedbl
1976
1977 \AtEndPreamble{
1978 \IfPackageLoadedTF{libertinus-otf}{
1979    \renewcommand{\LWR@orig@textquotedbl}{\LWR@orig@@textquotedbl\kern-.15em}
1980     \LetLtxMacro\textquotedbl\LWR@orig@textquotedbl
1981 }{}
1982 }
```

```
1983 \LetLtxMacro\LWR@origttfamily\ttfamily
```

```
1984
1985 \LetLtxMacro\LWR@origem\em
1986
1987 \LetLtxMacro\LWR@orignormalfont\normalfont
1988
1989 \let\LWR@origonecolumn\onecolumn
1990
1991 \let\LWR@origsp\sp
1992 \let\LWR@origsb\sb
1993
1994 \LetLtxMacro\LWR@origunderline\underline

1995 \let\LWR@orignewpage\newpage
1996
1997 \let\LWR@origpagestyle\pagestyle
1998 \let\LWR@origthispagestyle\thispagestyle
1999 \LetLtxMacro\LWR@origpagenumbering\pagenumbering
2000
2001 \let\LWR@orignewline\newline
2002
2003 \AtBeginDocument{% in case packages change definitions
2004 \let\LWR@orig@trivlist\@trivlist
2005 \let\LWR@origtrivlist\trivlist
2006 \let\LWR@origendtrivlist\endtrivlist
2007 \LetLtxMacro\LWR@origitem\item
2008 \LetLtxMacro\LWR@origitemize\itemize
2009 \LetLtxMacro\LWR@endorigitemize\enditemize
2010 \LetLtxMacro\LWR@origenumerate\enumerate
2011 \LetLtxMacro\LWR@endorigenumerate\endenumerate
2012 \LetLtxMacro\LWR@origdescription\description
2013 \LetLtxMacro\LWR@endorigdescription\enddescription
2014 \let\LWR@orig@mklab\@mklab
2015 \let\LWR@origmakelabel\makelabel
2016 \let\LWR@orig@donoparitem\@donoparitem
2017 \LetLtxMacro\LWR@orig@item\@item
2018 \let\LWR@orig@nbitem\@nbitem
2019 }
2020
2021 \let\LWR@origpar\par
2022
2023 \LetLtxMacro\LWR@origfootnote\footnote
2024 \let\LWR@orig@mpfootnotetext\@mpfootnotetext
2025
2026
2027 \AtBeginDocument{% in case packages change definition
2028 \LetLtxMacro\LWR@orighline\hline%
2029 \LetLtxMacro\LWR@origcline\cline%
2030 }

2031 \end{warpHTML}
```

41   Accents

Native LATEX accents such as \" will work, but many more kinds of accents are available when using Unicode-aware XƎLATEX and LuaLATEX. If using accents in section names which will become file names, it is recommended to use the LATEX accents such as \" and \v instead of Unicode accents. The LATEX accents will have

the accents stripped when creating the filenames, whereas the Unicode accents will appear in the file names, which may cause issues with some operating systems.

for HTML output: 2032 `\begin{warpHTML}`

Without `\AtBeginDocument`, `\t` was being re-defined somewhere.

2033 `\AtBeginDocument{`

The following are restored for print when inside a lateximage.

For Unicode engines, only `\t` needs to be redefined:

2034 `\LetLtxMacro\LWR@origtie\t`

For PDF LATEX, additional work is required:

```
2035 \ifPDFTeX% pdflatex or dvi latex
2036 \LetLtxMacro\LWR@origgraveaccent\`
2037 \LetLtxMacro\LWR@origacuteaccent\'
2038 \LetLtxMacro\LWR@origcircumflexaccent\^
2039 \LetLtxMacro\LWR@origtildeaccent\~
2040 \LetLtxMacro\LWR@origmacronaccent\=
2041 \LetLtxMacro\LWR@origbreve\u
2042 \LetLtxMacro\LWR@origdotaccent\.
2043 \LetLtxMacro\LWR@origdiaeresisaccent\"
2044 \LetLtxMacro\LWR@origdoubleacuteaccent\H
2045 \LetLtxMacro\LWR@origcaronaccent\v
2046 \LetLtxMacro\LWR@origdotbelowaccent\d
2047 \LetLtxMacro\LWR@origcedillaaccent\c
2048 \LetLtxMacro\LWR@origmacronbelowaccent\b
```

The HTML redefinitions follow.

For PDF LATEX, Unicode diacritical marks are used:

```
2049 \renewcommand*{\`}[1]{#1\HTMLunicode{0300}}
2050 \renewcommand*{\'}[1]{#1\HTMLunicode{0301}}
2051 \renewcommand*{\^}[1]{#1\HTMLunicode{0302}}
2052 \renewcommand*{\~}[1]{#1\HTMLunicode{0303}}
2053 \renewcommand*{\=}[1]{#1\HTMLunicode{0304}}
2054 \renewcommand*{\u}[1]{#1\HTMLunicode{0306}}
2055 \renewcommand*{\.}[1]{#1\HTMLunicode{0307}}
2056 \renewcommand*{\"}[1]{#1\HTMLunicode{0308}}
2057 \renewcommand*{\H}[1]{#1\HTMLunicode{030B}}
2058 \renewcommand*{\v}[1]{#1\HTMLunicode{030C}}
2059 \renewcommand*{\d}[1]{#1\HTMLunicode{0323}}
2060 \renewcommand*{\c}[1]{#1\HTMLunicode{0327}}
2061 \renewcommand*{\b}[1]{#1\HTMLunicode{0331}}
2062 \fi
```

For all engines, a Unicode diacritical tie is used:

```
2063 \def\LWR@t#1#2{#1\HTMLunicode{0361}#2}
2064 \renewcommand*{\t}[1]{\LWR@t#1}
```

`\LWR@restoreorigaccents` Called from `\restoreoriginalformatting` when a lateximage is begun.

```
2065 \ifPDFTeX% pdflatex or dvi latex
2066 \newcommand*{\LWR@restoreorigaccents}{%
2067     \LetLtxMacro\`\LWR@origgraveaccent%
2068     \LetLtxMacro\'\LWR@origacuteaccent%
2069     \LetLtxMacro\^\LWR@origcircumflexaccent%
2070     \LetLtxMacro\~\LWR@origtildeaccent%
2071     \LetLtxMacro\=\LWR@origmacronaccent%
2072     \LetLtxMacro\u\LWR@origbreve%
2073     \LetLtxMacro\.\LWR@origdotaccent%
2074     \LetLtxMacro\"\LWR@origdiaeresisaccent%
2075     \LetLtxMacro\H\LWR@origdoubleacuteaccent%
2076     \LetLtxMacro\v\LWR@origcaronaccent%
2077     \LetLtxMacro\t\LWR@origtie%
2078     \LetLtxMacro\d\LWR@origdotbelowaccent%
2079     \LetLtxMacro\c\LWR@origcedillaaccent%
2080     \LetLtxMacro\b\LWR@origmacronbelowaccent%
2081 }%
2082 \else% XeLaTeX, LuaLaTeX:
2083 \newcommand*{\LWR@restoreorigaccents}{%
2084     \LetLtxMacro\t\LWR@origtie%
2085 }%
2086 \fi%
2087 }% AtBeginDocument


2088 \end{warpHTML}
```

42   Configuration files

42.1   Decide whether to generate configuration files

Configuration files are only written if processing the print version of the document, and not processing a pstool image. pstool uses an additional compile for each image using the original document's preamble, which includes lwarp, so the lwarp configuration files are turned off if -pstool is part of the \jobname.

Default to no configuration files:

```
2089 \LWR@excludecomment{LWRwriteconf}{writeconf}
```

Generate configuration files if print mode and not -pstool:

```
for PRINT output: 2090 \begin{warpprint}
                  2091 \fullexpandarg%
                  2092 \IfSubStr*{\jobname}{-pstool}
                  2093     {
                  2094         \PackageInfo{lwarp}{%
                  2095             Jobname with -pstool is found.\MessageBreak
                  2096             Not generating lwarp configuration files,%
                  2097         }
                  2098     }
                  2099     {
                  2100         \PackageInfo{lwarp}{Generating lwarp configuration files,}%
                  2101         \LWR@includecomment{LWRwriteconf}{writeconf}
                  2102     }
                  2103 \end{warpprint}
```

42.2 `<project>_html.tex`

*_html.tex (*file*)   Used to allow an HTML version of the document to exist alongside the print version.

Config file:
```
2104 \begin{LWRwriteconf}
2105 \immediate\openout\LWR@quickfile=\jobname_html.tex
2106 \immediate\write\LWR@quickfile{%
2107 \detokenize{\PassOptionsToPackage}%
2108 {warpHTML,BaseJobname=\jobname}{lwarp}%
2109 }
2110 \immediate\write\LWR@quickfile{%
2111 \detokenize{\input}\string{\jobname.tex\string }%
2112 }
2113 \immediate\closeout\LWR@quickfile
2114 \end{LWRwriteconf}
```

42.3 *lwarpmk* configuration files

Config file:
```
2115 \begin{LWRwriteconf}
```

\LWR@lwarpconfversion   The version number of the configuration file, allowing *lwarpmk* to detect an obsolete configuration file format. Incremented by one each time the configuration file format changes. (This is NOT the same as the lwarp version number.)

```
2116 \newcommand*{\LWR@lwarpconfversion}{2}% also in lwarpmk.lua
```

42.3.1   Helper macros

\LWR@shellescapecmd   The LaTeX compile option for shell escape, if used.

```
2117 \ifshellescape
2118    \def\LWR@shellescapecmd{--shell-escape }
2119 \else
2120    \def\LWR@shellescapecmd{}
2121 \fi
```

\LWR@compilecmd {⟨*engine*⟩} {⟨*suffix*⟩}

Used to form the basic compilation command for a document, adding the optional shell escape.

Engine is *pdflatex*, etc. Suffix is empty or _html

```
2122 \newcommand*{\LWR@compilecmd}[2]{%
2123    #1 \LWR@shellescapecmd \jobname#2%
2124 }
```

\LWR@addcompilecmd {⟨*cmd*⟩} {⟨*suffix*⟩}

Adds to the compilation command.

Cmd is *dvipdfmx*, etc. Suffix is empty or _html

```
2125 \newcommand*{\LWR@addcompilecmd}[2]{%
```

```
2126        \LWRopseq
2127        #1 \jobname#2%
2128 }
```

\LWR@unknownengine Error message if not sure which LATEX engine is being used.

```
2129 \newcommand*{\LWR@unknownengine}{%
2130        \PackageError{lwarp}%
2131            {Unknown LaTeX engine}%
2132            {%
2133                Lwarp only knows about pdflatex, DVI latex,
2134                xelatex, lualatex, and upLateX.%
2135            }%
2136 }
```

\LWR@latexmkvar {⟨*varname*⟩} {⟨*value*⟩}

Adds a *latexmk* variable assignment.

```
2137 \newcommand*{\LWR@latexmkvar}[2]{%
2138        -e
2139        \LWRopquote%
2140        \LWRdollar #1=q/#2/%
2141        \LWRopquote
2142 }
```

\LWR@latexmkcmd {⟨*latexmk options*⟩}

Sets a call to *latexmk* with the given options, possibly adding --shell-escape, and also adding the indexing program.

```
2143 \newcommand*{\LWR@latexmkcmd}[1]{%
2144        latexmk  \space \LWR@shellescapecmd \space  #1 \space
2145        -recorder \space
2146        \LWR@latexmkvar{makeindex}{\LWR@LatexmkIndexCmd}%
2147 }
```

\LWR@latexmkdvipdfm {⟨*dvipdfm or dvipdfmx*⟩}

Adds the options settings for *dvipdfm* or *dvipdfmx*.

```
2148 \newcommand*{\LWR@latexmkdvipdfm}[1]{%
2149        -pdfdvi \space
2150        \LWR@latexmkvar{dvipdf}{%
2151            #1
2152            \@percentchar O
2153            -o \@percentchar D
2154            \@percentchar S%
2155        }
2156 }
```

\LWR@compileuplatex Sets compile options for upLATEX with ujarticle or related classes.

```
2157 \newcommand*{\LWR@compileuplatex}{
2158        \def\LWR@tempprintlatexcmd{%
2159            \LWR@compilecmd{uplatex}{}
```

```
2160        \LWR@addcompilecmd{dvipdfmx}{}
2161    }
2162    \def\LWR@tempHTMLlatexcmd{%
2163        \LWR@compilecmd{uplatex}{_html}
2164        \LWR@addcompilecmd{dvipdfmx}{_html}
2165    }
2166 }
```

\LWR@PrintLatexCmd If not set by the user, the following sets the command to use to compile the source
\LWR@HTMLLatexCmd   to PDF form.

If using *latexmk*, a complicated string is created, eventually resulting in something
such as:

For *xelatex* with `--shell-escape`:

```
[[latexmk  -xelatex  --shell-escape -recorder
   -e '$makeindex = q/makeindex -s lwarp.ist/'  <jobname>_html]]
```

For *dvipdfmx*:

```
[[latexmk  -pdfdvi  -e '$dvipdf=q/dvipdfmx %O -o %D %S/'
   -recorder
   -e '$makeindex=q/makeindex -s lwarp.ist/'   <jobname>_html]]
```

For the following, temporary values are computed, but the permanent values are
only set if the originals were not assigned by the user.

```
2167 \ifbool{LWR@latexmk}{
```

For *latexmk* with *pdflatex* or *lualatex*:

```
2168    \ifpdf
```

For *latexmk* with *pdflatex*:

```
2169        \ifPDFTeX
2170            \def\LWR@latexcmd{\LWR@latexmkcmd{-pdf -dvi- -ps-}}
2171        \else
```

For *latexmk* with *lualatex*:

```
2172            \ifLuaTeX
2173                \def\LWR@latexcmd{\LWR@latexmkcmd{-lualatex}}
2174            \else
2175                \LWR@unknownengine
2176            \fi
2177        \fi
2178    \else% \ifpdf
```

For *latexmk* with *xelatex* or DVI *latex*:

```
2179        \ifXeTeX
```

For *latexmk* with *xelatex*:

```
2180          \def\LWR@latexcmd{\LWR@latexmkcmd{-xelatex}}
2181      \else% \ifXeTeX
```

For *latexmk* with DVI *latex*:

```
2182          \ifbool{LWR@dvipdfm}{
2183              \def\LWR@latexcmd{%
2184                  \LWR@latexmkcmd{%
2185                      \LWR@latexmkdvipdfm{dvipdfm}%
2186                  }
2187              }
2188          }{
2189              \ifbool{LWR@dvipdfmx}{
2190                  \def\LWR@latexcmd{%
2191                      \LWR@latexmkcmd{%
2192                          \LWR@latexmkdvipdfm{dvipdfmx}%
2193                      }
2194                  }
2195              }{
2196                  \def\LWR@latexcmd{\LWR@latexmkcmd{-pdfps}}
2197              }
2198          }
2199      \fi
2200    \fi% \ifpdf
```

The final assignment if *latexmk*:

```
2201    \def\LWR@tempprintlatexcmd{\LWR@latexcmd \space \jobname}
2202    \def\LWR@tempHTMLlatexcmd{\LWR@latexcmd \space \jobname_html}
2203 }% latexmk
```

Without *latexmk*, the compiling command is simply the compiler name and the optional shell escape:

```
2204 {% not latexmk
2205    \ifpdf
```

For *pdflatex* or *lualatex*:

```
2206        \ifPDFTeX
```

For *pdflatex*:

```
2207          \def\LWR@tempprintlatexcmd{\LWR@compilecmd{pdflatex}{}}
2208          \def\LWR@tempHTMLlatexcmd{\LWR@compilecmd{pdflatex}{_html}}
2209      \else
2210          \ifLuaTeX
```

For *lualatex*:

```
2211              \def\LWR@tempprintlatexcmd{\LWR@compilecmd{lualatex}{}}
2212              \def\LWR@tempHTMLlatexcmd{\LWR@compilecmd{lualatex}{_html}}
2213          \else
2214              \LWR@unknownengine
2215          \fi
2216      \fi
2217    \else% \ifpdf
```

For DVI *latex* or *xelatex*:

```
2218        \ifXeTeX
```

For *xelatex*:

```
2219            \def\LWR@tempprintlatexcmd{\LWR@compilecmd{xelatex}{}}
2220            \def\LWR@tempHTMLlatexcmd{\LWR@compilecmd{xelatex}{_html}}
2221        \else
```

For DVI *latex*. Default to *dvips*, unless told to use *dvipdfm* or *dvipdfmx*:

```
2222            \ifbool{LWR@dvipdfm}{
```

For DVI *latex* with *dvipdfm*:

```
2223                \def\LWR@tempprintlatexcmd{%
2224                    \LWR@compilecmd{latex}{}
2225                    \LWR@addcompilecmd{dvipdfm}{}
2226                }
2227                \def\LWR@tempHTMLlatexcmd{%
2228                    \LWR@compilecmd{latex}{_html}
2229                    \LWR@addcompilecmd{dvipdfm}{_html}
2230                }
2231            }{
2232                \ifbool{LWR@dvipdfmx}{
```

For DVI *latex* with *dvipdfmx*:

```
2233                    \def\LWR@tempprintlatexcmd{%
2234                        \LWR@compilecmd{latex}{}
2235                        \LWR@addcompilecmd{dvipdfmx}{}
2236                    }
2237                    \def\LWR@tempHTMLlatexcmd{%
2238                        \LWR@compilecmd{latex}{_html}
2239                        \LWR@addcompilecmd{dvipdfmx}{_html}
2240                    }
2241                }{% dvips
```

For DVI *latex* with *dvips* and *ps2pdf*:

```
2242                    \def\LWR@tempprintlatexcmd{%
2243                        \LWR@compilecmd{latex}{}
2244                        \LWR@addcompilecmd{dvips}{}
2245                      \LWR@addcompilecmd{ps2pdf -dALLOWPSTRANSPARENCY}{}.ps
2246                    }
2247                    \def\LWR@tempHTMLlatexcmd{%
2248                        \LWR@compilecmd{latex}{_html}
2249                        \LWR@addcompilecmd{dvips}{_html}
2250                  \LWR@addcompilecmd{ps2pdf -dALLOWPSTRANSPARENCY}{_html}.ps
2251                    }
2252                }
2253            }
2254        \fi% \ifXeTeX
2255    \fi% \ifpdf
2256 }% latexmk
```

For ujarticle, utarticle, and related, using upLATEX and *dvipdfmx*:

```
2257 \IfClassLoadedTF{ujarticle}{\LWR@compileuplatex}{}
2258 \IfClassLoadedTF{ujbook}{\LWR@compileuplatex}{}
2259 \IfClassLoadedTF{ujreport}{\LWR@compileuplatex}{}
2260 \IfClassLoadedTF{utarticle}{\LWR@compileuplatex}{}
2261 \IfClassLoadedTF{utbook}{\LWR@compileuplatex}{}
2262 \IfClassLoadedTF{utreport}{\LWR@compileuplatex}{}
```

Only make the setting permanent if the original was empty:

```
2263 \ifdefempty{\LWR@PrintLatexCmd}{
2264     \def\LWR@PrintLatexCmd{\LWR@tempprintlatexcmd}
2265 }{}
2266 \ifdefempty{\LWR@HTMLLatexCmd}{
2267     \def\LWR@HTMLLatexCmd{\LWR@tempHTMLlatexcmd}
2268 }{}
```

\LWR@writeconf {⟨*filename*⟩}

Common code for each of lwarpmk.conf and <project>.lwarpmkconf. Each entry is a variable name, the equal sign, and a quoted string inside [[and]], which are *lua*'s long quote characters, allowing the use of single and double quotes inside.

```
2269 \newcommand{\LWR@writeconf}[1]{
2270 \ifcsdef{LWR@quickfile}{}{\newwrite{\LWR@quickfile}}
2271 \immediate\openout\LWR@quickfile=#1
2272 \immediate\write\LWR@quickfile{confversion = [[\LWR@lwarpconfversion]]}
2273 \ifbool{usingOSWindows}{
2274     \immediate\write\LWR@quickfile{opsystem = [[Windows]]}
2275 }{
2276     \immediate\write\LWR@quickfile{opsystem = [[Unix]]}
2277 }
2278 \immediate\write\LWR@quickfile{sourcename = [[\jobname]]}
2279 \immediate\write\LWR@quickfile{homehtmlfilename = [[\HomeHTMLFilename]]}
2280 \immediate\write\LWR@quickfile{htmlfilename = [[\HTMLFilename]]}
2281 \immediate\write\LWR@quickfile{imagesdirectory = [[\LWR@ImagesDirectory]]}
2282 \immediate\write\LWR@quickfile{imagesname = [[\LWR@ImagesName]]}
2283 \immediate\write\LWR@quickfile{latexmk = [[\ifbool{LWR@latexmk}{true}{false}]]}
2284 \immediate\write\LWR@quickfile{printlatexcmd = [[\LWR@PrintLatexCmd]]}
2285 \immediate\write\LWR@quickfile{HTMLlatexcmd = [[\LWR@HTMLLatexCmd]]}
2286 \immediate\write\LWR@quickfile{printindexcmd = [[\LWR@PrintIndexCmd]]}
2287 \immediate\write\LWR@quickfile{HTMLindexcmd = [[\LWR@HTMLIndexCmd]]}
2288 \immediate\write\LWR@quickfile{latexmkindexcmd = [[\LWR@LatexmkIndexCmd]]}
2289 \immediate\write\LWR@quickfile{glossarycmd = [[\LWR@GlossaryCmd]]}
2290 \immediate\write\LWR@quickfile{pdftotextenc = [[\LWR@pdftotextEnc]]}
2291 \immediate\closeout\LWR@quickfile
2292 }
2293
2294 \end{LWRwriteconf}
```

42.3.2 `lwarpmk.conf`

lwarpmk.conf (*file*) lwarpmk.conf is automatically (re-)created by the lwarp package when executing
pdflatex <project.tex>,
or similar for *xelatex* or *lualatex*, in print-document generation mode, which is the default unless the warpHTML option is given. lwarpmk.conf is then used by the utility *lwarpmk*.

Config file:
```
2295 \begin{LWRwriteconf}
2296
2297 \AtBeginDocument{\LWR@writeconf{lwarpmk.conf}}
2298
2299 \end{LWRwriteconf}
```

42.3.3 `<project>.lwarpmkconf`

project.lwarpmkconf (*file*)   A project-specific configuration file for *lwarpmk*.

The makeindex and xindy options have already been handled for lwarp.conf.

Config file:
```
2300 \begin{LWRwriteconf}
2301
2302 \AtBeginDocument{\LWR@writeconf{\jobname.lwarpmkconf}}
2303
2304 \end{LWRwriteconf}
```

42.4 `lwarp.css`

lwarp.css (*file*)   This is the base css layer used by lwarp.

This must be present both when compiling the project and also when distributing the HTML files.

Config file:
```
2305 \begin{LWRwriteconf}
2306 \begin{filecontents*}[overwrite]{lwarp.css}
2307 /*
2308   CSS stylesheet for the LaTeX Lwarp package
2309   Copyright 2016-2025 Brian Dunn — BD Tech Concepts LLC
2310 */
2311
2312
2313
2314
2315 /* Page layout */
2316
2317 div.sidetoccontainer {
2318     font-family: "DejaVu Serif", "Bitstream Vera Serif",
2319         "Lucida Bright", Georgia, serif;
2320     float: left ;
2321     width: 19%; /* room for border-right next to 80% main */
2322     margin: 0pt 0em 3ex 0pt ;
2323     border-right: 1px solid silver;
2324     border-bottom: 1px solid silver;
2325     background: #FAF7F4 ;
2326     font-size:.9em ;
2327     border-radius: 0px 0px 20px 0px ;
2328     max-height: 100vh ;
2329     overflow-y: auto ;
2330 }
2331
2332 div.sidetoccontents {
2333     overflow-y: auto ;
2334     width: 100% ;
2335     text-align: left ;
2336 }
```

```
2337
2338
2339 nav.sidetoc p {line-height:1.2 ; margin: 1ex .5em 1ex .5em ;
2340     text-indent: 0 ; }
2341
2342 nav.sidetoc p a {color:black ; font-size: .7em ;}
2343
2344 div.sidetoctitle {font-size: 1.2em; font-weight:bold; text-align:center;
2345     border-bottom: 1px solid silver ;     }
2346
2347 nav.sidetoc a:hover {text-decoration: underline ; }
2348
2349
2350 nav.topnavigation{
2351     text-align: left ;
2352     padding: 0.5ex 1em 0.5ex 1em ;
2353 /*     margin: 2ex 0em 3ex 0em ; */
2354     margin: 0 ;
2355     border-bottom: 1px solid silver ;
2356     border-top: 1px solid silver ;
2357     clear:both ;
2358 }
2359
2360 nav.botnavigation{
2361     text-align: left ;
2362     padding: 0.5ex 1em 0.5ex 1em ;
2363 /*     margin: 3ex 0em 2ex 0em ; */
2364     margin: 0 ;
2365     border-top: 1px solid silver ;
2366     border-bottom: 1px solid silver ;
2367     clear:both ;
2368 }
2369
2370
2371 header {
2372     line-height: 1.2 ;
2373     font-size: 1em ;
2374     border-bottom: 1px solid silver ;
2375     margin: 0px ;
2376     padding: 2ex 1em 2ex 1em ;
2377     text-align:left ;
2378 }
2379
2380
2381 footer {
2382     font-size: .85em ;
2383     line-height: 1.2 ;
2384     margin-top: 1ex ;
2385     border-top: 1px solid silver ;
2386     padding: 2ex 1em 2ex 1em ;
2387     clear:both ;
2388     text-align:left ;
2389 }
2390
2391
2392
2393 /* a fix for older browsers: */
2394 header, section, footer, aside, nav, main,
2395     article, figure { display: block; }
2396
```

```
2397
2398 A:link {color:#000080 ; text-decoration: none ; }
2399 A:visited {color:#800000 ; }
2400 A:hover {color:#000080 ; text-decoration: underline ;}
2401 A:active {color:#800000 ; }
2402
2403 a.tocbook {display: inline-block ; margin-left: 0em ;
2404     font-weight: bold ; margin-top: 1ex ; margin-bottom: 1ex ; }
2405 a.tocpart {display: inline-block ; margin-left: 0em ;
2406     font-weight: bold ;}
2407 a.tocchapter {display: inline-block ; margin-left: 0em ;
2408     font-weight: bold ;}
2409 a.tocsection {display: inline-block ; margin-left: 1em ;
2410     text-indent: -.5em ; font-weight: bold ; }
2411 a.tocsubsection {display: inline-block ; margin-left: 2em ;
2412     text-indent: -.5em ; }
2413 a.tocsubsubsection {display: inline-block ; margin-left: 3em ;
2414     text-indent: -.5em ; }
2415 a.tocparagraph {display: inline-block ; margin-left: 4em ;
2416     text-indent: -.5em ; }
2417 a.tocsubparagraph {display: inline-block ; margin-left: 5em ;
2418     text-indent: -.5em ; }
2419 a.tocfigure {margin-left: 0em}
2420 a.tocsubfigure {margin-left: 2em}
2421 a.toctable {margin-left: 0em}
2422 a.tocsubtable {margin-left: 2em}
2423 a.toctheorem {margin-left: 0em}
2424 a.toclstlisting {margin-left: 0em}
2425
2426 body {
2427     font-family: "DejaVu Serif", "Bitstream Vera Serif",
2428         "Lucida Bright", Georgia, serif;
2429     background: #FAF7F4 ;
2430     color: black ;
2431     margin:0em ;
2432     padding:0em ;
2433     font-size: 100% ;
2434     line-height: 1.2 ;
2435 }
2436
2437
2438 p {margin: 1.5ex 0em 1.5ex 0em ;}
2439 table p {margin: .5ex 0em .5ex 0em ;}
2440
2441 /* Holds a section number */
2442 span.sectionnumber { margin-right: 0em }
2443
2444 /* Inserted in front of index lines */
2445 span.indexitem {margin-left: 0em}
2446 span.indexsubitem {margin-left: 2em}
2447 span.indexsubsubitem {margin-left: 4em}
2448 div.indexheading {margin-top: 2ex ; font-weight: bold}
2449
2450 div.hidden, span.hidden { display: none ; }
2451
2452 kbd, span.texttt, p span.texttt {
2453     font-family: "DejaVu Mono", "Bitstream Vera Mono", "Lucida Console",
2454         "Nimbus Mono L", "Liberation Mono", "FreeMono", "Andale Mono",
2455         "Courier New", monospace;
2456     font-size: 100% ;
```

```
2457 }
2458
2459 pre { padding: 3pt ; }
2460
2461 span.strong, span.textbf, div.strong, div.textbf, table td.tdbfseries { font-weight: bold; }
2462
2463 span.textit, div.textit, table td.tditshape { font-style: italic; }
2464
2465 table td.tdbfit { font-weight: bold ; font-style:italic }
2466
2467 span.textnormal, div.textnormal {
2468     font-weight: normal;
2469     font-style: normal;
2470     font-variant: normal;
2471     font-variant-numeric: normal ;
2472     font-family: "DejaVu Serif", "Bitstream Vera Serif",
2473         "Lucida Bright", Georgia, serif;
2474 }
2475
2476 span.textmd, div.textmd { font-weight: normal; }
2477
2478 span.textup, div.textup {
2479     font-style: normal;
2480     font-variant: normal;
2481     font-variant-numeric: normal ;
2482 }
2483
2484
2485 /* For complex number i,j symbols */
2486 span.ijit {font-style: italic; font-variant: normal}
2487 span.ijup {font-style: normal; font-variant: normal}
2488
2489
2490 span.textsc, div.textsc {
2491     font-variant: small-caps;
2492     font-variant-numeric: oldstyle-nums ;
2493 }
2494
2495 span.textulc, div.textulc {
2496     font-variant: normal ;
2497     font-variant-numeric: normal ;
2498 }
2499
2500 span.textsl, div.textsl { font-style: oblique; }
2501
2502 span.textrm, div.textrm {
2503     font-family: "DejaVu Serif", "Bitstream Vera Serif",
2504     "Lucida Bright", Georgia, serif;
2505 }
2506
2507 span.textsf, div.textsf {
2508     font-family: "DejaVu Sans", "Bitstream Vera Sans",
2509         Geneva, Verdana, sans-serif ;
2510 }
2511
2512 /* nfssext-cfr lining figures */
2513 span.textln, div.textln {
2514     font-variant-numeric: lining-nums ;
2515 }
2516
```

```
2517 /* nfssext-cfr proportional figures */
2518 span.textp, div.textp {
2519     font-variant-numeric: proportional-nums ;
2520 }
2521
2522 /* nfssext-cfr tabular figures */
2523 span.textt, div.textt {
2524     font-variant-numeric: tabular-nums ;
2525 }
2526
2527 /* nfssext-cfr font weights */
2528 span.textdb, div.textdb {
2529     font-weight: 500 ;
2530 }
2531
2532 span.textsb, div.textsb {
2533     font-weight: 600 ;
2534 }
2535
2536 span.texteb, div.texteb {
2537     font-weight: 800 ;
2538 }
2539
2540 span.textub, div.textub {
2541     font-weight: 900 ;
2542 }
2543
2544 span.textlg, div.textlg {
2545     font-weight: 300 ;
2546 }
2547
2548 span.textel, div.textel {
2549     font-weight: 200 ;
2550 }
2551
2552 span.textul, div.textul {
2553     font-weight: 100 ;
2554 }
2555
2556
2557
2558 span.textcircled { border: 1px solid black ; border-radius: 1ex ; }
2559
2560 span.underline {
2561     text-decoration: underline ;
2562     text-decoration-skip: auto ;
2563 }
2564
2565 span.overline {
2566     text-decoration: overline ;
2567     text-decoration-skip: auto ;
2568 }
2569
2570 div.hrule { border-top: 1px solid silver }
2571
2572
2573 /* for vertical text: */
2574 div.verticalrl { writing-mode: vertical-rl }
2575 div.horizontaltb { writing-mode: horizontal-tb }
2576
```

```
2577
2578 /* for diagbox */
2579 div.diagboxtitleN { border-bottom: 1px solid gray }
2580 div.diagboxtitleS { border-top: 1px solid gray }
2581
2582 div.diagboxE {
2583     padding-left: 2em ;
2584     text-align: right ;
2585 }
2586
2587 div.diagboxW {
2588     padding-right: 2em ;
2589     text-align: left ;
2590 }
2591
2592
2593
2594 /* For realscripts */
2595 .supsubscript {
2596     display: inline-block;
2597     text-align:left ;
2598 }
2599
2600 .supsubscript sup,
2601 .supsubscript sub {
2602     position: relative;
2603     display: block;
2604     font-size: .7em;
2605     line-height: 1;
2606 }
2607
2608 .supsubscript sup {
2609     top: .3em;
2610 }
2611
2612 .supsubscript sub {
2613     top: .3em;
2614 }
2615
2616 div.attribution p {
2617     text-align: right ;
2618     font-size: 80%
2619 }
2620
2621 span.poemtitle {
2622   font-size: 120% ; font-weight: bold;
2623 }
2624
2625 pre.tabbing {
2626     font-family: "Linux Libertine Mono O", "Lucida Console",
2627         "Droid Sans Mono", "DejaVu Mono", "Bitstream Vera Mono",
2628         "Liberation Mono", "FreeMono", "Andale Mono",
2629         "Nimbus Mono L", "Courier New", monospace;
2630 }
2631
2632 blockquote {
2633     display: block ;
2634     margin-left: 2em ;
2635     margin-right: 2em ;
2636 }
```

```
2637
2638 /* quotchap is for the quotchap package */
2639 div.quotchap {
2640     display: block ;
2641     font-style: oblique ;
2642     overflow-x: auto ;
2643     margin-left: 2em ;
2644     margin-right: 2em ;
2645 }
2646
2647 blockquote p, div.quotchap p {
2648     line-height: 1.5;
2649     text-align: left ;
2650     font-size: .85em ;
2651 }
2652
2653 /* qauthor is for the quotchap package */
2654 div.qauthor {
2655   display: block ;
2656   text-align: right ;
2657   margin-left: auto ;
2658   margin-right: 2em ;
2659   font-size: 80% ;
2660   font-variant: small-caps;
2661 }
2662
2663 div.qauthor p {
2664   text-align: right ;
2665 }
2666
2667 div.epigraph, div.dictum {
2668   line-height: 1.2;
2669     text-align: left ;
2670     padding: 3ex 1em 0ex 1em ;
2671 /*     margin: 3ex auto 3ex auto ; */ /* Epigraph centered */
2672     margin: 3ex 1em 3ex auto ; /* Epigraph to the right */
2673 /*     margin: 3ex 1em 3ex 1em ; */ /* Epigraph to the left */
2674     font-size: .85em ;
2675     max-width: 27em ;
2676 }
2677
2678 div.epigraphsource, div.dictumauthor {
2679     text-align:right ;
2680     margin-left:auto ;
2681 /*     max-width: 50% ; */
2682     border-top: 1px solid #A0A0A0 ;
2683     padding-bottom: 3ex ;
2684     line-height: 1.2;
2685 }
2686
2687 div.epigraph p, div.dictum p { padding: .5ex ; margin: 0ex ;}
2688 div.epigraphsource p, div.dictumauthor p { padding: .5ex 0ex 0ex 0ex ; margin: 0ex ;}
2689 div.dictumauthor { font-style:italic }
2690
2691
2692 /* copyrightbox package: */
2693 div.copyrightbox { margin: .5ex .5em }
2694 div.copyrightbox p {margin: 0px .5em ; padding: 0px}
2695 div.copyrightboxnote {text-align: left ; font-size: 60%}
2696
```

```
2697
2698 /* lettrine package: */
2699 span.lettrine { font-size: 4ex ; float: left ; }
2700 span.lettrinetext { font-variant: small-caps ; }
2701
2702 /* ulem, soul, umoline packages: */
2703 span.uline {
2704     text-decoration: underline ;
2705     text-decoration-skip: auto ;
2706 }
2707
2708 span.uuline {
2709     text-decoration: underline ;
2710     text-decoration-skip: auto ;
2711     text-decoration-style: double ;
2712 }
2713
2714 span.uwave {
2715     text-decoration: underline ;
2716     text-decoration-skip: auto ;
2717     text-decoration-style: wavy ;
2718 }
2719
2720 span.sout {
2721     text-decoration: line-through ;
2722 }
2723
2724 span.oline {
2725     text-decoration: overline ;
2726     text-decoration-skip: auto ;
2727 }
2728
2729 span.xout {
2730     text-decoration: line-through ;
2731 }
2732
2733 span.dashuline {
2734     text-decoration: underline ;
2735     text-decoration-skip: auto ;
2736     text-decoration-style: dashed ;
2737 }
2738
2739 span.dotuline {
2740     text-decoration: underline ;
2741     text-decoration-skip: auto ;
2742     text-decoration-style: dotted ;
2743 }
2744
2745 span.letterspacing { letter-spacing: .2ex ; }
2746
2747 span.capsspacing {
2748     font-variant: small-caps ;
2749     letter-spacing: .1ex ;
2750 }
2751
2752 span.highlight { background: #F8E800 ; }
2753
2754
2755 /* keystroke package: */
2756 span.keystroke {
```

```
2757      border-style: outset ;
2758      padding: 0pt .5em 0pt .5em ;
2759 }
2760
2761
2762 html body {
2763   margin: 0 ;
2764   line-height: 1.2;
2765 }
2766
2767
2768 body div {
2769   margin: 0ex;
2770 }
2771
2772
2773 div.book, h1, h2, h3, h4, h5, h6, span.paragraph, span.subparagraph
2774 {
2775      font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
2776          "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
2777          "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
2778          "Times New Roman", serif;
2779      font-style: normal ;
2780      font-weight: bold ;
2781      text-align: left ;
2782 }
2783
2784 h1 {    /* title of the entire website, used on each page */
2785      text-align: center ;
2786      font-size: 2.5em ;
2787      padding: .4ex 0em 0ex 0em ;
2788 }
2789
2790 div.book {
2791      text-align: center ;
2792      font-size: 2.325em ;
2793      padding: .4ex 0em 0ex 0em ;
2794 }
2795
2796 h2 { font-size: 2.25em }
2797 h3 { font-size: 2em }
2798 h4 { font-size: 1.75em }
2799 h5 { font-size: 1.5em }
2800 h6 { font-size: 1.25em }
2801 span.paragraph {font-size: 1em ; font-variant: normal ;
2802      margin-right: 1em ; }
2803 span.subparagraph {font-size: 1em ; font-variant: normal ;
2804      margin-right: 1em ;   }
2805
2806 div.minisec {
2807      font-family: "DejaVu Sans", "Bitstream Vera Sans",
2808          Geneva, Verdana, sans-serif ;
2809      font-style: normal ;
2810      font-weight: bold ;
2811      text-align: left ;
2812 }
2813
2814 h1 {
2815   margin: 0ex 0em 0ex 0em ;
2816   line-height: 1.3;
```

```
2817   text-align: center ;
2818 }
2819
2820 h2 {
2821   margin: 1ex 0em 1ex 0em ;
2822   line-height: 1.3;
2823   text-align: center ;
2824 }
2825
2826 h3 {
2827   margin: 3ex 0em 1ex 0em ;
2828   line-height: 1.3;
2829 }
2830
2831 h4 {
2832   margin: 3ex 0em 1ex 0em ;
2833   line-height: 1.3;
2834 }
2835
2836 h5 {
2837   margin: 3ex 0em 1ex 0em ;
2838   line-height: 1.3;
2839 }
2840
2841 h6 {
2842   margin: 3ex 0em 1ex 0em ;
2843   line-height: 1.3;
2844 }
2845
2846
2847 div.titlepage {
2848   text-align: center ;
2849 }
2850
2851 .footnotes {
2852     text-align: left ;
2853     font-size: .85em ;
2854     margin: 3ex 2em 0ex 2em ;
2855     border-top: 1px solid silver ;
2856 }
2857
2858 .marginpar, .marginparblock {
2859     max-width: 50%;
2860     float: right ;
2861     clear: both ;
2862     text-align: left ;
2863     margin: 1ex 0.5em 1ex 1em ;
2864     padding: 1ex 0.5em 1ex 0.5em ;
2865     font-size: 85% ;
2866     border-top: 1px solid silver ;
2867     border-bottom: 1px solid silver ;
2868     overflow-x: auto ;
2869 }
2870
2871 .marginpar br { margin-bottom: 2ex ; }
2872
2873 div.marginblock, div.marginparblock {
2874     max-width:50%;
2875     min-width: 10em; /* room for caption */
2876     float:right;
```

```
2877        text-align:left;
2878        margin: 1ex 0.5em 1ex 1em ;
2879        padding: 1ex 0.5em 1ex 0.5em ;
2880        overflow-x: auto;
2881 }
2882
2883 div.marginblock div.minipage,
2884 div.marginparblock div.minipage {
2885        display: inline-block ;
2886        margin: 0pt auto 0pt auto ;
2887 }
2888
2889 div.marginblock div.minipage p ,
2890 div.marginparblock div.minipage p
2891        { font-size: 85%}
2892
2893 div.marginblock br ,
2894 div.marginparblock br
2895        { margin-bottom: 2ex ; }
2896
2897 main.bodycontainer {
2898        float: left ;
2899        width: 80% ;
2900 }
2901
2902 div.bodywithoutsidetoc main.bodycontainer {
2903        float: none ;
2904        width: 100% ;
2905 }
2906
2907 section.textbody div.footnotes{
2908        margin: 1ex 2em 2ex 2em ;
2909        border-bottom: 2px solid silver ;
2910 }
2911
2912 .footnoteheader {
2913        border-top: 2px solid silver ;
2914        margin-top: 3ex ;
2915        padding-top: 1ex ;
2916        font-weight: bold ;
2917 }
2918
2919 .mpfootnotes {
2920        text-align: left ;
2921        font-size: .85em ;
2922        margin-left: 1em ;
2923        border-top: 1px solid silver ;
2924 }
2925
2926 /* Remove footnote top border in the title page. */
2927 div.titlepage div.mpfootnotes {
2928        border-top: none ;
2929 }
2930
2931
2932
2933 ul, ol {
2934   margin: 1ex 1em 1ex 0em;
2935   line-height: 1.2;
2936 }
```

```
2937
2938 body dir, body menu {
2939   margin: 3ex 1em 3ex 0em;
2940   line-height: 1.2;
2941 }
2942
2943 li { margin: 0ex 0em 1ex 0em; }
2944
2945 li.p { display: inline ; }
2946
2947 html {
2948   margin: 0;
2949   padding: 0;
2950 }
2951
2952 .programlisting {
2953   font-family: "DejaVu Mono", "Bitstream Vera Mono", "Lucida Console",
2954       "Nimbus Mono L", "Liberation Mono", "FreeMono", "Andale Mono",
2955       "Courier New", monospace;
2956   margin: 1ex 0ex 1ex 0ex ;
2957   padding: .5ex 0pt .5ex 0pt ;
2958   overflow-x: auto;
2959 }
2960
2961 section.textbody>pre.programlisting {
2962 border-top: 1px solid silver ;
2963 border-bottom: 1px solid silver ;
2964 }
2965
2966
2967 div.displaymath {
2968     text-align: center ;
2969 }
2970
2971 div.displaymathnumbered {
2972     text-align: right ;
2973     margin-left: 5% ;
2974     margin-right: 5% ;
2975     min-width: 2.5in ;
2976 }
2977
2978 @media all and (min-width: 400px) {
2979     div.displaymathnumbered {
2980         margin-left: 10% ;
2981         margin-right: 10% ;
2982     }
2983 }
2984
2985 @media all and (min-width: 800px) {
2986     div.displaymathnumbered {
2987         margin-right: 20% ;
2988     }
2989 }
2990
2991 @media all and (min-width: 1200px) {
2992     div.displaymathnumbered {
2993         margin-right: 30% ;
2994     }
2995 }
2996
```

```
2997
2998 .inlineprogramlisting {
2999   font-family: "DejaVu Mono", "Bitstream Vera Mono", "Lucida Console",
3000         "Nimbus Mono L", "Liberation Mono", "FreeMono", "Andale Mono",
3001         "Courier New", monospace;
3002   overflow-x: auto;
3003 }
3004
3005 span.listinglabel {
3006     display: inline-block ;
3007     font-size: 70% ;
3008     width: 4em ;
3009     text-align: right ;
3010     margin-right: 2em ;
3011 }
3012
3013 div.abstract {
3014   margin: 2em 5% 2em 5% ;
3015   padding: 1ex 1em 1ex 1em ;
3016 /*  font-weight: bold ; */
3017   font-size: 90% ;
3018     text-align: left ;
3019 }
3020
3021 div.abstract dl {line-height:1.5;}
3022 div.abstract dt {color:#304070;}
3023
3024 div.abstracttitle{
3025     font-family: "URW Classico", Optima, "Linux Biolinum O",
3026         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
3027         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
3028     font-weight:bold;
3029     font-size:1.25em;
3030     text-align: center ;
3031 }
3032
3033 span.abstractrunintitle{
3034     font-family: "URW Classico", Optima, "Linux Biolinum O",
3035         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
3036         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
3037     font-weight:bold;
3038 }
3039
3040
3041 .verbatim {
3042     overflow-x: auto ;
3043 }
3044
3045 .alltt {
3046     overflow-x: auto ;
3047 }
3048
3049
3050 .bverbatim {
3051     margin: 1ex 0pt 1ex 0pt ;
3052     padding: .5ex 0pt .5ex 0pt ;
3053     overflow-x: auto ;
3054 }
3055
3056 .lverbatim {
```

```
3057       margin: 1ex 0pt 1ex 0pt ;
3058       padding: .5ex 0pt .5ex 0pt ;
3059       overflow-x: auto ;
3060 }
3061
3062 .fancyvrb {
3063       margin: 3ex 0pt 3ex 0pt ;
3064       font-family: "DejaVu Mono", "Bitstream Vera Mono", "Lucida Console",
3065          "Nimbus Mono L", "Liberation Mono", "FreeMono", "Andale Mono",
3066          "Courier New", monospace;
3067 }
3068
3069 .fancyvrblabel {
3070       font-size: .85em ;
3071       text-align: center ;
3072       font-weight: bold ;
3073       margin-top: 1ex ;
3074       margin-bottom: 1ex ;
3075 }
3076
3077
3078 .verse {
3079       font-family: "Linux Libertine Mono O", "Lucida Console",
3080          "Droid Sans Mono", "DejaVu Mono", "Bitstream Vera Mono",
3081          "Liberation Mono", "FreeMono", "Andale Mono",
3082          "Nimbus Mono L", "Courier New", monospace;
3083       margin-left: 1em ;
3084 }
3085
3086
3087 div.singlespace { line-height: 1.2 ; }
3088 div.onehalfspace { line-height: 1.5 ; }
3089 div.doublespace { line-height: 2 ; }
3090
3091
3092 /* Word processor format output: */
3093 div.wpfigure { border: 1px solid red ; margin: .5ex ; padding: .5ex ; }
3094 div.wptable { border: 1px solid blue ; margin: .5ex ; padding: .5ex ; }
3095 div.wpminipage { border: 1px solid green ; margin: .5ex ; padding: .5ex ;}
3096
3097
3098
3099
3100 /* Minipage environments, vertically aligned to top, center, bottom: */
3101 .minipage, .fminipage, .fcolorminipage {
3102       /* display: inline-block ; */
3103          /* Mini pages which follow each other will be tiled. */
3104       text-align:left;
3105       margin: .25em .25em .25em .25em;
3106       padding: .25em .25em .25em .25em;
3107       display: inline-flex;
3108       flex-direction: column ;
3109       overflow: auto;
3110 }
3111
3112 .inlineminipage {
3113       display: inline-block ;
3114       text-align: left
3115 }
3116
```

```
3117 /* Paragraphs in the flexbox did not collapse their margins. */
3118 /* Have not yet researched this. */
3119 .minipage p {margin: .75ex 0em .75ex 0em ;}
3120
3121 .fboxBlock .minipage, .colorbox .minipage, .colorboxBlock .minipage,
3122 .fcolorbox .minipage, .fcolorboxBlock .minipage
3123     {border: none ; background: none;}
3124
3125 .fbox, .fboxBlock { border: 1px solid black ; padding: 4pt }
3126
3127 .fbox, .fboxBlock, .fcolorbox, .fcolorboxBlock, .colorbox, .colorboxBlock,
3128 .fminipage, .fcolorminipage
3129     {display: inline-block}
3130
3131 .shadowbox, .shabox {
3132    border: 1px solid black;
3133    box-shadow: 3px 3px 3px #808080 ;
3134     border-radius: 0px ;
3135    padding: .4ex .3em .4ex .3em ;
3136    margin: 0pt .3ex 0pt .3ex ;
3137  display: inline-block ;
3138 }
3139
3140 .doublebox {
3141    border: 3px double black;
3142     border-radius: 0px ;
3143    padding: .4ex .3em .4ex .3em ;
3144    margin: 0pt .3ex 0pt .3ex ;
3145  display: inline-block ;
3146 }
3147
3148 .ovalbox, .Ovalbox {
3149    border: 1px solid black;
3150     border-radius: 1ex ;
3151    padding: .4ex .3em .4ex .3em ;
3152    margin: 0pt .3ex 0pt .3ex ;
3153  display: inline-block ;
3154 }
3155
3156 .Ovalbox { border-width: 2px ; }
3157
3158 .framebox {
3159    border: 1px solid black;
3160     border-radius: 0px ;
3161    padding: .3ex .2em 0ex .2em ;
3162    margin: 0pt .1ex 0pt .1ex ;
3163  display: inline-block ;
3164 }
3165
3166
3167 /* mdframed, tcolorbox, shadebox packages */
3168 .mdframed, .tcolorbox, .shadebox {
3169     padding: 0ex ;
3170     margin: 2ex 0em 2ex 0em ;
3171     border: 1px solid black ;
3172 }
3173
3174 .tcolorbox {
3175     border-radius: 10pt ;
3176     margin: 2ex 1em 2ex 1em ;
```

```
3177 }
3178
3179 .mdframed p, .tcolorbox p { padding: 0ex .5em 0ex .5em ; }
3180
3181 .mdframed dl, .tcolorbox dl { padding: 1ex .5em 0ex .5em ; }
3182
3183 .mdframedtitle, .tcolorboxtitle {
3184     padding: .5ex 0pt 0pt 0pt ;
3185     border-radius: 10pt 10pt 0pt 0pt ;
3186     display: block ;
3187     margin-bottom: 1ex ;
3188     border-bottom: 1px solid silver ;
3189 }
3190
3191 .tcolorboxsubtitle .tcolorbox {
3192     margin: 2ex 0em 2ex 0em ;
3193     border-radius: 0pt ;
3194 }
3195
3196 .mdframedsubtitle {
3197     display: block ;
3198 }
3199
3200 .mdframedsubsubtitle {
3201     display: block ;
3202 }
3203
3204 .mdtheorem {
3205     padding: 0ex .5em 0ex .5em ;
3206     margin: 3ex 5% 3ex 5% ;
3207 }
3208
3209
3210 /* framed package */
3211 .framed, pre.boxedverbatim, fcolorbox {
3212     margin: 3ex 0em 3ex 0em ;
3213   border: 1px solid black;
3214     border-radius: 0px ;
3215     padding: .3ex 1em 0ex 1em ;
3216   display: block ;
3217 }
3218
3219 .shaded {
3220     margin: 3ex 0em 3ex 0em ;
3221     padding: .3ex 1em .3ex 1em ;
3222     display: block ;
3223 }
3224
3225 .snugframed {
3226     margin: 3ex 0em 3ex 0em ;
3227   border: 1px solid black;
3228     border-radius: 0px ;
3229   display: block ;
3230 }
3231
3232 .framedleftbar {
3233     margin: 3ex 0em 3ex 0em ;
3234   border-left: 3pt solid black;
3235     border-radius: 0px ;
3236     padding: .3ex .2em .3ex 1em ;
```

```
3237   display: block ;
3238 }
3239
3240 .framedtitle {
3241     margin: 0em ;
3242     padding: 0em ;
3243     font-size: 130%
3244 }
3245
3246 .framedtitle p { padding: .3em }
3247
3248
3249 /* For the niceframe package: */
3250
3251 div.niceframe, div.curlyframe, div.artdecoframe, div.generalframe {
3252     padding: 1ex ;
3253     margin: 2ex auto ;
3254     border-radius: 2ex ;
3255 }
3256
3257 div.niceframe {
3258     border: 6px groove black ;
3259 }
3260
3261 div.curlyframe {
3262     border-left: 3px dotted black ;
3263     border-right: 3px dotted black ;
3264     border-radius: 6ex ;
3265 }
3266
3267 div.artdecoframe {
3268     border-left: 10px double black ;
3269     border-right: 10px double black ;
3270     border-radius: 6ex ;
3271 }
3272
3273 div.generalframe {
3274     border: 6px groove black ;
3275 }
3276
3277
3278 /* For beamerarticle: */
3279 div.beamerframe {
3280     margin: 3ex 1em 3ex 1em ;
3281   border: 1px solid gray;
3282     border-radius: 0px ;
3283     padding: .3ex 1em 0ex 1em ;
3284   display: block ;
3285 }
3286
3287
3288 dl {
3289   margin: 1ex 2em 1ex 0em;
3290   line-height: 1.3;
3291 }
3292
3293 li dl { margin-left: 2em }
3294
3295 dl dt {
3296     display: block ;
```

```
3297        float:left ;
3298        font-weight: bold;
3299        padding-right: 1em ;
3300 }
3301
3302 dl dd { display: block ; }
3303
3304 dl dd:after { content: "" ; display: block ; clear: both }
3305
3306 dl dd p { margin-top: 0em; }
3307
3308 dd ul, dd ol, dd dl {
3309        clear: both ;
3310 /*     padding-top: 1ex ; */
3311 }
3312
3313
3314 nav {
3315        font-family: "URW Classico", Optima, "Linux Biolinum O",
3316            "DejaVu Sans", "Bitstream Vera Sans",
3317            Geneva, Verdana, sans-serif ;
3318        margin-bottom: 4ex ;
3319 }
3320
3321 nav p {
3322        line-height: 1.2 ;
3323        margin-top:.5ex ;
3324        margin-bottom:.5ex;
3325        font-size: .9em ;
3326 }
3327
3328
3329
3330 img, img.hyperimage, img.borderimage {
3331        max-width: 600px;
3332        border: 1px solid silver;
3333        box-shadow: 3px 3px 3px #808080 ;
3334        padding: .5% ;
3335        margin: .5% ;
3336        background: none ;
3337 }
3338
3339 img.inlineimage{
3340        padding: 0px ;
3341        box-shadow: none ;
3342        border: none ;
3343        background: none ;
3344        margin: 0px ;
3345        display: inline-block ;
3346        border-radius: 0px ;
3347 }
3348
3349 img.logoimage{
3350        max-width: 300px ;
3351        box-shadow: 3px 3px 3px #808080 ;
3352        border: 1px solid black ;
3353        background:none ;
3354        padding:0 ;
3355        margin:.5ex ;
3356        border-radius: 10px ;
```

```
3357 }
3358
3359
3360 .section {
3361 /*
3362     To have each section float relative to each other:
3363 */
3364 /*
3365     display: block ;
3366     float: left ;
3367     position: relative ;
3368     background: white ;
3369     border: 1px solid silver ;
3370     padding: .5em ;
3371 */
3372     margin: 0ex .5em 0ex .5em ;
3373     padding: 0 ;
3374 }
3375
3376
3377 figure {
3378     margin: 5ex auto 5ex auto ;
3379     padding: 1ex 1em 1ex 1em ;
3380     overflow-x: auto ;
3381 }
3382
3383
3384 /* To automatically center images in figures: */
3385 /*
3386 figure img.inlineimage {
3387     margin: 0ex auto 0ex auto ;
3388     display: block ;
3389 }
3390 */
3391
3392 /* To automatically center minipages in figures: */
3393 /*
3394 figure div.minipage, figure div.minipage div.minipage {
3395     margin: 1ex auto 1ex auto ;
3396     display: block ;
3397 }
3398 */
3399
3400 figure figure { margin: 0pt }
3401
3402 figure div.minipage p { font-size: 85% ; }
3403
3404 figure.subfigure, figure.subtable  {
3405     display: inline-block ; margin: 3ex 1em 3ex 1em ;
3406 }
3407
3408 div.figurecaption .minipage { margin:0 ; padding: 0 }
3409
3410 /* for subcaptions: */
3411 figure div.minipage div.figurecaption {
3412     max-width: 100% ;    /* fallback if min() does not work */
3413     max-width: min(30em,100%)
3414 }
3415
3416 div.minipage figure { border: none ; box-shadow: none ; }
```

```
3417 div.minipage figure.table { margin: 0ex }
3418 div.minipage div.footnotes { margin: 1ex 2em 0ex 2em }
3419
3420 div.floatrow { text-align: center; }
3421
3422 div.floatrow figure { display: inline-block ; margin: 1ex 2% ; }
3423
3424 div.floatfoot { font-size: .85em ;
3425     border-top: 1px solid silver ; line-height: 1.2 ; }
3426
3427 /* Center if only one line, "start" align if more than one line: */
3428 div.figurecaption , .lstlistingtitle {
3429     font-size: .85em ;
3430     font-weight: bold ;
3431     text-align: start ;
3432     margin: 1ex auto;
3433     width: max-content;
3434     max-width: 100%;
3435 }
3436
3437 /* A marginblock is small, so always center and don't mess with the width. */
3438 div.marginblock div.figurecaption {
3439     width: 100% ;
3440     text-align: center ;
3441 }
3442
3443 figure.subfigure div.figurecaption, figure.subtable div.figurecaption  {
3444     border-bottom: none ; background: none ;
3445 }
3446
3447 div.nonfloatcaption {
3448     margin: 1ex auto 1ex auto ;
3449     font-size: .85em ;
3450     text-align: center ;
3451     font-weight: bold ;
3452 }
3453
3454 /* For a \RawCaption inside a minipage inside a figure's floatrow: */
3455 figure div.floatrow div.minipage div.figurecaption {
3456     border: none ;
3457     background: none ;
3458 }
3459
3460
3461 /* For packages such as float, rotfloat, and algorithm2e: */
3462
3463 figure.boxed, figure.boxruled {
3464     border: 1px solid black ;
3465 }
3466
3467 figure.ruled {
3468     border-top: 1px solid black ;
3469     border-bottom: 1px solid black ;
3470     border-left: 0px ;
3471     border-right: 0px ;
3472     border-radius: 0px ;
3473     background: none ;
3474     box-shadow: none ;
3475 }
3476
```

```
3477 figure.ruled div.figurecaption, figure.boxruled div.figurecaption {
3478     border-top: 1px solid silver ;
3479     border-bottom: 1px solid silver ;
3480 }
3481
3482
3483 table {
3484     margin: 1ex auto 1ex auto ;
3485     border-collapse: separate ;
3486     border-spacing: 0px ;
3487     line-height: 1.3 ;
3488     }
3489
3490 table > tbody > tr.hline > td {border-top: 1px solid #808080 ; margin-top: 0ex ;
3491     margin-bottom: 0ex ; } /* for \hline */
3492
3493 tr.tbrule td {border-top: 1px solid black ; margin-top: 0ex ;
3494     margin-bottom: 0ex ; } /* for \toprule, \bottomrule */
3495
3496 td {padding: .5ex .5em .5ex .5em ;}
3497
3498 table td.tdl { text-align: left ; vertical-align: middle ; }
3499 table td.tdc { text-align: center ; vertical-align: middle ; }
3500 table td.tdat { text-align: center ; vertical-align: middle ; padding: 0px ; margin: 0px ; }
3501 table td.tdbang { text-align: center ; vertical-align: middle ; }
3502 table td.tdr { text-align: right ; vertical-align: middle ; }
3503 table td.tdp { text-align: left ; vertical-align: bottom ; }
3504 table td.tdm { text-align: left ; vertical-align: middle ; }
3505 table td.tdb { text-align: left ; vertical-align: top ; }
3506
3507 table td.tvertbarl { border-left: 1px solid black }
3508 table td.tvertbarldouble { border-left: 4px double black }
3509 table td.tvertbarr { border-right: 1px solid black }
3510 table td.tvertbarrdouble { border-right: 4px double black }
3511
3512 table td.tvertbarldash { border-left: 1px dashed black }
3513 table td.tvertbarldoubledash { border-left: 2px dashed black }
3514 table td.tvertbarrdash { border-right: 1px dashed black }
3515 table td.tvertbarrdoubledash { border-right: 2px dashed black }
3516
3517 table td.tdcenter { text-align: center}
3518 table td.tdleft { text-align: left}
3519 table td.tdright { text-align: right}
3520
3521
3522 /* for cmidrules: */
3523 table td.tdrule {
3524     border-top: 1px solid #A0A0A0 ;
3525 }
3526
3527 table td.tdrulel {
3528     border-top-left-radius:.5em ;
3529     border-top: 1px solid #A0A0A0 ;
3530 }
3531
3532 table td.tdruler {
3533     border-top-right-radius:.5em ;
3534     border-top: 1px solid #A0A0A0 ;
3535 }
3536
```

```
3537 table td.tdrulelr {
3538     border-top-left-radius:.5em ;
3539     border-top-right-radius:.5em ;
3540     border-top: 1px solid #A0A0A0 ;
3541 }
3542
3543
3544 /* Margins of paragraphs inside table cells: */
3545 td.tdp p , td.tdprule p , td.tdP p , td.tdPrule p { padding-top: 1ex ;
3546     padding-bottom: 1ex ; margin: 0ex ; }
3547 td.tdm p , td.tmbrule p , td.tdM p , td.tdMrule p { padding-top: 1ex ;
3548     padding-bottom: 1ex ; margin: 0ex ; }
3549 td.tdb p , td.tdbrule p , td.tdB p , td.tdBrule p { padding-top: 1ex ;
3550     padding-bottom: 1ex ; margin: 0ex ; }
3551
3552 td.tdp , td.tdprule , td.tdP , td.tdPrule
3553     { padding: 0ex .5em 0ex .5em ; }
3554 td.tdm , td.tdmrule , td.tdM , td.tdMrule
3555     { padding: 0ex .5em 0ex .5em ; }
3556 td.tdb , td.tdbrule , td.tdB , td.tdBrule
3557     { padding: 0ex .5em 0ex .5em ; }
3558
3559
3560 /* table notes: */
3561 .tnotes {
3562     margin: 0ex 5% 1ex 5% ;
3563     padding: 0.5ex 1em 0.5ex 1em;
3564     font-size:.80em;
3565     text-align: left ;
3566 }
3567
3568 .minipage .tnotes {
3569     margin: 0pt ;
3570     padding: 0pt ;
3571 }
3572
3573 .tnotes dl dt p {margin-bottom:0px;}
3574
3575 .tnoteitemheader {margin-right: 1em;}
3576
3577
3578 /* for colortbl and cell color */
3579 div.cellcolor {
3580     width: 100% ;
3581     padding: .5ex .5em .5ex .5em ;
3582     margin: -.5ex -.5em -.5ex -.5em ;
3583 }
3584
3585
3586 /* for lyluatex */
3587 span.lyluatex {
3588     display: inline-block ;
3589 }
3590
3591 div.lyluatex p span.lateximagesource img {
3592     display: block ;
3593     margin-top: 3ex ;
3594     margin-bottom: 3ex ;
3595 }
3596
```

```
3597
3598 /* for bigdelim */
3599 .ldelim, .rdelim { font-size: 200% }
3600
3601
3602 /* center, flushleft, flushright environments */
3603 div.center{text-align:center;}
3604 div.center table {margin-left:auto;margin-right:auto;}
3605 div.flushleft{text-align:left;}
3606 div.flushleft table {margin-left:0em ; margin-right:auto;}
3607 div.flushright{text-align:right;}
3608 div.flushright table {margin-left:auto ; margin-right: 0em ;}
3609
3610
3611 /* Fancybox */
3612 div.Btrivlist table tr td {
3613     padding: .2ex 0em ;
3614 }
3615
3616
3617 /* program listing callouts: */
3618 span.callout {
3619     font-family: "DejaVu Sans", "Bitstream Vera Sans",
3620         Geneva, Verdana, sans-serif ;
3621     border-radius: .5em;
3622     background-color:black;
3623     color:white;
3624     padding:0px .25em 0px .25em;
3625     margin: 0 ;
3626     font-weight: bold;
3627     font-size:.72em ;
3628 }
3629
3630 div.programlisting pre.verbatim span.callout{
3631     font-size: .85em ;
3632 }
3633
3634 span.verbatim, span.verb {
3635     font-family: "DejaVu Mono", "Bitstream Vera Mono", "Lucida Console",
3636         "Nimbus Mono L", "Liberation Mono", "FreeMono", "Andale Mono",
3637         "Courier New", monospace;
3638 }
3639
3640
3641
3642 div.titlehead
3643 {
3644     text-align: left ;
3645     font-style: normal ;
3646     font-weight: normal ;
3647     font-style: normal ;
3648     font-size: .8em ;
3649     margin: 1ex 0em 1ex 0em ;
3650 }
3651
3652 div.subject
3653 {
3654     text-align: center ;
3655     font-style: normal ;
3656     font-weight: bold ;
```

```
3657        font-style: normal ;
3658        font-size: .8em ;
3659        margin: 1ex 0em 1ex 0em ;
3660 }
3661
3662 div.published
3663 {
3664        text-align: center ;
3665        font-variant: normal ;
3666        font-style: italic ;
3667        font-size: 1em ;
3668        margin: 1ex 0em 1ex 0em ;
3669 }
3670
3671 div.subtitle
3672 {
3673        text-align: center ;
3674        font-variant: normal ;
3675        font-style: italic ;
3676        font-size: 1.25em ;
3677        margin: 1ex 0em 1ex 0em ;
3678 }
3679
3680 div.subtitle p { margin: 1ex ; }
3681
3682 div.author
3683 {
3684        text-align: center ;
3685        font-variant: normal ;
3686        font-style: normal ;
3687        font-size: 1em ;
3688        margin: 1ex 0em 1ex 0em ;
3689 }
3690
3691 div.oneauthor {
3692        display: inline-block ;
3693        margin: 0ex 1em 0ex 1em ;
3694 }
3695
3696 /*
3697 div.author table {
3698        margin: 1ex auto 0ex auto ;
3699        background: none ;
3700 }
3701
3702 div.author table tbody tr td { padding: .25ex ; }
3703 */
3704
3705 span.affiliation {font-size: .85em ; font-variant: small-caps; }
3706
3707 div.titledate {
3708        text-align: center ;
3709        font-size: .85em ;
3710        font-style: italic;
3711        margin: 1ex 0em 1ex 0em ;
3712 }
3713
3714
3715
3716 /* for \LinkHome, \LinkPrevious, and \LinkNext: */
```

```
3717 a.linkhome { font-weight:bold ; font-size: 1em ;}
3718
3719
3720 div.lateximagesource { padding: 0px ; margin: 0px ; display: none; }
3721
3722 img.lateximage{
3723     padding: 0pt ;
3724     margin: 0pt ;
3725     box-shadow: none ;
3726     border: none ;
3727     background: none ;
3728     max-width: 100% ;
3729     border-radius: 0ex ;
3730     border: none ;
3731 }
3732
3733
3734
3735
3736 section.textbody { margin: 0ex 1em 0ex 1em ;}
3737
3738
3739 div.multicolsheading { -webkit-column-span: all;
3740     -moz-column-span: all; column-span: all; }
3741 div.multicols {
3742     -webkit-columns: 3 auto ;
3743     -moz-columns: 3 auto ;
3744     columns: 3 auto ;
3745 }
3746 div.multicols p {margin-top: 0ex}
3747
3748
3749 /* Used for xfrac and nicefrac: */
3750 span.numerator {
3751     font-size: 60% ;
3752     vertical-align: .4em ;
3753 }
3754
3755 span.denominator {
3756     font-size: 60%
3757 }
3758
3759
3760 /* Used for algorithm2e: */
3761 div.alg2evline{
3762     margin-left: 1em ;
3763     padding-left: 1em ;
3764     border-left: 1px solid black ;
3765     border-radius: 0px 0px 0px 1ex ;
3766 }
3767
3768 div.alg2evsline{
3769     margin-left: 1em ;
3770     padding-left: 1em ;
3771     border-left: 1px solid black ;
3772 }
3773
3774 div.alg2enoline{
3775     margin-left: 1em ;
3776     padding-left: 1em ;
```

```
3777 }
3778
3779 span.alg2elinenumber{
3780     margin-right: .5em ;
3781     font-size: 60% ;
3782     color: red ;
3783 }
3784
3785
3786 /* Used for algorithmicx: */
3787 span.floatright { float: right ; }
3788
3789
3790 /* keyfloat and tocdata: */
3791 .floatnotes {
3792     margin: 0ex 5% 0ex 5% ;
3793     padding: 0ex 1em 0ex 1em ;
3794     font-size:.80em ;
3795     text-align: left ;
3796 }
3797
3798 .authorartist{
3799     display:block ;
3800     font-size:.70em ;
3801     font-style: italic;
3802 }
3803
3804 nav .authorartist{ display:inline; }
3805
3806
3807
3808 /* Native LaTeX theorems: */
3809
3810 .theoremcontents {
3811     font-style: italic; margin-top: 3ex ; margin-bottom: 3ex ;
3812 }
3813
3814 .theoremlabel {
3815     font-style: normal; font-weight: bold ; margin-right: .5em ;
3816 }
3817
3818
3819
3820 /* theorem, amsthm, and ntheorem packages */
3821
3822 span.theoremheader,
3823 span.theoremheaderplain,
3824 span.theoremheaderdefinition,
3825 span.theoremheaderbreak,
3826 span.theoremheadermarginbreak,
3827 span.theoremheaderchangebreak,
3828 span.theoremheaderchange,
3829 span.theoremheadermargin
3830 {
3831     font-style:normal ; font-weight: bold ; margin-right: 1em ;
3832 }
3833
3834 span.amsthmnameplain,
3835 span.amsthmnamedefinition,
3836 span.amsthmnumberplain,
```

```
3837 span.amsthmnumberdefinition
3838 {
3839     font-style:normal ; font-weight: bold ;
3840 }
3841
3842
3843 span.amsthmnameremark,
3844 span.amsthmnumberremark
3845 {font-style:italic ; font-weight: normal ; }
3846
3847
3848 span.amsthmnoteplain,
3849 span.amsthmnotedefinition
3850 {font-style:normal ;}
3851
3852
3853 span.theoremheaderremark,
3854 span.theoremheaderproof,
3855 span.amsthmproofname
3856 {font-style:italic ; font-weight: normal ; margin-right: 1em ; }
3857
3858 span.theoremheadersc
3859 {
3860     font-style:normal ;
3861     font-variant: small-caps ;
3862     font-weight: normal ;
3863     margin-right: 1em ;
3864 }
3865
3866 .theoremendmark {float:right}
3867
3868 div.amsthmbodyplain, div.theorembodyplain, div.theorembodynonumberplain,
3869 div.theorembodybreak, div.theorembodynonumberbreak,
3870 div.theorembodymarginbreak,
3871 div.theorembodychangebreak,
3872 div.theorembodychange,
3873 div.theorembodymargin
3874 {
3875     font-style:italic;
3876     margin-top: 3ex ; margin-bottom: 3ex ;
3877 }
3878
3879 div.theorembodydefinition, div.theorembodyremark, div.theorembodyproof,
3880 div.theorembodyplainupright, nonumberplainuprightsc,
3881 div.amsthmbodydefinition, div.amsthmbodyremark,
3882 div.amsthmproof
3883 {
3884     font-style: normal ;
3885     margin-top: 3ex ; margin-bottom: 3ex ;
3886 }
3887
3888 span.amsthmnoteremark {}
3889
3890
3891 /* thmbox */
3892
3893 .thmbox {
3894     font-style: italic; margin-top: 3ex ; margin-bottom: 3ex ;
3895     border: 1px solid gray ;
3896     padding: 1ex ;
```

```
3897 }
3898
3899 .thmboxtitle {
3900     font-style: normal; font-weight: bold ; margin-right: .5em ;
3901     border-bottom: 1px solid gray ;
3902 }
3903
3904 span.thmboxproofname, span.thmboxexamplename {
3905     font-weight: bold ;
3906 }
3907
3908 div.thmboxproof, div.thmboxexample {
3909     font-size: 0.85em ;
3910     margin: 2ex ;
3911 }
3912
3913 div.thmboxleftbar {
3914     border-left: 2px solid black ;
3915     padding-left: 1em ;
3916 }
3917
3918
3919
3920 /* For the backnaur package: */
3921 div.backnaur {
3922     display: block ;
3923     margin: 2ex 2em 2ex 2em ;
3924 }
3925
3926 div.backnaur p {
3927     margin: .25ex 0ex .25ex 0ex ;
3928 }
3929
3930 div.backnaurprod {
3931     display: inline-block ;
3932     min-width: 8em ;
3933     text-align:right ;
3934 }
3935
3936 div.backnaurdesc {
3937     display: inline-block ;
3938 }
3939
3940
3941 /* For the notes package: */
3942 div.notesimportantnote, div.noteswarningnote, div.notesinformationnote {
3943     clear: both ;
3944     margin: 2ex 2em 2ex 2em ;
3945     border: 1px solid silver ;
3946 }
3947
3948 div.notesicon {
3949     float:left ;
3950     display: inline-block ;
3951     background: gold ;
3952     padding: 0ex 1em 0ex 1em ;
3953     margin-right: 1em ;
3954     font-weight: bold ;
3955 }
3956
```

```
3957 div.notescontents { font-style: italic }
3958
3959
3960 /* nolbreaks package: */
3961 span.nolbreaks { white-space: nowrap ; }
3962
3963
3964 /*
3965 For CSS LaTeX and related logos:
3966 Based on spacing demonstrated by the metafont package.
3967
3968 The subscripts are shrunk instead of lowered below the baseline,
3969 to avoid browser rendering errors with the line height in lists, etc.
3970 */
3971
3972 .latexlogofont {
3973     font-family: "Linux Libertine O", "Nimbus Roman No 9 L",
3974         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
3975     font-variant: normal ;
3976 }
3977
3978 .latexlogo {
3979     font-family: "Linux Libertine O", "Nimbus Roman No 9 L",
3980         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
3981 }
3982
3983 .latexlogosup {
3984   text-transform: uppercase;
3985   letter-spacing: .03em ;
3986   font-size: 0.7em;
3987   vertical-align: 0.25em;
3988   margin-left: -0.4em;
3989   margin-right: -0.15em;
3990 }
3991
3992 .latexlogosub {
3993   text-transform: uppercase;
3994 /*  vertical-align: -0.27ex; */
3995   margin-left: -0.08em;
3996   margin-right: -0.07em;
3997 /*  font-size: 1em; */
3998     font-size: .7em ;
3999 }
4000
4001 .latexlogotwoe {
4002   text-transform: none ;
4003   font-variant-numeric: oldstyle-nums ;
4004 }
4005
4006 .latexlogotwoesub {
4007   font-style:italic ;
4008 /*  vertical-align: -0.27ex; */
4009   margin-left: -0.11em;
4010   margin-right: -0.1em;
4011 /*  font-size: 1em; */
4012     font-size: .7em ;
4013 }
4014
4015 .xelatexlogo {
4016     font-family: "Linux Libertine O", "Nimbus Roman No 9 L",
```

```
4017        "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
4018     letter-spacing: .03em ;
4019 }
4020
4021 .xelatexlogosub {
4022 /*  vertical-align: -0.27ex; */
4023   margin-left: -0.0667em;
4024   margin-right: -.05em;
4025 /*  font-size: 1em; */
4026     font-size: .7em ;
4027   letter-spacing: .03em ;
4028 }
4029
4030 .amslogo {
4031     font-family: "TeXGyreChorus","URW Chancery L",
4032         "Apple Chancery","ITC Zapf Chancery","Monotype Corsiva",
4033         "Linux Libertine O", "Nimbus Roman No 9 L",  "FreeSerif",
4034         "Hoefler Text", Times, "Times New Roman", serif ;
4035     font-style: italic ;
4036 }
4037
4038 .lyxlogo {
4039     font-family: "URW Classico", Optima, "Linux Biolinum O",
4040         "DejaVu Sans", "Bitstream Vera Sans", Geneva,
4041         Verdana, sans-serif ;
4042 }
4043
4044
4045 /* Only display top and bottom navigation if a small screen: */
4046 /* Hide the sidetoc if a small screen: */
4047 nav.topnavigation { display:none; }
4048 nav.botnavigation { display:none; }
4049
4050 /* Only display the sidetoc's webpage title if a small screen */
4051 span.sidetocthetitle { display: none }
4052
4053 @media screen and (max-width: 100em) {
4054     div.multicols {
4055         -webkit-columns: 2 auto ;
4056         -moz-columns: 2 auto ;
4057         columns: 2 auto ;
4058     }
4059 }
4060
4061 @media screen and (max-width: 50em) {
4062     div.sidetoccontainer {
4063         float: none ;
4064         width: 100% ;
4065         padding: 0 ;
4066         border-radius: 0 ;
4067         border-bottom: 1px solid black ;
4068         border-top: 1px solid black ;
4069         box-shadow: none ;
4070     }
4071     span.sidetocthetitle { display: inline }
4072     nav.topnavigation { display:block }
4073     nav.botnavigation { display:block }
4074     main.bodycontainer { width: 100% }
4075     .marginpar {
4076         max-width: 100%;
```

```
4077        float: none;
4078        display:block ;
4079        margin: 1ex 1em 1ex 1em ;
4080    }
4081    div.multicols {
4082        -webkit-columns: 1 auto ;
4083        -moz-columns: 1 auto ;
4084        columns: 1 auto ;
4085    }
4086 }
4087
4088 @media print {
4089    body {
4090        font-family: "Linux Libertine O",
4091        "DejaVu Serif", "Bitstream Vera Serif",
4092        "Liberation Serif", "Nimbus Roman No 9 L",
4093        "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
4094    }
4095    div.sidetoccontainer { display:none; }
4096    nav.topnavigation { display: none; }
4097    nav.botnavigation { display: none; }
4098    main.bodycontainer { width: 100% }
4099 }
4100
4101 @media handheld {
4102    div.sidetoccontainer { display:none; }
4103    nav.topnavigation { display:block }
4104    nav.botnavigation { display:block }
4105    main.bodycontainer { width: 100% }
4106 }
4107
4108 @media projection {
4109    div.sidetoccontainer { display:none; }
4110    nav.topnavigation { display:block }
4111    nav.botnavigation { display:block }
4112    main.bodycontainer { width: 100% }
4113 }
4114 \end{filecontents*}
4115 % \end{Verbatim}% for syntax highlighting
4116 \end{LWRwriteconf}
```

42.5   `lwarp_sagebrush.css`

lwarp_sagebrush.css (*file*)   An optional CSS which may be used for a semi-modern appearance.

If used, this must be present both when compiling the project and also when distributing the HTML files.

Config file:
```
4117 \begin{LWRwriteconf}
4118 \begin{filecontents*}[overwrite]{lwarp_sagebrush.css}
4119 @import url("lwarp.css") ;
4120
4121
4122 A:link {color:#105030 ; text-decoration: none ; }
4123 A:visited {color:#705030 ; text-shadow:1px 1px 2px #a0a0a0;}
4124 A:hover {color:#006000 ; text-decoration: underline ; text-shadow:0px 0px 2px #a0a0a0;}
4125 A:active {color:#00C000 ; text-shadow:1px 1px 2px #a0a0a0;}
4126
```

```
4127
4128
4129 div.book, h1, h2, h3, h4, h5, h6, span.paragraph, span.subparagraph
4130 {
4131     font-family: "URW Classico", Optima, "Linux Biolinum O",
4132         "Linux Libertine O", "Liberation Serif",
4133         "Nimbus Roman No 9 L",  "FreeSerif",
4134         "Hoefler Text", Times, "Times New Roman", serif;
4135     font-variant: small-caps ;
4136     font-weight: normal ;
4137     color: #304070 ;
4138     text-shadow: 2px 2px 3px #808080;
4139 }
4140
4141 h1 {    /* title of the entire website, used on each page */
4142     font-variant: small-caps ;
4143     color: #304070 ;
4144     text-shadow: 2px 2px 3px #808080;
4145     background-color: #F7F7F0 ;
4146     background-image: linear-gradient(to bottom, #F7F7F0, #C0C0C4);
4147 }
4148
4149 h1 {
4150   border-bottom: 1px solid #304070;
4151 /*  border-top: 2px solid #304070; */
4152 }
4153
4154 h2 {
4155   border-bottom: 1px solid #304070;
4156 /*  border-top: 2px solid #304070; */
4157     background-color: #F7F7F0 ;
4158     background-image: linear-gradient(to bottom, #F7F7F0, #DAD0C0);
4159 }
4160
4161
4162
4163 div.abstract {
4164     background: #f5f5eb ;
4165     background-image: linear-gradient(to bottom, #f5f5eb, #C8C8B8);
4166
4167   border: 1px solid silver;
4168     border-radius: 1em ;
4169 }
4170
4171 div.abstract dl {line-height:1.5;}
4172 div.abstract dt {color:#304070;}
4173
4174 div.abstracttitle{
4175     font-family: "URW Classico", Optima, "Linux Biolinum O",
4176         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
4177         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
4178     font-weight:bold;
4179     font-variant: small-caps ;
4180     font-size:1.5em;
4181     border-bottom: 1px solid silver ;
4182     color: #304070 ;
4183     text-align: center ;
4184     text-shadow: 1px 1px 2px #808080;
4185 }
4186
```

```
4187 span.abstractrunintitle{
4188     font-family: "URW Classico", Optima, "Linux Biolinum O",
4189         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
4190         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
4191     font-weight:bold;
4192 }
4193
4194
4195 div.epigraph, div.dictum {
4196     background: #f5f5eb ;
4197     background-image: linear-gradient(to bottom, #f5f5eb, #C8C8B8);
4198
4199     border: 1px solid silver ;
4200     border-radius: 1ex ;
4201     box-shadow: 3px 3px 3px #808080 ;
4202 }
4203
4204
4205 .example {
4206     background-color: #f5f5eb ;
4207     background-image: linear-gradient(to bottom, #f5f5eb, #C8C8B8);
4208
4209 }
4210
4211 div.exampletitle{
4212     font-family: "URW Classico", Optima, "Linux Biolinum O",
4213         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
4214         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
4215     font-weight:bold;
4216     font-variant: small-caps ;
4217     border-bottom: 1px solid silver ;
4218     color: #304070 ;
4219     text-align: center ;
4220     text-shadow: 1px 1px 2px #808080;
4221 }
4222
4223
4224 .sidebar {
4225     background-color: #f5f5eb ;
4226     background-image: linear-gradient(to bottom, #f5f5eb, #C8C8B8);
4227
4228 }
4229
4230 div.sidebartitle{
4231     font-family: "URW Classico", Optima, "Linux Biolinum O",
4232         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
4233         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
4234     font-weight:bold;
4235     font-variant: small-caps ;
4236     border-bottom: 1px solid silver ;
4237     color: #304070 ;
4238     text-align: center ;
4239     text-shadow: 1px 1px 2px #808080;
4240 }
4241
4242
4243 .fancyvrblabel {
4244     font-family: "URW Classico", Optima, "Linux Biolinum O",
4245         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
4246         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
```

```
4247       font-weight:bold;
4248       font-variant: small-caps ;
4249       font-size: 1.5em ;
4250       color: #304070 ;
4251       text-align: center ;
4252       text-shadow: 1px 1px 2px #808080;
4253 }
4254
4255 div.minipage {
4256       background-color: #eeeee7 ;
4257       border: 1px solid silver ;
4258       border-radius: 1ex ;
4259 }
4260
4261 table div.minipage { background: none ; border: none ; }
4262
4263 div.framebox div.minipage {border:none ; background:none}
4264
4265 section.textbody > div.minipage {
4266       box-shadow: 3px 3px 3px #808080 ;
4267 }
4268
4269 div.fboxBlock div.minipage { box-shadow: none ; }
4270
4271 .framed .minipage , .framedleftbar .minipage {
4272       border: none ;
4273       background: none ;
4274       padding: 0ex ;
4275       margin: 0ex ;
4276 }
4277
4278 figure.figure .minipage, div.figurecaption .minipage { border: none; }
4279
4280 div.marginblock div.minipage ,
4281 div.marginparblock div.minipage
4282       { border: none; }
4283
4284 figure , div.marginblock {
4285       background-color: #eeeee7 ;
4286       border: 1px solid silver ;
4287       border-radius: 1ex ;
4288       box-shadow: 3px 3px 3px #808080 ;
4289 }
4290
4291 figure figure {
4292       border: 1px solid silver ;
4293       margin: 0em ;
4294       box-shadow: none ;
4295 }
4296
4297 /*
4298 div.figurecaption {
4299       border-top: 1px solid silver ;
4300       border-bottom: 1px solid silver ;
4301       background-color: #e8e8e8 ;
4302 }
4303 */
4304
4305
4306 div.table {
```

```
4307        box-shadow: 3px 3px 3px #808080 ;
4308 }
4309
4310 /*
4311 .tnotes {
4312      background: #e8e8e8;
4313      border: 1px solid silver;
4314 }
4315 */
4316
4317
4318 nav.topnavigation{
4319      background-color: #b0b8b0 ;
4320      background-image: linear-gradient(to bottom,#e0e0e0,#b0b8b0) ;
4321 }
4322
4323 nav.botnavigation{
4324      background-color: #b0b8b0 ;
4325      background-image: linear-gradient(to top,#e0e0e0,#b0b8b0) ;
4326 }
4327
4328
4329
4330 header{
4331      background-color: #F7F7F0 ;
4332      background-image: linear-gradient(to top, #F7F7F0, #b0b8b0);
4333 }
4334
4335 footer{
4336      background-color: #F7F7F0 ;
4337      background-image: linear-gradient(to bottom, #F7F7F0, #b0b8b0);
4338 }
4339
4340
4341
4342 div.sidetoccontainer {
4343      background-color: #F7F7F0 ;
4344      background-image: linear-gradient(to bottom, #F7F7F0, #C0C0C0);
4345      box-shadow: 3px 3px 3px #808080 ;
4346      }
4347
4348 div.sidetoctitle {color: #304070 ; }
4349
4350 nav.sidetoc a:hover {
4351      color:#006000 ;
4352      text-decoration: none ;
4353      text-shadow:0px 0px 2px #a0a0a0;
4354 }
4355
4356
4357 @media screen and (max-width: 45em) {
4358      div.sidetoccontainer { border-radius: 0 ; }
4359 }
4360
4361
4362 \end{filecontents*}
4363 % \end{Verbatim}% for syntax highlighting
4364 \end{LWRwriteconf}
```

42.6 `lwarp_formal.css`

lwarp_formal.css (*file*) An optional CSS which may be used for a more formal appearance.

If used, this must be present both when compiling the project and also when distributing the HTML files.

Config file:
```
4365 \begin{LWRwriteconf}
4366 \begin{filecontents*}[overwrite]{lwarp_formal.css}
4367 @import url("lwarp.css") ;
4368
4369
4370
4371 A:link {color:#802020 ; text-decoration:none; }
4372 A:visited {color:#802020 ; text-shadow:none ;}
4373 A:hover {color:#400000 ; text-shadow:none ;}
4374 A:active {color:#C00000 ; text-shadow:none ;}
4375
4376
4377 body {
4378     font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
4379         "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
4380         "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
4381         "Times New Roman", serif;
4382     background: #fffcf5;
4383 }
4384
4385 span.textrm {
4386     font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
4387         "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
4388         "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
4389         "Times New Roman", serif;
4390 }
4391
4392 span.textsf {
4393      font-family: "DejaVu Sans", "Bitstream Vera Sans",
4394         Geneva, Verdana, sans-serif ;
4395 }
4396
4397
4398
4399 div.book, h1, h2, h3, h4, h5, h6, span.paragraph, span.subparagraph
4400 {
4401     font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
4402         "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
4403         "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
4404         "Times New Roman", serif;
4405     color: #800000 ;
4406     text-shadow: none ;
4407 }
4408
4409 h1, h2 {
4410     background-color: #fffcf5 ;
4411     background-image: none ;
4412     border-bottom: 1px solid #808080;
4413 /*    border-top: 2px solid #808080; */
4414 }
4415
4416 div.abstracttitle {
4417     font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
```

```
4418          "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
4419          "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
4420          "Times New Roman", serif;
4421      color: black ;
4422      text-shadow: none ;
4423 }
4424
4425 span.abstractrunintitle {
4426      font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
4427          "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
4428          "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
4429          "Times New Roman", serif;
4430      color: black ;
4431      text-shadow: none ;
4432 }
4433
4434 div.abstract { font-size: 100% }
4435
4436 .sidebar {
4437      background: #fffcf5;
4438      background-image: none ;
4439   margin: 2em 5% 2em 5%;
4440   padding: 0.5em 1em;
4441   border: none ;
4442   border-top : 1px solid silver;
4443   border-bottom : 1px solid silver;
4444   font-size: 90% ;
4445 }
4446
4447 div.sidebartitle{
4448      font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
4449          "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
4450          "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
4451          "Times New Roman", serif;
4452      color: #800000 ;
4453      text-shadow: none ;
4454      border: none ;
4455 }
4456
4457 .example {
4458      background: #fffcf5;
4459      background-image: none ;
4460   margin: 2em 5% 2em 5%;
4461   padding: 0.5em 1em;
4462   border: none ;
4463   border-top : 1px solid silver;
4464   border-bottom : 1px solid silver;
4465 }
4466
4467 div.exampletitle{
4468      font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
4469          "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
4470          "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
4471          "Times New Roman", serif;
4472      color: #800000 ;
4473      text-shadow: none ;
4474      border: none ;
4475 }
4476
4477 div.fancyvrblabel{
```

```
4478      font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
4479          "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
4480          "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
4481          "Times New Roman", serif;
4482      color: #800000 ;
4483      text-shadow: none ;
4484      border: none ;
4485 }
4486
4487
4488
4489 figure {
4490      margin: 5ex 5% 5ex 5% ;
4491      padding: 1ex 1em 1ex 1em ;
4492      background-color: #fffcf5 ;
4493      overflow-x: auto ;
4494      border: none ;
4495 /*      border-top: 1px solid silver; */
4496 /*      border-bottom: 1px solid silver; */
4497 }
4498
4499
4500 div.figurecaption , .lstlisting {
4501      border: none ;
4502 /*      border-top: 1px solid silver ; */
4503 /*      border-bottom: 1px solid silver ; */
4504      background-color: #fffcf5 ;
4505 }
4506
4507 .tnotes {
4508      background: #fffcf5 ;
4509      border-top: 1px solid silver ;
4510      border-bottom: 1px solid silver ;
4511 }
4512
4513 .theorem {
4514          background: none ;
4515 }
4516
4517 .minipage {
4518      background-color: #fffcf5 ;
4519      border: none ;
4520 }
4521
4522 div.floatrow figure { border: none ; }
4523
4524 figure figure { border: none ; }
4525
4526
4527 nav.toc, nav.lof, nav.lot, nav.lol {
4528      font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
4529          "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
4530          "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
4531          "Times New Roman", serif;
4532 }
4533
4534 div.sidetoccontainer {
4535      font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
4536          "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
4537          "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
```

```
4538          "Times New Roman", serif;
4539      background-image: linear-gradient(to bottom, #fffcf5, #C0C0C0);
4540 }
4541
4542 div.sidetoctitle{
4543     color: #800000 ;
4544 }
4545
4546 header{
4547     background-color: #e0e0e0 ;
4548     background-image: linear-gradient(to top, #fffcf5, #b0b0b0);
4549     text-align:center ;
4550 }
4551
4552 footer{
4553     background-color: #e0e0e0 ;
4554     background-image: linear-gradient(to bottom, #fffcf5, #b0b0b0);
4555     padding: 2ex 1em 2ex 1em ;
4556     text-align:left ;
4557 }
4558
4559 nav.botnavigation {
4560     background: #dedcd5 ;
4561     border-top: 1px solid black ;
4562 }
4563 \end{filecontents*}
4564 % \end{Verbatim}% for syntax highlighting
4565 \end{LWRwriteconf}
```

42.7 `sample_project.css`

sample_project.css (*file*)   The project-specific CSS file. Use with `\CSSFilename`.

If used, this must be present both when compiling the project and also when distributing the HTML files.

Config file:
```
4566 \begin{LWRwriteconf}
4567 \begin{filecontents*}[overwrite]{sample_project.css}
4568 /* ( --- Start of project.css --- ) */
4569 /* ( --- A sample project-specific CSS file for lwarp --- ) */
4570
4571 /* Uncomment one of the following: */
4572 @import url("lwarp.css") ;
4573 /* @import url("lwarp_formal.css") ; */
4574 /* @import url("lwarp_sagebrush.css") ; */
4575
4576 /* Project-specific CSS setting follow here. */
4577 /* . . . */
4578
4579 /* ( --- End of project.css --- ) */
4580 \end{filecontents*}
4581 % \end{Verbatim}% for syntax highlighting
4582 \end{LWRwriteconf}
```

42.8 `lwarp.ist`

lwarp.ist (*file*)   Used to modify the index for lwarp.

This must be present when compiling the project, but does not need to be present when distributing the resulting HTML files.

The page compositor line is for **memoir**'s \specialindex.

Config file:
```
4583 \begin{LWRwriteconf}
4584 \begin{filecontents*}[overwrite]{lwarp.ist}
4585 preamble
4586 "\\begin{theindex}
4587   \\providecommand*\\lettergroupDefault[1]{}
4588   \\providecommand*\\lettergroup[1]{%
4589       \\par\\textbf{#1}\\par
4590       \\nopagebreak
4591   }
4592 "
4593 headings_flag 1
4594 heading_prefix "
4595   \\lettergroup{"
4596 heading_suffix "}"
4597 delim_0 ", \\hyperindexref{"
4598 delim_1 ", \\hyperindexref{"
4599 delim_2 ", \\hyperindexref{"
4600 delim_n "}, \\hyperindexref{"
4601 delim_r "} -- \\hyperindexref{"
4602 delim_t "}"
4603 page_compositor "."
4604 \end{filecontents*}
4605 % \end{Verbatim}% for syntax highlighting
4606 \end{LWRwriteconf}
```

42.9 `lwarp.xdy`

lwarp.xdy (*file*) Used to modify the index for **lwarp**.

This must be present when compiling the project, but does not need to be present when distributing the resulting HTML files.

See:
https://tex.stackexchange.com/questions/80300/
        how-can-i-convince-hyperref-and-xindy-to-play-together-nicely

Config file:
```
4607 \begin{LWRwriteconf}
4608 \begin{filecontents*}[overwrite]{lwarp.xdy}
4609 (require "tex/inputenc/latin.xdy")
4610 (merge-rule "\\PS *" "Postscript")
4611 (require "texindy.xdy")
4612 (require "page-ranges.xdy")
4613 (require "book-order.xdy")
4614 (define-location-class "arabic-page-numbers"
4615    ("arabic-numbers") :min-range-length 1)
4616 (require "makeindex.xdy")
4617 (define-attributes (("hyperindexref")))
4618 (markup-locref :open "\hyperindexref{" :close "}")
4619 (markup-locref :open "\hyperindexref{" :close "}" :attr "hyperpage")
4620 (markup-locref :open "\textbf{\hyperindexref{" :close "}}" :attr "textbf")
4621 (markup-locref :open "\textit{\hyperindexref{" :close "}}" :attr "textit")
4622 (define-location-class-order ("roman-page-numbers"
4623                   "arabic-page-numbers"
```

```
4624                    "alpha-page-numbers"
4625                    "Roman-page-numbers"
4626                    "Alpha-page-numbers"
4627                    "see"
4628                    "seealso"))
4629 \end{filecontents*}
4630 % \end{Verbatim}% for syntax highlighting
4631 \end{LWRwriteconf}
```

42.10 `lwarp_one_limage.cmd`

lwarp_one_limage.cmd (*file*)   Used by lwarp to help make lateximages when using WINDOWS.

This must be present when compiling the project, but does not need to be present when distributing the resulting HTML files.

The arguments are each of the three fields from `<project>-images.txt`, and also the base name of the source file.

MiKTeX does not allow file `lwarp_one_limage.cmd` to be created directly by *lwarpmk*, so `lwarp_one_limage.txt` is created instead, then copied to `lwarp_one_limage.cmd` by *lwarpmk*. This occurs each time *lwarpmk* used to create lateximages.

Config file:
```
4632 \begin{LWRwriteconf}
4633 \immediate\openout\LWR@quickfile=lwarp_one_limage.txt
4634 \immediate\write\LWR@quickfile{%
4635     pdfseparate -f \LWRpercent 1 -l \LWRpercent 1 \LWRpercent 4_html.pdf %
4636     \LWR@ImagesDirectory\OSPathSymbol lateximagetemp-\LWRpercent\LWRpercent d.pdf%
4637 }
4638 \immediate\write\LWR@quickfile{%
4639     pdfcrop --hires  --margins \LWRopquote0 1 0 0\LWRopquote\space %
4640     \LWR@ImagesDirectory\OSPathSymbol lateximagetemp-\LWRpercent 1.pdf %
4641     \LWR@ImagesDirectory\OSPathSymbol\LWRpercent 3.pdf%
4642 }
4643 \immediate\write\LWR@quickfile{%
4644   pdftocairo -svg -noshrink \LWR@ImagesDirectory\OSPathSymbol\LWRpercent 3.pdf %
4645     \LWR@ImagesDirectory\OSPathSymbol\LWRpercent 3.svg%
4646 }
4647 \immediate\write\LWR@quickfile{%
4648     del \LWR@ImagesDirectory\OSPathSymbol\LWRpercent 3.pdf%
4649 }
4650 \immediate\write\LWR@quickfile{%
4651     del \LWR@ImagesDirectory\OSPathSymbol lateximagetemp-\LWRpercent 1.pdf%
4652 }
4653 \immediate\write\LWR@quickfile{exit}
4654 \immediate\closeout\LWR@quickfile
4655 \end{LWRwriteconf}
```

42.11 `lwarp_mathjax.txt`

(*Emulates or patches code by* DAVIDE P. CERVONE, BRIAN DUNN.)

lwarp_mathjax.txt (*file*)   The default MATHJAX script used by lwarp when using MATHJAX. A recent version of MATHJAX is used, as served by the recommended repository. Adjustments are made to allow LaTeX to control the equation tags and provide for starred macros.

\MathJaxFilename determines which script file is copied into the HTML pages, and
defaults to lwarp_mathjax.txt. The script files must be present when compiling
the project, but do not need to be present when distributing the resulting HTML
files.

custom script   To generate a custom script, such as to use a local repository, copy lwarp_mathjax.txt
to a new file, make changes while keeping lwarp's adjustments for equation num-
bering and starred macros, and use \MathJaxFilename to select the new filename.

Config file:
```
4656 \begin{LWRwriteconf}
4657 \begin{filecontents*}[overwrite]{lwarp_mathjax.txt}
4658 <script>
4659 // Lwarp MathJax emulation code
4660 //
4661 // Based on code by Davide P. Cervone.
4662 // Equation numbering: https://github.com/mathjax/MathJax/issues/2427
4663 // Starred and ifnextchar macros: https://github.com/mathjax/MathJax/issues/2428
4664 // \left, \right delimiters: https://github.com/mathjax/MathJax/issues/2535
4665 //
4666 // Modified by Brian Dunn to adjust equation numbering and add subequations.
4667 //
4668 // LaTeX can use \seteqnumber{subequations?}{section}{number} before each equation.
4669 // subequations? is 0 usually, 1 if inside subequations.
4670 // section is a string printed as-is, or empty.
4671 // number is auto-incremented by MathJax between equations.
4672 //
4673 MathJax = {
4674   subequations: "0",
4675   section: "",
4676   loader: {
4677     load: ['[tex]/tagformat', '[tex]/textmacros'],
4678   },
4679   startup: {
4680     ready() {
4681       // These would be replaced by import commands if you wanted to make
4682       // a proper extension.
4683       const Configuration = MathJax._.input.tex.Configuration.Configuration;
4684       const CommandMap = MathJax._.input.tex.SymbolMap.CommandMap;
4685       const Macro = MathJax._.input.tex.Symbol.Macro;
4686       const TexError = MathJax._.input.tex.TexError.default;
4687       const ParseUtil = MathJax._.input.tex.ParseUtil.default;
4688       const expandable = MathJax._.util.Options.expandable;
4689
4690       // Insert the replacement string into the TeX string, and check
4691       // that there haven't been too many maxro substitutions (prevents
4692       // infinite loops).
4693       const useArgument = (parser, text) => {
4694      parser.string = ParseUtil.addArgs(parser, text, parser.string.slice(parser.i));
4695         parser.i = 0;
4696         if (++parser.macroCount > parser.configuration.options.maxMacros) {
4697           throw new TexError('MaxMacroSub1',
4698           'MathJax maximum macro substitution count exceeded; ' +
4699           'is there a recursive macro call?');
4700         }
4701       }
4702
4703       // Create the command map for:
4704       //   \ifstar, \ifnextchar, \ifblank, \ifstrequal, \gsub, \seteqnumber
4705       new CommandMap('Lwarp-macros', {
4706         ifstar: 'IfstarFunction',
```

```
4707            ifnextchar: 'IfnextcharFunction',
4708            ifblank: 'IfblankFunction',
4709            ifstrequal: 'IfstrequalFunction',
4710            gsubstitute: 'GsubstituteFunction',
4711            seteqnumber: 'SeteqnumberFunction'
4712         }, {
4713           //  This function implements an ifstar macro.
4714           IfstarFunction(parser, name) {
4715             const resultstar = parser.GetArgument(name);
4716             const resultnostar = parser.GetArgument(name);
4717            const star = parser.GetStar();                    // true if there is a *
4718             useArgument(parser, star ? resultstar : resultnostar);
4719           },
4720
4721           //  This function implements an ifnextchar macro.
4722           IfnextcharFunction(parser, name) {
4723             let whichchar = parser.GetArgument(name);
4724             if (whichchar.match(/^(?:0x[0-9A-F]+|[0-9]+)$/i)) {
4725               // $ syntax highlighting
4726               whichchar = String.fromCodePoint(parseInt(whichchar));
4727             }
4728             const resultnextchar = parser.GetArgument(name);
4729             const resultnotnextchar = parser.GetArgument(name);
4730             const gotchar = (parser.GetNext() === whichchar);
4731             useArgument(parser, gotchar ? resultnextchar : resultnotnextchar);
4732           },
4733
4734           // This function implements an ifblank macro.
4735           IfblankFunction(parser, name) {
4736             const blankarg = parser.GetArgument(name);
4737             const resultblank = parser.GetArgument(name);
4738             const resultnotblank = parser.GetArgument(name);
4739             const isblank = (blankarg.trim() == "");
4740             useArgument(parser, isblank ? resultblank : resultnotblank);
4741           },
4742
4743           // This function implements an ifstrequal macro.
4744           IfstrequalFunction(parser, name) {
4745             const strequalfirst = parser.GetArgument(name);
4746             const strequalsecond = parser.GetArgument(name);
4747             const resultequal = parser.GetArgument(name);
4748             const resultnotequal = parser.GetArgument(name);
4749             const isequal = (strequalfirst == strequalsecond);
4750             useArgument(parser, isequal ? resultequal : resultnotequal);
4751           },
4752
4753           // This function implements a gsub macro.
4754           GsubstituteFunction(parser, name) {
4755             const gsubfirst = parser.GetArgument(name);
4756             const gsubsecond = parser.GetArgument(name);
4757             const gsubthird = parser.GetArgument(name);
4758             let gsubresult=gsubfirst.replace(gsubsecond, gsubthird);
4759             useArgument(parser, gsubresult);
4760           },
4761
4762           //  This function modifies the equation numbers.
4763           SeteqnumberFunction(parser, name) {
4764              //  Get the macro parameters
4765             const star = parser.GetStar();                    // true if there is a *
4766         const optBrackets = parser.GetBrackets(name);  // contents of optional brackets
```

```
4767        const newsubequations = parser.GetArgument(name); // the subequations argument
4768        const neweqsection = parser.GetArgument(name); // the eq section argument
4769        const neweqnumber = parser.GetArgument(name);  // the eq number argument
4770        MathJax.config.subequations=newsubequations ;  // a string with boolean meaning
4771        MathJax.config.section=neweqsection ;          // a string with numeric meaning
4772          parser.tags.counter = parser.tags.allCounter = neweqnumber ;
4773        }

4775    });

4777    //  Create the Lwarp-macros package
4778    Configuration.create('Lwarp-macros', {
4779      handler: {macro: ['Lwarp-macros']}
4780    });

4782    MathJax.startup.defaultReady();

4784    // For forward references:
4785    MathJax.startup.input[0].preFilters.add(({math}) => {
4786      if (math.inputData.recompile){
4787      MathJax.config.subequations = math.inputData.recompile.subequations;
4788          MathJax.config.section = math.inputData.recompile.section;
4789      }
4790    });
4791    MathJax.startup.input[0].postFilters.add(({math}) => {
4792      if (math.inputData.recompile){
4793      math.inputData.recompile.subequations = MathJax.config.subequations;
4794          math.inputData.recompile.section = MathJax.config.section;
4795      }
4796    });

4798    // For \left, \right with unicode-math:
4799    const {DelimiterMap} = MathJax._.input.tex.SymbolMap;
4800    const {Symbol} = MathJax._.input.tex.Symbol;
4801    const {MapHandler} = MathJax._.input.tex.MapHandler;
4802    const delimiter = MapHandler.getMap('delimiter');
4803    delimiter.add('\\lBrack', new Symbol('\\lBrack', '\u27E6'));
4804    delimiter.add('\\rBrack', new Symbol('\\rBrack', '\u27E7'));
4805    delimiter.add('\\lAngle', new Symbol('\\lAngle', '\u27EA'));
4806    delimiter.add('\\rAngle', new Symbol('\\rAngle', '\u27EB'));
4807    delimiter.add('\\lbrbrak', new Symbol('\\lbrbrak', '\u2772'));
4808    delimiter.add('\\rbrbrak', new Symbol('\\rbrbrak', '\u2773'));
4809    delimiter.add('\\lbag', new Symbol('\\lbag', '\u27C5'));
4810    delimiter.add('\\rbag', new Symbol('\\rbag', '\u27C6'));
4811  delimiter.add('\\llparenthesis', new Symbol('\\llparenthesis', '\u2987'));
4812  delimiter.add('\\rrparenthesis', new Symbol('\\rrparenthesis', '\u2988'));
4813    delimiter.add('\\llangle', new Symbol('\\llangle', '\u2989'));
4814    delimiter.add('\\rrangle', new Symbol('\\rrangle', '\u298A'));
4815    delimiter.add('\\Lbrbrak', new Symbol('\\Lbrbrak', '\u27EC'));
4816    delimiter.add('\\Rbrbrak', new Symbol('\\Rbrbrak', '\u27ED'));
4817    delimiter.add('\\lBrace', new Symbol('\\lBrace', '\u2983'));
4818    delimiter.add('\\rBrace', new Symbol('\\rBrace', '\u2984'));
4819    delimiter.add('\\lParen', new Symbol('\\lParen', '\u2985'));
4820    delimiter.add('\\rParen', new Symbol('\\rParen', '\u2986'));
4821    delimiter.add('\\lbrackubar', new Symbol('\\lbrackubar', '\u298B'));
4822    delimiter.add('\\rbrackubar', new Symbol('\\rbrackubar', '\u298C'));
4823  delimiter.add('\\lbrackultick', new Symbol('\\lbrackultick', '\u298D'));
4824  delimiter.add('\\rbracklrtick', new Symbol('\\rbracklrtick', '\u298E'));
4825  delimiter.add('\\lbracklltick', new Symbol('\\lbracklltick', '\u298F'));
4826    delimiter.add('\\rbrackurtick', new Symbol('\\rbrackurtick', '\u2990'));
```

```
4827          delimiter.add('\\langledot', new Symbol('\\langledot', '\u2991'));
4828          delimiter.add('\\rangledot', new Symbol('\\rangledot', '\u2992'));
4829         delimiter.add('\\lparenless', new Symbol('\\lparenless', '\u2993'));
4830          delimiter.add('\\rparengtr', new Symbol('\\rparengtr', '\u2994'));
4831          delimiter.add('\\Lparengtr', new Symbol('\\Lparengtr', '\u2995'));
4832         delimiter.add('\\Rparenless', new Symbol('\\Rparenless', '\u2996'));
4833          delimiter.add('\\lblkbrbrak', new Symbol('\\lblkbrbrak', '\u2997'));
4834          delimiter.add('\\rblkbrbrak', new Symbol('\\rblkbrbrak', '\u2998'));
4835          delimiter.add('\\lvzigzag', new Symbol('\\lvzigzag', '\u29D8'));
4836          delimiter.add('\\rvzigzag', new Symbol('\\rvzigzag', '\u29D9'));
4837          delimiter.add('\\Lvzigzag', new Symbol('\\Lvzigzag', '\u29DA'));
4838          delimiter.add('\\Rvzigzag', new Symbol('\\Rvzigzag', '\u29DB'));
4839       delimiter.add('\\lcurvyangle', new Symbol('\\lcurvyangle', '\u29FC'));
4840       delimiter.add('\\rcurvyangle', new Symbol('\\rcurvyangle', '\u29FD'));
4841          delimiter.add('\\Vvert', new Symbol('\\Vvert', '\u2980'));
4842     }   // ready
4843   },    // startup
4844
4845   tex: {
4846     packages: {'[+]': ['tagformat', 'Lwarp-macros', 'textmacros']},
4847     tags: "ams",
4848         tagformat: {
4849            number: function (n) {
4850                 if(MathJax.config.subequations==0)
4851                    return(MathJax.config.section + n);
4852                 else
4853                   return(MathJax.config.section + String.fromCharCode(96+n));
4854            },
4855         },
4856   }
4857 }
4858 </script>
4859
4860 <script
4861     id="MathJax-script"
4862     src="https://cdn.jsdelivr.net/npm/mathjax@3/es5/tex-svg.js"
4863 ></script>
4864 \end{filecontents*}
4865 % \end{Verbatim}% for syntax highlighting
4866 \end{LWRwriteconf}
```

42.12   `lwarpmk.lua` — `lwarpmk` option

lwarpmk (*Opt*)   Creates a local copy of *lwarpmk*.

lwarpmk (*Prog*)   Command-line utility to process lwarp files and images.

parallel processing   lateximages and SVG math images are generated using multiple processes in parallel. For UNIX and LINUX, every 32 images the wait command is issued to wait for the previous batch of images to finish processing before starting a new batch. For WINDOWS, every 32 images one task is dispatched with

     START /B /WAIT /BELOWNORMAL

which causes the operating system to wait until this lesser-priority tasks finishes, hopefully also waiting for the normal priority tasks which were already in progress to also complete. Afterwards, the next batch of images is started.

The following is only generated if the lwarpmk option was given to lwarp.

```
4867 \begin{LWRcreatelwarpmk}

4868 \begin{filecontents*}[overwrite]{lwarpmk.lua}
4869 #!/usr/bin/env texlua
4870
4871 -- Copyright 2016-2025 Brian Dunn
4872
4873
4874 printversion = "v0.918"
4875 requiredconfversion = "2" -- also at *lwarpmk.conf
4876
4877 function printhelp ()
4878 print ("lwarpmk: Use lwarpmk -h or lwarpmk --help for help.") ;
4879 end
4880
4881
4882 function printusage ()
4883 --
4884 -- Print the usage of the lwarpmk command:
4885 --
4886 print ( [[
4887
4888 lwarpmk print [-p project]: Compile the print version if necessary.
4889 lwarpmk print1 [-p project]: Forced single compile of the print version.
4890 lwarpmk printindex [-p project]: Process print indexes.
4891 lwarpmk printglossary [-p project]: Process the glossary for the print version.
4892 lwarpmk html [-p project]: Compile the HTML version if necessary.
4893 lwarpmk html1 [-p project]: Forced single compile of the HTML version.
4894 lwarpmk htmlindex [-p project]: Process HTML indexes.
4895 lwarpmk htmlglossary [-p project]: Process the glossary for the html version.
4896 lwarpmk again [-p project]: Touch the source code to trigger recompiles.
4897 lwarpmk limages [-p project]: Process the "lateximages" created by lwarp.sty.
4898 lwarpmk pdftohtml [-p project]:
4899     For use with latexmk or a Makefile:
4900     Converts project_html.pdf to project_html.html and individual HTML files.
4901     Finishes the HTML conversion even if there was a compile error.
4902 lwarpmk pdftosvg <list of file names>: Converts each PDF file to SVG.
4903 lwarpmk epstopdf <list of file names>: Converts each EPS file to PDF.
4904 lwarpmk clean [-p project]: Remove *.aux, *.toc, *.lof/t,
4905     *.idx, *.ind, *.bbl, *.log, *_html_inc.*, .gl*,
4906     *_html.pdf, *_html.html, *_html.sidetoc
4907 lwarpmk cleanall [-p project]: Remove auxiliary files, project.pdf, *.html
4908 lwarpmk cleanlimages: Removes all images from the "lateximages" directory.
4909 lwarpmk -v: Print the version number.
4910 lwarpmk -h: Print this help message.
4911 lwarpmk --help: Print this help message.
4912
4913 ]] )
4914 -- printconf ()
4915 end
4916
4917
4918 function splitfilename ( pathandfilename )
4919 --
4920 -- Separates out the path and extension from a filename.
4921 -- Returns path, filename with extension, and extension.
4922 -- Ex: thispath, thisfilename, thisextension = splitfilename ("path/to/filename.ext")
4923 --
```

```
4924 -- https://www.fhug.org.uk/wiki/wiki/doku.php?id=plugins:code_snippets:
4925 --      split_filename_in_to_path_filename_and_extension
4926 --
4927     if lfs.attributes(pathandfilename,"mode") == "directory" then
4928       local strPath = pathandfilename:gsub("[\\/]$","") -- $ (syntax highlighting)
4929         return strPath.."\\","",""
4930     end
4931     pathandfilename = pathandfilename.."."
4932     return pathandfilename:match("^(.-)([^\\/]-)%.([^\\/%.]-)%.?$")
4933 end
4934
4935
4936 function splitfile (destfile,sourcefile)
4937 --
4938 -- Split one large sourcefile into a number of files,
4939 -- starting with destfile.
4940 -- The file is split at each occurance of <!--|Start file|newfilename|*
4941 -- If lwarp is in use, sets usinglwarp.
4942 --
4943 usinglwarp = false ;
4944 print ("lwarpmk: Splitting " .. sourcefile .. " into " .. destfile) ;
4945 local sfile = io.open(sourcefile)
4946 io.output(destfile)
4947 for line in sfile:lines() do
4948 i,j,copen,cstart,newfilename = string.find (line,"(.*)|(.*)|(.*)|") ;
4949 if ( (i~= nil) and (copen == "<!--") and (cstart == "Start file")) then
4950     -- split the file
4951     io.output(newfilename) ;
4952 else
4953 if ( (i~= nil) and (copen == "<!--") and (cstart == "Using lwarp")) then
4954     -- verified the use of \usepackage{lwarp}
4955     usinglwarp = true ;
4956 else
4957     -- not a splitpoint
4958     io.write (line .. "\n") ;
4959 end end
4960 end -- do
4961 io.close(sfile)
4962 if ( usinglwarp == false ) then
4963     print ("lwarpmk: ===")
4964     print ("lwarpmk: \\usepackage{lwarp} was not detected.")
4965     print ("lwarpmk: The HTML output will not be correct.")
4966     print ("lwarpmk: Ensured that \\usepackage{lwarp} is enabled,")
4967     print ("lwarpmk: then lwarpmk print and lwarpmk html again.")
4968     print ("lwarpmk: ===")
4969 end
4970 end -- function
4971
4972
4973 function cvalueerror ( line, linenum , cvalue )
4974 --
4975 -- Incorrect value, so print an error and exit.
4976 --
4977     print ("lwarpmk: ===")
4978     print ("lwarpmk: " .. linenum .. " : " .. line ) ;
4979     print (
4980         "lwarpmk: incorrect variable value \"" .. cvalue ..
4981         "\" in lwarpmk.conf.\n"
4982     ) ;
4983     print ("lwarpmk: ===")
```

```
4984 --    printconf () ;
4985    os.exit(1) ;
4986 end
4987
4988
4989 function printhowtorecompile ()
4990 -- Tells the user how to recompile to regenerate the configuration files.
4991    print ("lwarpmk: The configuration files lwarpmk.conf and "..sourcename..".lwarpmkconf" )
4992    print ("lwarpmk:   must be updated.  To do so, recompile" )
4993    print ("lwarpmk:   " , sourcename..".tex" )
4994    if ( printlatexcmd == "" ) then
4995       print ("lwarpmk:   using xe/lua/pdflatex," )
4996    else
4997       print ("lwarpmk:   using the command:")
4998       print ("lwarpmk:   " , printlatexcmd )
4999    end
5000    print ("lwarpmk:   then use lwarpmk again.")
5001 end -- printhowtorecompile
5002
5003
5004 function ignoreconf ()
5005 -- Global argument index
5006 argindex = 2
5007 end
5008
5009 function loadconf ()
5010 --
5011 -- Load settings from the project's "lwarpmk.conf" file:
5012 --
5013 -- Default configuration filename:
5014 local conffile = "lwarpmk.conf"
5015 local confroot = "lwarpmk"
5016 -- Global argument index
5017 argindex = 2
5018 -- Optional configuration filename:
5019 if ( arg[argindex] == "-p" ) then
5020    argindex = argindex + 1
5021    confroot = arg[argindex]
5022    conffile = confroot..".lwarpmkconf"
5023    argindex = argindex + 1
5024 end
5025 -- Additional defaults:
5026 confversion = "0"
5027 opsystem = "Unix"
5028 imagesdirectory = "lateximages"
5029 imagesname = "image-"
5030 latexmk = "false"
5031 printlatexcmd = ""
5032 HTMLlatexcmd = ""
5033 printindexcmd = ""
5034 HTMLindexcmd = ""
5035 latexmkindexcmd = ""
5036 -- to be removed:
5037 -- indexprog = "makeindex"
5038 -- makeindexstyle = "lwarp.ist"
5039 -- xindylanguage = "english"
5040 -- xindycodepage = "utf8"
5041 -- xindystyle = "lwarp.xdy"
5042 -- pdftotextenc = "UTF-8"
5043 glossarycmd = "makeglossaries"
```

```
5044 -- Verify the file exists:
5045 if (lfs.attributes(conffile,"mode")==nil) then
5046     -- file not exists
5047     print ("lwarpmk: ===")
5048     print ("lwarpmk: File \"" .. conffile .."\" does not exist.")
5049     print ("lwarpmk: Move to the project's source directory,")
5050     print ("lwarpmk: recompile using pdflatex, xelatex, or lualatex,")
5051     print ("lwarpmk: then try using lwarpmk again.")
5052     if ( arg[argindex] ~= nil ) then
5053         print (
5054             "lwarpmk: (\"" .. confroot ..
5055             "\" does not appear to be a project name.)"
5056         )
5057     end
5058     print ("lwarpmk: ===")
5059     printhelp () ;
5060     os.exit(1) -- exit the entire lwarpmk script
5061 else -- file exists
5062 -- Read the file:
5063 print ("lwarpmk: Reading " .. conffile ..".")
5064 local cfile = io.open(conffile)
5065 -- Scan each line, parsing each line as: name = [[string]]
5066 local linenum = 0
5067 for line in cfile:lines() do -- scan lines
5068 linenum = linenum + 1
5069 i,j,cvarname,cvalue = string.find (line,"([%w-_]*)%s*=%s*%[%[([^%]]*)%]%]") ;
5070 -- Error if incorrect enclosing characters:
5071 if ( i == nil ) then
5072     print ("lwarpmk: ===")
5073     print ("lwarpmk: " ..  linenum .. " : " .. line ) ;
5074     print ("lwarpmk: Incorrect entry in " .. conffile ..".\n" ) ;
5075     print ("lwarpmk: ===")
5076 --    printconf () ;
5077     os.exit(1) ;
5078 end -- nil
5079 if ( cvarname == "confversion" ) then
5080     confversion = cvalue
5081 elseif ( cvarname == "opsystem" ) then
5082     -- Verify choice of opsystem:
5083     if ( (cvalue == "Unix") or (cvalue == "Windows") ) then
5084         opsystem = cvalue
5085     else
5086         cvalueerror ( line, linenum , cvalue )
5087     end
5088 elseif ( cvarname == "sourcename" ) then sourcename = cvalue
5089 elseif ( cvarname == "homehtmlfilename" ) then homehtmlfilename = cvalue
5090 elseif ( cvarname == "htmlfilename" ) then htmlfilename = cvalue
5091 elseif ( cvarname == "imagesdirectory" ) then imagesdirectory = cvalue
5092 elseif ( cvarname == "imagesname" ) then imagesname = cvalue
5093 elseif ( cvarname == "latexmk" ) then latexmk = cvalue
5094 elseif ( cvarname == "printlatexcmd" ) then printlatexcmd = cvalue
5095 elseif ( cvarname == "HTMLlatexcmd" ) then HTMLlatexcmd = cvalue
5096 elseif ( cvarname == "printindexcmd" ) then printindexcmd = cvalue
5097 elseif ( cvarname == "HTMLindexcmd" ) then HTMLindexcmd = cvalue
5098 elseif ( cvarname == "latexmkindexcmd" ) then latexmkindexcmd = cvalue
5099 elseif ( cvarname == "glossarycmd" ) then glossarycmd = cvalue
5100 elseif ( cvarname == "pdftotextenc" ) then pdftotextenc = cvalue
5101 else
5102     print ("lwarpmk: ===")
5103     print ("lwarpmk: " .. linenum .. " : " .. line ) ;
```

```
5104     print (
5105         "lwarpmk: Incorrect variable name \"" .. cvarname .. "\" in " ..
5106         conffile ..".\n"
5107     ) ;
5108     print ("lwarpmk: ===")
5109 --    printconf () ;
5110 os.exit(1) ;
5111 end -- cvarname
5112 end -- do scan lines
5113 io.close(cfile)
5114 end -- file exists
5115 -- Error if sourcename is "lwarp".
5116 -- This could happen if a local copy of lwarp has recently been recompiled.
5117 if sourcename=="lwarp" then
5118     print ("lwarpmk: ===")
5119   print ("lwarpmk: lwarp.sty has recently been recompiled in this directory,")
5120   print ("lwarpmk: and \"lwarpmk.conf\" is no longer set for your own project.")
5121     print ("lwarpmk:    (Perhaps you are not in your project's directory?)")
5122     print ("lwarpmk: In your project directory, recompile your project")
5123     print ("lwarpmk: using pdf/lua/xelatex <projectname>.")
5124   print ("lwarpmk: After a recompile, \"lwarpmk.conf\" will be set for your project,")
5125     print ("lwarpmk: and you may again use lwarpmk.")
5126     print ("lwarpmk: ===")
5127     os.exit(1)
5128 end -- sourcename of "lwarp"
5129 -- Select some operating-system commands:
5130 if opsystem=="Unix" then  -- For Unix / Linux / Mac OS:
5131     rmname = "rm"
5132     mvname = "mv"
5133     cpname = "cp"
5134     touchnamepre = "touch"
5135     touchnamepost = ""
5136     newtouchname = "touch"
5137     dirslash = "/"
5138     opquote= "\'"
5139     cmdgroupopenname = " ( "
5140     cmdgroupclosename = " ) "
5141     seqname = " && "
5142     bgname = " &"
5143 elseif opsystem=="Windows" then -- For Windows
5144     rmname = "DEL"
5145     mvname = "MOVE"
5146     cpname = "COPY"
5147     touchnamepre = "COPY /b"
5148     touchnamepost = "+,,"
5149     newtouchname = "echo empty >"
5150     dirslash = "\\"
5151     opquote= "\""
5152     cmdgroupopenname = ""
5153     cmdgroupclosename = ""
5154     seqname = " & "
5155     bgname = ""
5156 else
5157     print ("lwarpmk: ===")
5158     print ("lwarpmk: Select Unix or Windows for opsystem." )
5159     print ("lwarpmk: ===")
5160     os.exit(1)
5161 end --- for Windows
5162 -- Warning if the operating system does not appear to be correct,
5163 -- in case files were transferred to another system.
```

```
5164 if ( (package.config:sub(1,1)) ~= dirslash ) then
5165     print ("lwarpmk: ===")
5166   print ("lwarpmk: It appears that lwarpmk.conf is for a different operating system." )
5167     printhowtorecompile ()
5168     print ("lwarpmk: ===")
5169     os.exit(1)
5170 end
5171 -- Error if the configuration file's version is not current:
5172 if ( confversion ~= requiredconfversion ) then
5173     print ("lwarpmk: ===")
5174     printhowtorecompile ()
5175     print ("lwarpmk: ===")
5176     os.exit(1)
5177 end
5178 end -- loadconf
5179
5180
5181 function executecheckerror ( executecommands , errormessage )
5182 --
5183 -- Execute an operating system call,
5184 -- and maybe exit with an error message.
5185 --
5186 local err
5187 err = os.execute ( executecommands )
5188 if ( err ~= 0 ) then
5189     print ("lwarpmk: ===")
5190     print ("lwarpmk: " .. errormessage )
5191     print ("lwarpmk: ===")
5192     os.exit(1)
5193 end
5194 end -- executecheckerror
5195
5196
5197 function refreshdate ()
5198 os.execute(touchnamepre .. " " .. sourcename .. ".tex " .. touchnamepost)
5199 end
5200
5201
5202
5203 function reruntoget (filesource)
5204 --
5205 -- Scan the LaTeX log file for the phrase "Rerun to get",
5206 -- indicating that the file should be compiled again.
5207 -- Return true if found.
5208 --
5209 local fsource = io.open(filesource)
5210 for line in fsource:lines() do
5211 if ( string.find(line,"Rerun to get") ~= nil ) then
5212     io.close(fsource)
5213     return true
5214 end -- if
5215 end -- do
5216 io.close(fsource)
5217 return false
5218 end
5219
5220
5221
5222 function onetime (latexcmd, fsuffix)
5223 --
```

```
5224 -- Compile one time, return true if should compile again.
5225 -- fsuffix is "" for print, "_html" for HTML output.
5226 --
5227 print("lwarpmk: Compiling with: " .. latexcmd)
5228 executecheckerror (
5229     latexcmd ,
5230     "Compile error."
5231 )
5232 return (reruntoget(sourcename .. fsuffix .. ".log") ) ;
5233 end
5234
5235
5236 function manytimes (latexcmd, fsuffix)
5237 --
5238 -- Compile up to five times.
5239 -- fsuffix is "" for print, "_html" for HTML output
5240 --
5241 if onetime(latexcmd, fsuffix) == true then
5242 if onetime(latexcmd, fsuffix) == true then
5243 if onetime(latexcmd, fsuffix) == true then
5244 if onetime(latexcmd, fsuffix) == true then
5245 if onetime(latexcmd, fsuffix) == true then
5246 end end end end end
5247 end
5248
5249
5250 function verifyfileexists (filename)
5251 --
5252 -- Exit if the given file does not exist.
5253 --
5254 if (lfs.attributes ( filename , "modification" ) == nil ) then
5255     print ("lwarpmk: ===")
5256     print ("lwarpmk: " .. filename .. " not found." ) ;
5257     print ("lwarpmk: ===")
5258     os.exit (1) ;
5259 end
5260 end
5261
5262
5263
5264 function pdftohtml ()
5265 --
5266 -- Convert <project>_html.pdf into HTML files:
5267 --
5268 -- Convert to text:
5269 print ("lwarpmk: Converting " .. sourcename
5270     .."_html.pdf to " .. sourcename .. "_html.html")
5271 err = os.execute("pdftotext -enc " .. pdftotextenc .. " -nopgbrk -layout "
5272     .. sourcename .. "_html.pdf " .. sourcename .. "_html.html")
5273 if ( err ~= 0 ) then
5274     print ("lwarpmk: ===")
5275     print ("lwarpmk: Ensure that the Poppler utilities are installed." )
5276   print ("lwarpmk: See the Lwarp manual: `Installing additional utilities'." )
5277     print ("lwarpmk: ===")
5278     os.exit(1)
5279 end
5280 -- Split the result into individual HTML files:
5281 splitfile (homehtmlfilename .. ".html" , sourcename .. "_html.html")
5282 end
5283
```

```
5284
5285 function removeaux ()
5286 --
5287 -- Remove auxiliary files:
5288 -- All .aux files are removed since there may be many bbl*.aux files.
5289 -- Also removes sourcename_html.pdf, sourcename_html.html,
5290 -- and sourcename_html.sidetoc, plus comment_*.cut.
5291 --
5292 os.execute ( rmname .. " *.aux " ..
5293     sourcename ..".toc " .. sourcename .. "_html.toc " ..
5294     sourcename ..".lof " .. sourcename .. "_html.lof " ..
5295     sourcename ..".lot " .. sourcename .. "_html.lot " ..
5296     sourcename ..".bbl " .. sourcename .. "_html.bbl " ..
5297     " *.idx " ..
5298     " *.ind " ..
5299     sourcename ..".ps " .. sourcename .."_html.ps " ..
5300     sourcename ..".log " .. sourcename .. "_html.log " ..
5301     sourcename ..".gl* " .. sourcename .. "_html.gl* " ..
5302     sourcename .. "_html.pdf " ..
5303     sourcename .. "_html.html " ..
5304     sourcename .. "_html.sidetoc " ..
5305     " *_html_inc.* " ..
5306     " comment_*.cut"
5307     )
5308 end
5309
5310 function checkhtmlpdfexists ()
5311 --
5312 -- Error if the HTML document does not exist.
5313 -- The lateximages are drawn from the HTML PDF version of the document,
5314 -- so "lwarpmk html" must be done before "lwarpmk limages".
5315 --
5316 local htmlpdffile = io.open(sourcename .. "_html.pdf", "r")
5317 if ( htmlpdffile == nil ) then
5318     print ("")
5319     print ("lwarpmk: ===")
5320     print ("lwarpmk: The HTML version of the document does not exist.")
5321     print ("lwarpmk: Enter \"lwarpmk html\" to compile the HTML version.")
5322     print ("lwarpmk: ===")
5323     os.exit(1)
5324 end
5325 io.close (htmlpdffile)
5326 end -- checkhtmlpdfexists
5327
5328
5329 function warnlimages ()
5330 --
5331 -- Warning of a missing <sourcename>-images.txt file:
5332     print ("lwarpmk: ===")
5333     print ("lwarpmk: \"" .. sourcename .. "-images.txt\" does not exist.")
5334    print ("lwarpmk: Your project does not use SVG math or other lateximages,")
5335     print ("lwarpmk: or the file has been deleted somehow.")
5336     print ("lwarpmk: Use \"lwarpmk html1\" to recompile your project")
5337     print ("lwarpmk: and recreate \"" .. sourcename .. "-images.txt\".")
5338    print ("lwarpmk: If your project does not use SVG math or other lateximages,")
5339    print ("lwarpmk: then \"" .. sourcename .. "-images.txt\" will never exist, and")
5340     print ("lwarpmk: \"lwarpmk limages\" will not be necessary.")
5341     print ("lwarpmk: ===")
5342 end -- warnlimages
5343
```

```
5344
5345 function warnlimagesrecompile ()
5346 -- Warning if must recompile before creating limages:
5347     print ("")
5348     print ("lwarpmk: ===")
5349     print ("lwarpmk: Cross-references are not yet correct.")
5350   print ("lwarpmk: The document must be recompiled before creating the lateximages.")
5351   print ("lwarpmk: Enter \"lwarpmk html1\" again, then try \"lwarpmk limages\" again.")
5352     print ("lwarpmk: ===")
5353 end --warnlimagesrecompile
5354
5355
5356 function checklimages ()
5357 --
5358 -- Check <sourcename>.txt to see if need to recompile first.
5359 -- If any entry has a page number of zero, then there were incorrect images.
5360 --
5361 print ("lwarpmk: Checking for a valid " .. sourcename .. "-images.txt file.")
5362 local limagesfile = io.open(sourcename .. "-images.txt", "r")
5363 if ( limagesfile == nil ) then
5364     warnlimages ()
5365     os.exit(1)
5366 end
5367 -- Track warning to recompile if find a page 0
5368 local pagezerowarning = false
5369 -- Scan <sourcename>.txt
5370 for line in limagesfile:lines() do
5371     -- lwimgpage is the page number in the PDF which has the image
5372     -- lwimghash is true if this filename is a hash
5373     -- lwimgname is the lateximage filename root to assign for the image
5374     i,j,lwimgpage,lwimghash,lwimgname = string.find (line,"|(.*)|(.*)|(.*)|")
5375     -- For each entry:
5376     if ( (i~=nil) ) then
5377         -- If the page number is 0, image references are incorrect
5378         --  and must recompile the soure document:
5379         if ( lwimgpage == "0" ) then
5380             pagezerowarning = true
5381         end
5382     end -- if i~=nil
5383 end -- do
5384 -- The last line should be |end|end|end|.
5385 -- If not, the compile must have aborted, and the images are incomplete.
5386 if ( lwimgpage ~= "end" ) then
5387     warnlimagesrecompile()
5388     os.exit(1) ;
5389 end
5390 if ( pagezerowarning ) then
5391     warnlimagesrecompile()
5392     os.exit(1) ;
5393 end -- pagezerowarning
5394 end -- checklimages
5395
5396
5397 function createuniximage ( lwimgfullname )
5398 --
5399 -- Create one lateximage for Unix / Linux / Mac OS.
5400 --
5401 executecheckerror (
5402     cmdgroupopenname ..
5403     "pdfseparate -f " .. lwimgpage .. " -l " .. lwimgpage .. " " ..
```

```
5404         sourcename .."_html.pdf " ..
5405         imagesdirectory .. dirslash .."lateximagetemp-%d" .. ".pdf" ..
5406         seqname ..
5407    -- Crop the image:
5408    "pdfcrop --hires --margins \"0 1 0 0\" " .. imagesdirectory .. dirslash .. "lateximagetemp-" ..
5409         lwimgpage .. ".pdf " ..
5410         imagesdirectory .. dirslash .. lwimgname .. ".pdf" ..
5411         seqname ..
5412    -- Convert the image to svg:
5413    "pdftocairo -svg -noshrink " .. imagesdirectory .. dirslash .. lwimgname .. ".pdf " ..
5414         imagesdirectory .. dirslash .. lwimgname .."svg" ..
5415         seqname ..
5416    -- Remove the temporary files:
5417    rmname .. " " .. imagesdirectory .. dirslash .. lwimgname .. ".pdf" .. seqname ..
5418    rmname .. " " .. imagesdirectory .. dirslash .. "lateximagetemp-" .. lwimgpage .. ".pdf" ..
5419    cmdgroupclosename .. " >/dev/null " .. bgname
5420    ,
5421    "File error trying to convert " .. lwimgfullname
5422 )
5423 -- Every 32 images, wait for completion at below normal priority,
5424 --  allowing other image tasks to catch up.
5425 numimageprocesses = numimageprocesses + 1
5426 if ( numimageprocesses > 32 ) then
5427    numimageprocesses = 0
5428    print ( "lwarpmk: waiting" )
5429    executecheckerror ( "wait" , "File error trying to wait.")
5430 end
5431 end -- createuniximage
5432
5433
5434 function createwindowsimage ( lwimgfullname )
5435 --
5436 -- Create one lateximage for Windows.
5437 --
5438 -- Every 32 images, wait for completion at below normal priority,
5439 --  allowing other image tasks to catch up.
5440 numimageprocesses = numimageprocesses + 1
5441 if ( numimageprocesses > 32 ) then
5442    numimageprocesses = 0
5443    thiswaitcommand = "/WAIT /BELOWNORMAL"
5444    print ( "lwarpmk: waiting" )
5445 else
5446    thiswaitcommand = ""
5447 end
5448 -- Execute the image generation command
5449 executecheckerror (
5450    "start /B " .. thiswaitcommand .. " \"\" lwarp_one_limage " ..
5451    lwimgpage .. " " ..
5452    lwimghash .. " " ..
5453    lwimgname .. " " ..
5454    sourcename .. " <nul >nul"
5455    ,
5456    "File error trying to create image."
5457 )
5458 end -- createwindowsimage
5459
5460
5461 function createonelateximage ( line )
5462 --
5463 -- Given the next line of <sourcename>.txt, convert a single image.
```

```
5464 --
5465 -- lwimgpage is the page number in the PDF which has the image
5466 -- lwimghash is true if this filename is a hash
5467 -- lwimgname is the lateximage filename root to assign for the image
5468 i,j,lwimgpage,lwimghash,lwimgname = string.find (line,"|(.*)|(.*)|(.*)|")
5469 -- For each entry:
5470 if ( (i~=nil) ) then
5471     -- Skip if the page number is 0:
5472     if ( lwimgpage == "0" ) then
5473         pagezerowarning = true
5474     -- Skip if the page number is "end":
5475     else if ( lwimgpage == "end" ) then
5476     else
5477         -- Skip is this image is hashed and already exists:
5478       local lwimgfullname = imagesdirectory .. dirslash .. lwimgname .. ".svg"
5479         if (
5480             (lwimghash ~= "true") or
5481             (lfs.attributes(lwimgfullname,"mode")==nil) -- file not exists
5482         )
5483         then -- not hashed or not exists:
5484             -- Print the name of the file being generated:
5485             print ( "lwarpmk: " .. lwimgname )
5486         -- Touch/create the dest so that only once instance tries to build it:
5487             executecheckerror (
5488                 newtouchname .. " " .. lwimgfullname ,
5489                 "File error trying to touch " .. lwimgfullname
5490             )
5491             -- Separate out the image into its own single-page pdf:
5492             if opsystem=="Unix" then
5493                 createuniximage (lwimgfullname)
5494             elseif opsystem=="Windows" then
5495                 createwindowsimage (lwimgfullname)
5496             end
5497         end -- not hashed or not exists
5498     end -- not page "end"
5499     end -- not page 0
5500 end -- not nil
5501 end -- createonelateximage
5502
5503
5504 function createlateximages ()
5505 --
5506 -- Create lateximages based on <sourcename>-images.txt:
5507 --
5508 -- See if the document must be recompiled first:
5509 checklimages ()
5510 -- See if the HTML version exists:
5511 checkhtmlpdfexists ()
5512 -- Attempt to create the lateximages:
5513 print ("lwarpmk: Creating lateximages.")
5514 local limagesfile = io.open(sourcename .. "-images.txt", "r")
5515 if ( limagesfile == nil ) then
5516     warnlimages ()
5517     os.exit(1)
5518 end
5519 -- Create the lateximages directory, ignore error if already exists
5520 err = os.execute("mkdir " .. imagesdirectory)
5521 -- For Windows, create lwarp_one_limage.cmd from lwarp_one_limage.txt:
5522 if opsystem=="Windows" then
5523     executecheckerror (
```

```
5524          cpname .. " lwarp_one_limage.txt lwarp_one_limage.cmd" ,
5525        "File error trying to copy lwarp_one_limage.txt to lwarp_one_limage.cmd"
5526    )
5527 end -- create lwarp_one_limage.cmd
5528 -- Track the number of parallel processes
5529 numimageprocesses = 0
5530 -- Track warning to recompile if find a page 0
5531 pagezerowarning = false
5532 -- Scan <sourcename>.txt
5533 for line in limagesfile:lines() do
5534    createonelateximage ( line )
5535 end -- do
5536 io.close(limagesfile)
5537 print ( "lwarpmk limages: ===")
5538 print ( "lwarpmk limages: Wait a moment for the images to complete" )
5539 print ( "lwarpmk limages:   before reloading the page." )
5540 print ( "lwarpmk limages: ===")
5541 print ( "lwarpmk limages: Done." )
5542 if ( pagezerowarning == true ) then
5543    print ( "lwarpmk limages: WARNING: Images will be incorrect." )
5544    print ( "lwarpmk limages:   Enter \"lwarpmk cleanlimages\", then" )
5545    print ( "lwarpmk limages:   recompile the document one more time, then" )
5546    print ( "lwarpmk limages:   repeat \"lwarpmk images\" again." )
5547 end -- pagezerowarning
5548 end -- function
5549
5550
5551 function convertepstopdf ()
5552 --
5553 -- Converts EPS files to PDF files.
5554 -- The filenames are arg[argindex] and up.
5555 -- arg[1] is the command "epstopdf".
5556 --
5557 ignoreconf ()
5558 for i = argindex , #arg do
5559    if (lfs.attributes(arg[i],"mode")==nil) then
5560        print ("lwarpmk: File \"" .. arg[i] .. "\" does not exist.")
5561    else
5562        print ("lwarpmk: Converting \"" .. arg[i] .. "\"")
5563        thispath, thisfilename, thisextension = splitfilename(arg[i])
5564        if ( thispath == nil ) then
5565            os.execute ( "epstopdf " .. arg[i] )
5566        else
5567            os.execute (
5568                "epstopdf " ..
5569                thispath .. thisfilename .. "." .. thisextension .. " " ..
5570                thispath .. thisfilename .. ".pdf"
5571            )
5572        end
5573    end -- if
5574 end -- do
5575 end --function
5576
5577
5578 function convertpdftosvg ()
5579 --
5580 -- Converts PDF files to SVG files.
5581 -- The filenames are arg[argindex] and up.
5582 -- arg[1] is the command "pdftosvg".
5583 --
```

```
5584 ignoreconf ()
5585 for i = argindex , #arg do
5586     if (lfs.attributes(arg[i],"mode")==nil) then
5587         print ("lwarpmk: File \"" .. arg[i] .. "\" does not exist.")
5588     else
5589         print ("lwarpmk: Converting \"" .. arg[i] .. "\"")
5590         thispath, thisfilename, thisextension = splitfilename(arg[i])
5591         if ( thispath == nil ) then
5592             os.execute ( "pdftocairo -svg " .. arg[i] )
5593         else
5594             os.execute (
5595                 "pdftocairo -svg " ..
5596                 thispath .. thisfilename .. "." .. thisextension .. " " ..
5597                 thispath .. thisfilename .. ".svg"
5598             )
5599         end
5600     end -- if
5601 end -- do
5602 end --function
5603
5604
5605 -- Force an update and conclude processing:
5606 function updateanddone ()
5607 print ("lwarpmk: Forcing an update of " .. sourcename ..".tex.")
5608 refreshdate ()
5609 print ("lwarpmk: " .. sourcename ..".tex is ready to be recompiled.")
5610 print ("lwarpmk: Done.")
5611 end -- function
5612
5613
5614 -- Start of the main code: --
5615
5616
5617 -- lwarpmk --version :
5618
5619 if (arg[1] == "--version") then
5620 print ( "lwarpmk: " .. printversion )
5621
5622 else -- not --version
5623
5624
5625 -- print intro:
5626
5627 print ("lwarpmk: " .. printversion .. " Automated make for the LaTeX Lwarp package.")
5628
5629
5630 -- lwarpmk print:
5631
5632 if arg[1] == "print" then
5633 loadconf ()
5634 if ( latexmk == "true" ) then
5635     print ("lwarpmk: Compiling with: " .. printlatexcmd)
5636     executecheckerror (
5637         printlatexcmd ,
5638         "Compile error."
5639     )
5640     print ("lwarpmk: Done.")
5641 else -- not latexmk
5642     verifyfileexists (sourcename .. ".tex") ;
5643     -- See if up to date:
```

```
5644     if (
5645        ( lfs.attributes ( sourcename .. ".pdf" , "modification" ) == nil ) or
5646        (
5647             lfs.attributes ( sourcename .. ".tex" , "modification" ) >
5648             lfs.attributes ( sourcename .. ".pdf" , "modification" )
5649        )
5650     ) then
5651        -- Recompile if not yet up to date:
5652        manytimes(printlatexcmd, "")
5653        print ("lwarpmk: Done.") ;
5654     else
5655        print ("lwarpmk: " .. sourcename .. ".pdf is up to date.") ;
5656     end
5657 end -- not latexmk
5658
5659
5660 -- lwarpmk print1:
5661
5662 elseif arg[1] == "print1" then
5663     loadconf ()
5664     verifyfileexists (sourcename .. ".tex") ;
5665     onetime(printlatexcmd, "")
5666     print ("lwarpmk: Done.") ;
5667
5668
5669 -- lwarpmk printindex:
5670 -- Compile the index then touch the source
5671 -- to trigger a recompile of the document:
5672
5673 elseif arg[1] == "printindex" then
5674 loadconf ()
5675 os.execute ( printindexcmd )
5676 print ("lwarpmk: -------")
5677 updateanddone ()
5678
5679
5680 -- lwarpmk printglossary:
5681 -- Compile the glossary then touch the source
5682 -- to trigger a recompile of the document:
5683
5684 elseif arg[1] == "printglossary" then
5685 loadconf ()
5686 print ("lwarpmk: Processing the glossary.")
5687
5688 os.execute(glossarycmd .. " " .. sourcename)
5689 updateanddone ()
5690
5691
5692 -- lwarpmk html:
5693
5694 elseif arg[1] == "html" then
5695 loadconf ()
5696 if ( latexmk == "true" ) then
5697     print ("lwarpmk: Compiling with: " .. HTMLlatexcmd)
5698     executecheckerror (
5699         HTMLlatexcmd ,
5700         "Compile error."
5701     )
5702     pdftohtml ()
5703     print ("lwarpmk: Done.")
```

```
5704 else -- not latexmk
5705     verifyfileexists ( sourcename .. ".tex" ) ;
5706     -- See if exists and is up to date:
5707     if (
5708       ( lfs.attributes ( homehtmlfilename .. ".html" , "modification" ) == nil ) or
5709         (
5710                 lfs.attributes ( sourcename .. ".tex" , "modification" ) >
5711                 lfs.attributes ( homehtmlfilename .. ".html" , "modification" )
5712         )
5713     ) then
5714         -- Recompile if not yet up to date:
5715         manytimes(HTMLlatexcmd, "_html")
5716         pdftohtml ()
5717         print ("lwarpmk: Done.")
5718     else
5719         print ("lwarpmk: " .. homehtmlfilename .. ".html is up to date.")
5720     end
5721 end -- not latexmk
5722
5723
5724 -- lwarpmk html1:
5725
5726 elseif arg[1] == "html1" then
5727     loadconf ()
5728     verifyfileexists ( sourcename .. ".tex" ) ;
5729     onetime(HTMLlatexcmd, "_html")
5730     pdftohtml ()
5731     print ("lwarpmk: Done.")
5732
5733
5734 -- lwarpmk pdftohtml:
5735 elseif arg[1] == "pdftohtml" then
5736     loadconf ()
5737     pdftohtml ()
5738
5739
5740 -- lwarpmk htmlindex:
5741 -- Compile the index then touch the source
5742 -- to trigger a recompile of the document:
5743
5744 elseif arg[1] == "htmlindex" then
5745 loadconf ()
5746 os.execute ( HTMLindexcmd )
5747 print ("lwarpmk: -------")
5748 updateanddone ()
5749
5750
5751 -- lwarpmk htmlglossary:
5752 -- Compile the glossary then touch the source
5753 -- to trigger a recompile of the document.
5754 -- The <sourcename>.xdy file is created by the glossaries package.
5755
5756 elseif arg[1] == "htmlglossary" then
5757 loadconf ()
5758 print ("lwarpmk: Processing the glossary.")
5759 os.execute(glossarycmd .. " " .. sourcename .. "_html")
5760 updateanddone ()
5761
5762
5763 -- lwarpmk limages:
```

```
5764 -- Scan the <sourcename>.txt file to create lateximages.
5765
5766 elseif arg[1] == "limages" then
5767 loadconf ()
5768 print ("lwarpmk: Processing images.")
5769 createlateximages ()
5770 print ("lwarpmk: Done.")
5771
5772
5773 -- lwarpmk again:
5774 -- Touch the source to trigger a recompile.
5775
5776 elseif arg[1] == "again" then
5777 loadconf ()
5778 updateanddone ()
5779
5780
5781 -- lwarpmk clean:
5782 -- Remove project.aux, .toc, .lof, .lot, .log, *.idx, *.ind, *_html_inc.*, .gl*
5783
5784 elseif arg[1] == "clean" then
5785 loadconf ()
5786 removeaux ()
5787 print ("lwarpmk: Done.")
5788
5789
5790 -- lwarpmk cleanall
5791 -- Remove project.aux, .toc, .lof, .lot, .log, *.idx, *.ind, *_html_inc.*, .gl*
5792 --     and also project.pdf, project.dvi, *.html
5793
5794 elseif arg[1] == "cleanall" then
5795 loadconf ()
5796 removeaux ()
5797 os.execute ( rmname .. " " ..
5798     sourcename .. ".pdf " .. sourcename .. "_html.pdf " ..
5799     sourcename .. ".dvi " .. sourcename .. "_html.dvi " ..
5800     "*.html"
5801     )
5802 print ("lwarpmk: Done.")
5803
5804
5805 -- lwarpmk cleanlimages
5806 -- Remove images from the imagesdirectory.
5807
5808 elseif arg[1] == "cleanlimages" then
5809 loadconf ()
5810 os.execute ( rmname .. " " .. imagesdirectory .. dirslash .. "*" )
5811 print ("lwarpmk: Done.")
5812
5813 -- lwarpmk epstopdf <list of file names>
5814 -- Convert EPS files to PDF using epstopdf
5815 elseif arg[1] == "epstopdf" then
5816 convertepstopdf ()
5817 print ("lwarpmk: Done.")
5818
5819
5820 -- lwarpmk pdftosvg <list of file names>
5821 -- Convert PDF files to SVG using pdftocairo
5822 elseif arg[1] == "pdftosvg" then
5823 convertpdftosvg ()
```

```
5824 print ("lwarpmk: Done.")
5825
5826
5827 -- lwarpmk with no argument :
5828
5829 elseif (arg[1] == nil) then
5830 printhelp ()
5831
5832
5833 -- lwarpmk -v:
5834
5835 elseif (arg[1] == "-v" ) then
5836 -- The version number has already been printed
5837 -- by the lwarpmk intro.
5838
5839 -- lwarpmk -h or lwarpmk --help :
5840
5841 elseif (arg[1] == "-h" ) or (arg[1] == "--help") then
5842 printusage ()
5843
5844
5845 -- Unknown command:
5846
5847 else
5848 printhelp ()
5849 print ("\nlwarpmk: ****** Unknown command \""..arg[1].."\". ******\n")
5850 end
5851
5852 end -- not --version
5853 \end{filecontents*}
5854 % \end{Verbatim}% for syntax highlighting

5855 \end{LWRcreatelwarpmk}
```

43 Stacks

5856 \begin{warpHTML}

⚠ Stacks are used to remember how to close sections and list items. Before a new section is started, previously nested sections and items must be closed out (un-nested) in proper order. Note that starting a new section may close several levels of previously nested items at the same time. For example, starting a new \section would close any currently open subsection, subsubsection, and paragraph. General environments are not nested on the stack since they have their own close mechanism. List environments are nested, and items inside those environments are nested one level deeper still. List environments may be nested inside other list environments, and list items are nested inside list environments as well. Thus, the stack may have items which are not necessarily in order, since a description may contain an enumerate, for example. Depths to be recorded in \LWR@closedepthone, etc.

43.1 Assigning depths

initial depths for empty stack entries:

```
5857 \newcommand*{\LWR@depthnone}{-5}
```

All sectioning depths are deeper than `LWR@depthfinished`:

```
5858 \newcommand*{\LWR@depthfinished}{-4}
5859 \newcommand*{\LWR@depthbook}{-2}
5860 \newcommand*{\LWR@depthpart}{-1}
5861 \newcommand*{\LWR@depthchapter}{0}
5862 \newcommand*{\LWR@depthsection}{1}
5863 \newcommand*{\LWR@depthsubsection}{2}
5864 \newcommand*{\LWR@depthsubsubsection}{3}
5865 \newcommand*{\LWR@depthparagraph}{4}
5866 \newcommand*{\LWR@depthsubparagraph}{5}
```

Used by `\itemize`, `\enumerate`, `\description`:

```
5867 \newcommand*{\LWR@depthlist}{6}
```

Used by `\item`:

```
5868 \newcommand*{\LWR@depthlistitem}{7}
5869 \let\LWR@depthdescitem\LWR@depthlistitem
```

43.2   Closing actions

A stack to record the action to take to close each nesting level: Add more levels of stack if necessary for a very deeply nested document, adding to `\pushclose` and `\popclose` as well.

```
5870 \newcommand*{\LWR@closeone}{}% top of the stack
5871 \newcommand*{\LWR@closetwo}{}
5872 \newcommand*{\LWR@closethree}{}
5873 \newcommand*{\LWR@closefour}{}
5874 \newcommand*{\LWR@closefive}{}
5875 \newcommand*{\LWR@closesix}{}
5876 \newcommand*{\LWR@closeseven}{}
5877 \newcommand*{\LWR@closeeight}{}
5878 \newcommand*{\LWR@closenine}{}
5879 \newcommand*{\LWR@closeten}{}
5880 \newcommand*{\LWR@closeeleven}{}
5881 \newcommand*{\LWR@closetwelve}{}
5882 \newcommand*{\LWR@closethirteen}{}
5883 \newcommand*{\LWR@closefourteen}{}
5884 \newcommand*{\LWR@closefifteen}{}
5885 \newcommand*{\LWR@closesixteen}{}
5886 \newcommand*{\LWR@closeseventeen}{}
5887 \newcommand*{\LWR@closeeighteen}{}
5888 \newcommand*{\LWR@closenineteen}{}
```

43.3   Closing depths

A stack to record the depth of each level:

⚠ Note that nested LaTeX structures may push depths which are non-sequential.

Ex:

```
\begin{itemize}
    \item{A}
    \begin{description}
        \item{B}
    \end{description}
\end{itemize}
```

```
5889 \newcommand*{\LWR@closedepthone}{\LWR@depthnone}% top of the stack
5890 \newcommand*{\LWR@closedepthtwo}{\LWR@depthnone}
5891 \newcommand*{\LWR@closedepththree}{\LWR@depthnone}
5892 \newcommand*{\LWR@closedepthfour}{\LWR@depthnone}
5893 \newcommand*{\LWR@closedepthfive}{\LWR@depthnone}
5894 \newcommand*{\LWR@closedepthsix}{\LWR@depthnone}
5895 \newcommand*{\LWR@closedepthseven}{\LWR@depthnone}
5896 \newcommand*{\LWR@closedeptheight}{\LWR@depthnone}
5897 \newcommand*{\LWR@closedepthnine}{\LWR@depthnone}
5898 \newcommand*{\LWR@closedepthten}{\LWR@depthnone}
5899 \newcommand*{\LWR@closedeptheleven}{\LWR@depthnone}
5900 \newcommand*{\LWR@closedepthtwelve}{\LWR@depthnone}
5901 \newcommand*{\LWR@closedepththirteen}{\LWR@depthnone}
5902 \newcommand*{\LWR@closedepthfourteen}{\LWR@depthnone}
5903 \newcommand*{\LWR@closedepthfifteen}{\LWR@depthnone}
5904 \newcommand*{\LWR@closedepthsixteen}{\LWR@depthnone}
5905 \newcommand*{\LWR@closedepthseventeen}{\LWR@depthnone}
5906 \newcommand*{\LWR@closedeptheighteen}{\LWR@depthnone}
5907 \newcommand*{\LWR@closedepthnineteen}{\LWR@depthnone}
```

43.4   Pushing and popping the stack

\LWR@pushclose {⟨*sectiontype*⟩}

Pushes one return action and its LaTeX depth onto the stacks.

```
5908 \NewDocumentCommand{\LWR@pushclose}{m}
5909 {%
5910 \global\let\LWR@closenineteen\LWR@closeeighteen%
5911 \global\let\LWR@closeeighteen\LWR@closeseventeen%
5912 \global\let\LWR@closeseventeen\LWR@closesixteen%
5913 \global\let\LWR@closesixteen\LWR@closefifteen%
5914 \global\let\LWR@closefifteen\LWR@closefourteen%
5915 \global\let\LWR@closefourteen\LWR@closethirteen%
5916 \global\let\LWR@closethirteen\LWR@closetwelve%
5917 \global\let\LWR@closetwelve\LWR@closeeleven%
5918 \global\let\LWR@closeeleven\LWR@closeten%
5919 \global\let\LWR@closeten\LWR@closenine%
5920 \global\let\LWR@closenine\LWR@closeeight%
5921 \global\let\LWR@closeeight\LWR@closeseven%
5922 \global\let\LWR@closeseven\LWR@closesix%
5923 \global\let\LWR@closesix\LWR@closefive%
5924 \global\let\LWR@closefive\LWR@closefour%
5925 \global\let\LWR@closefour\LWR@closethree%
5926 \global\let\LWR@closethree\LWR@closetwo%
5927 \global\let\LWR@closetwo\LWR@closeone%
```

```
5928 \global\csletcs{LWR@closeone}{LWR@printclose#1}%
5929 \global\let\LWR@closedepthnineteen\LWR@closedeptheighteen%
5930 \global\let\LWR@closedeptheighteen\LWR@closedepthseventeen%
5931 \global\let\LWR@closedepthseventeen\LWR@closedepthsixteen%
5932 \global\let\LWR@closedepthsixteen\LWR@closedepthfifteen%
5933 \global\let\LWR@closedepthfifteen\LWR@closedepthfourteen%
5934 \global\let\LWR@closedepthfourteen\LWR@closedepththirteen%
5935 \global\let\LWR@closedepththirteen\LWR@closedepthtwelve%
5936 \global\let\LWR@closedepthtwelve\LWR@closedeptheleven%
5937 \global\let\LWR@closedeptheleven\LWR@closedepthten%
5938 \global\let\LWR@closedepthten\LWR@closedepthnine%
5939 \global\let\LWR@closedepthnine\LWR@closedeptheight%
5940 \global\let\LWR@closedeptheight\LWR@closedepthseven%
5941 \global\let\LWR@closedepthseven\LWR@closedepthsix%
5942 \global\let\LWR@closedepthsix\LWR@closedepthfive%
5943 \global\let\LWR@closedepthfive\LWR@closedepthfour%
5944 \global\let\LWR@closedepthfour\LWR@closedepththree%
5945 \global\let\LWR@closedepththree\LWR@closedepthtwo%
5946 \global\let\LWR@closedepthtwo\LWR@closedepthone%
5947 \global\csletcs{LWR@closedepthone}{LWR@depth#1}%
```

Error if the deepest depth is no longer \LWR@depthnone, which means that it some-
how has been nested too deeply, or things are not being unnested correctly.

```
5948 \ifdefstring{\LWR@closedepthnineteen}{\LWR@depthnone}%
5949     {}%
5950     {%
5951         \PackageError{lwarp}%
5952             {The document is nested too deeply for Lwarp}%
5953             {PLEASE inform the Lwarp maintainer!}%
5954     }%
5955 }
```

\LWR@popclose Pops one action and its depth off the stacks.

```
5956 \newcommand*{\LWR@popclose}
5957 {%
5958 \global\let\LWR@closeone\LWR@closetwo%
5959 \global\let\LWR@closetwo\LWR@closethree%
5960 \global\let\LWR@closethree\LWR@closefour%
5961 \global\let\LWR@closefour\LWR@closefive%
5962 \global\let\LWR@closefive\LWR@closesix%
5963 \global\let\LWR@closesix\LWR@closeseven%
5964 \global\let\LWR@closeseven\LWR@closeeight%
5965 \global\let\LWR@closeeight\LWR@closenine%
5966 \global\let\LWR@closenine\LWR@closeten%
5967 \global\let\LWR@closeten\LWR@closeeleven%
5968 \global\let\LWR@closeeleven\LWR@closetwelve%
5969 \global\let\LWR@closetwelve\LWR@closethirteen%
5970 \global\let\LWR@closethirteen\LWR@closefourteen%
5971 \global\let\LWR@closefourteen\LWR@closefifteen%
5972 \global\let\LWR@closefifteen\LWR@closesixteen%
5973 \global\let\LWR@closesixteen\LWR@closeseventeen%
5974 \global\let\LWR@closeseventeen\LWR@closeeighteen%
5975 \global\let\LWR@closeeighteen\LWR@closenineteen%
5976 \global\let\LWR@closedepthone\LWR@closedepthtwo%
5977 \global\let\LWR@closedepthtwo\LWR@closedepththree%
5978 \global\let\LWR@closedepththree\LWR@closedepthfour%
5979 \global\let\LWR@closedepthfour\LWR@closedepthfive%
```

```
5980 \global\let\LWR@closedepthfive\LWR@closedepthsix%
5981 \global\let\LWR@closedepthsix\LWR@closedepthseven%
5982 \global\let\LWR@closedepthseven\LWR@closedeptheight%
5983 \global\let\LWR@closedeptheight\LWR@closedepthnine%
5984 \global\let\LWR@closedepthnine\LWR@closedepthten%
5985 \global\let\LWR@closedepthten\LWR@closedeptheleven%
5986 \global\let\LWR@closedeptheleven\LWR@closedepthtwelve%
5987 \global\let\LWR@closedepthtwelve\LWR@closedepththirteen%
5988 \global\let\LWR@closedepththirteen\LWR@closedepthfourteen%
5989 \global\let\LWR@closedepthfourteen\LWR@closedepthfifteen%
5990 \global\let\LWR@closedepthfifteen\LWR@closedepthsixteen%
5991 \global\let\LWR@closedepthsixteen\LWR@closedepthseventeen%
5992 \global\let\LWR@closedepthseventeen\LWR@closedeptheighteen%
5993 \global\let\LWR@closedeptheighteen\LWR@closedepthnineteen%
5994 }
```

```
5995 \end{warpHTML}
```

44   Data arrays

These macros are similar to the arrayjobx package, except that \LWR@setexparray's argument is expanded only once when assigned.

name has no backslash, index can be a number or a text name, and an empty value must be \relax instead of empty.

To assign an empty value:

> \LWR@setexparray{name}{index}{}

5996 \begin{warpHTML}

\LWR@setexparray {⟨*name*⟩} {⟨*index*⟩} {⟨*contents*⟩}

```
5997 \newbool{LWR@setexparray@doingparhooks}
5998
5999 \NewDocumentCommand{\LWR@setexparray}{m m m}{%
```

Temporarily disable paragraph handling during the assignment. This is not done in a group with global assignments because a table may be nested.

```
6000    \let\ifLWR@setexparray@doingparhooks\ifLWR@doingparhooks%
6001    \setbool{LWR@doingparhooks}{false}%
6002    \let\LWR@setexparray@par\par%
6003    \let\par\relax%
```

The name of the control sequence is the given name with the index appended.

```
6004    \xdef\LWR@thisexparrayname{#1#2}%
```

Locally assign the value to the control sequence:

```
6005    \ifstrempty{#3}%
6006        {\csdef{\LWR@thisexparrayname}{}}%
6007        {\csedef{\LWR@thisexparrayname}{#3}}%
```

Restore the paragraph handling:

```
6008    \let\ifLWR@doingparhooks\ifLWR@setexparray@doingparhooks%
6009    \let\par\LWR@setexparray@par%
6010 }
```

\LWR@getexparray {⟨*name*⟩} {⟨*index*⟩}

```
6011 \newcommand*{\LWR@getexparray}[2]{%
6012    \@nameuse{#1#2}%
6013 }
```

```
6014 \end{warpHTML}
```

45   Localizing catcodes

6015 \begin{warpall}

⚠ Misplaced alignment tab character &   Place \StartDefiningTabulars and \StopDefiningTabulars before and after defining macros or environments which include the tabular & character in their definitions.

The catcode of & must be changed before the definitions begin, and must be restored afterwards. Doing so avoids the error

        Misplaced alignment tab character &.

\StartDefiningTabulars   Place before defining something with & in it.

```
6016 \newcommand{\StartDefiningTabulars}{%
6017    \LWR@traceinfo{StartDefiningTabulars}%
6018    \warpHTMLonly{\catcode`\&=\active}%
6019 }
```

\StopDefiningTabulars   Place after defining something with & in it.

```
6020 \newcommand{\StopDefiningTabulars}{%
6021    \LWR@traceinfo{StopDefiningTabulars}%
6022    \warpHTMLonly{\catcode`\&=4}%
6023 }
```

LWR@mathmacro (*bool*)   True if currently defining math macros. Used to disable SVG math hashing and MATHJAX math contents while defining a macro using inline math. Begin a macro, it is not guaranteed that the contents are static, and so the image must be unique. The contents also almost certainly will not be parsed correctly by MATHJAX.

```
6024 \newbool{LWR@mathmacro}
6025 \boolfalse{LWR@mathmacro}
```

\StartDefiningMath   Place before defining something with $ in it.

```
6026 \newcommand{\StartDefiningMath}{%
6027    \LWR@traceinfo{StartDefiningMath}%
6028    \warpHTMLonly{\catcode`\$=\active}%
6029 }
```

\StopDefiningMath   Place after defining something with $ in it.

```
6030 \newcommand{\StopDefiningMath}{%
6031     \LWR@traceinfo{StopDefiningMath}%
6032     \warpHTMLonly{\catcode`\$=3}% math shift
6033 }
```

```
6034 \end{warpall}
```

for HTML output: 6035 \begin{warpHTML}

A definition for & in case it is referred to after \StartDefiningTabulars but outside a tabular.

```
6036 \StartDefiningTabulars
6037 \protected\gdef&{%
6038     \PackageWarning{lwarp}{%
6039         An ampersand is being used inside a tabular\MessageBreak
6040     }%
6041 }%
6042 \StopDefiningTabulars
```

```
6043 \end{warpHTML}
```

46   Localizing dynamic math

Inline svg math usually uses a hash of its contents to generate lateximages which are reusable for multiple instances with the same contents. If the contents may change for each use, such as depending on the current value of a counter, then \inlinemathother must be used before the inline math expression, and \inlinemathnormal must be used after.

For Mathjax, the inline math expression is usually printed for Mathjax to interpret. When marked as dynamic math, the following inline math expression will be displayed as an unhashed inline svg image instead.

For existing code and packages, it may be possible to patch macros after they have been defined, using the xpatch package, which is pre-loaded by lwarp:

```
\xpatchcmd{\macroname}
    {$math expression$}
    {\inlinemathother$math expression$\inlinemathnormal}
    {}
    {\typeout{Error patching macroname.}}
```

for HTML & PRINT: 6044 \begin{warpall}

LWR@dynamicmath (*bool*)   True to mark inline math which is dynamic in nature, thus should not be hashed
Default: false   for reuse.

```
6045 \newbool{LWR@dynamicmath}
6046 \boolfalse{LWR@dynamicmath}
```

\inlinemathother   Place before using $ ... $ or \(... \) if the contents of the math are not static,
                   depending on counters or dynamic macros.

```
6047 \newcommand{\inlinemathother}{%
6048 \LWR@traceinfo{inlinemathother}%
6049 \booltrue{LWR@dynamicmath}%
6050 }
```

\inlinemathnormal   Place after using $ ... $ or \(... \) with dynamic contents.

```
6051 \newcommand{\inlinemathnormal}{%
6052 \LWR@traceinfo{inlinemathnormal}%
6053 \boolfalse{LWR@dynamicmath}%
6054 }
```

```
6055 \end{warpall}
```

47   HTML entities

6056 \begin{warpHTML}

HTML Unicode entities:

```
6057 \let\LWR@origampersand\&
```

\LWR@fontfortags   {⟨*macro name*⟩} {⟨*argument*⟩}

Forces roman TT font for HTML tags.

```
6058 \newrobustcmd*{\LWR@fontfortags}[2]{%
6059     \ifmmode%
6060         \PackageError{lwarp}%
6061             {%
6062                 An HTML tag was generated inside math.\MessageBreak
6063                 This should never occur.\MessageBreak
6064                 Something is broken in Lwarp.\MessageBreak
6065                 Enter `h' for details%
6066             }%
6067             {(Using #1{#2}.)}%
6068     \else%
```

Used by ltjtbook, platex, and related.

```
6069         \ifdef{\romanencoding}%
6070             {%
6071                 \romanencoding{\encodingdefault}%
6072             }%
6073             {%
```

Used by babel:

```
6074                 \ifdef{\latintext}
6075                     {\latintext}
6076                     {\fontencoding\encodingdefault}%
```

```
6077             }%
6078          \LWR@print@normalfont%
6079          \LWR@origttfamily%
6080       \fi%
6081 }
```

\HTMLentity {⟨*entitytag*⟩}

       \protect is in case the tag appears in TOC, LOF, LOT.

```
6082 \newcommand*{\HTMLentity}[1]{%
6083 % \LWR@traceinfo{HTMLentity \detokenize{#1}}%
6084    \begingroup%
6085    \LWR@hook@processingtags%
6086    \LWR@fontfortags{HTMLentity}{\detokenize{#1}}%
6087    \protect\LWR@origampersand\LWR@isolate{#1};%
6088    \endgroup%
6089 % \LWR@traceinfo{HTMLentity done}%
6090 }
```

\HTMLunicode {⟨*hex_unicode*⟩}

```
6091 \newcommand*{\HTMLunicode}[1]{\HTMLentity{\LWR@origpound{}x#1}}
```

\&

```
6092 \renewrobustcmd*{\&}{\HTMLentity{amp}}
```

\textless

```
6093 \let\LWR@origtextless\textless
6094 \renewrobustcmd*{\textless}{\HTMLentity{lt}}
```

\textgreater

```
6095 \let\LWR@origtextgreater\textgreater
6096 \renewrobustcmd*{\textgreater}{\HTMLentity{gt}}
```

\%

```
6097 \let\LWR@origpercent\%
6098 \renewrobustcmd*{\%}{\HTMLentity{percnt}}
```

```
6099 \end{warpHTML}
```

48   HTML filename generation

The filename of the homepage is set to \HomeHTMLFilename.html. The filenames of additional sections start with \HTMLFilename, to which is appended a section number or a simplified section name, depending on FileSectionNames.

for HTML & PRINT: 6100 \begin{warpall}

\BaseJobname The \jobname of the printed version, even if currently compiling the HTML version. I.e. this is the \jobname without _html appended. This is used to set \HomeHTMLFilename if the user did not provide one.

```
6101 \providecommand*{\BaseJobname}{\jobname}
```

\HTMLFilename The prefix for all generated HTML files other than the home page, defaulting to empty. See section 7.6.1.

```
6102 \providecommand*{\HTMLFilename}{}
```

\HomeHTMLFilename The filename of the home page, defaulting to the \BaseJobname. See section 7.6.1.

```
6103 \providecommand*{\HomeHTMLFilename}{\BaseJobname}
```

\SetHTMLFileNumber {⟨*number*⟩}

Sets the file number for the next file to be generated. 0 is the home page. Use just before the next sectioning command, and set it to one less than the desired number of the next section. May be used to generate numbered groups of nodes such as 100+ for one chapter, 200+ for another chapter, etc.

```
6104 \newcommand*{\SetHTMLFileNumber}[1]{%
6105     \setcounter{LWR@htmlfilenumber}{#1}%
6106 }
```

FileSectionNames (*bool*) Selects how to create HTML file names.

Defaults to use section names in the filenames.

```
6107 \newbool{FileSectionNames}
6108 \booltrue{FileSectionNames}
```

```
6109 \end{warpall}
```

for HTML output: `6110 \begin{warpHTML}`

Updated each time a new HTML file is begun. Used to provide HTML previous/next web page links.

```
6111 \newcounter{LWR@HTMLpagenum}
6112 \setcounter{LWR@HTMLpagenum}{0}
```

LWR@htmlseqfilenumber (*Ctr*) A sequential count of the number of each HTML file as it is being created. Number 0 is the home page. Unlike \LWR@htmlfilenumber, this one is known to increment by one for each file. This is used to generate previous /next links for each web page, via labels called \BaseJobname-autofile-*, and the last page is also labelled \BaseJobname-autofile-last.

```
6113 \newcounter{LWR@htmlseqfilenumber}
6114 \setcounter{LWR@htmlseqfilenumber}{0}
```

LWR@setseqfilelabel (*bool*) At each new HTML file, this is false until a sectional unit is used, at which point

this is set true and a label is placed. In this way, the previous/next labels will point to a named section.

```
6115 \newbool{LWR@setseqfilelabel}
6116 \setbool{LWR@setseqfilelabel}{false}
```

LWR@htmlfilenumber (*Ctr*)   Records the number of each HTML file as it is being created. Number 0 is the home page. This might not be sequential, as the user may use \SetHTMLFileNumber to create groups of numbered nodes.

```
6117 \newcounter{LWR@htmlfilenumber}
6118 \setcounter{LWR@htmlfilenumber}{0}
```

\LWR@htmlsectionfilename {⟨*htmlfilenumber or name*⟩}

Prints the filename for a given section: \HTMLFilename{}filenumber/name.html

```
6119 \newcommand*\LWR@htmlsectionfilename}[1]{%
6120 \LWR@traceinfo{LWR@htmlsectionfilename A !\detokenize{#1}!}%
6121 \begingroup%
```

Disable CJK xpinyin while generating file names.

```
6122 \LWR@disablepinyin%
```

Section 0 or empty is given the home filename. The filename must be detokenized for underscores.

```
6123 % \LWR@traceinfo{about to assign temp}%
6124 \LWR@sanitize{#1}%
6125 \LWR@traceinfo{about to compare with ??}%
6126 \ifdefstring{\LWR@sanitized}{??}
6127     {\LWR@traceinfo{found ??}}%
6128     {\LWR@traceinfo{not found ??}}%
6129 \LWR@traceinfo{about to compare with zero or empty}%
6130 \ifboolexpr{
6131     test {\ifdefstring{\LWR@sanitized}{0}} or
6132     test {\ifdefstring{\LWR@sanitized}{}} or
6133     test {\ifdefstring{\LWR@sanitized}{??}}
6134 }
6135 {%
6136     \LWR@traceinfo{LWR@htmlsectionfilename B \HomeHTMLFilename.html}%
6137     \HomeHTMLFilename.html%
6138 }%
```

For a LATEX section named "Index" or "index" without a prefix, create a filename with a trailing -0 to avoid colliding with the HTML filename index.html:

```
6139 {%
6140     \LWR@traceinfo{LWR@htmlsectionfilename C \LWR@sanitized}%
6141     \ifboolexpr{
6142             test{\ifdefvoid{\HTMLFilename}} and
6143             (
6144                     test{\ifdefstring{\LWR@sanitized}{Index}} or
6145                     test{\ifdefstring{\LWR@sanitized}{index}}
6146             )
6147     }%
6148     {%
```

```
6149        \LWR@traceinfo{Adding a zero to the index filename.}%
6150        \LWR@sanitized-0.html%
6151    }%
```

Otherwise, create a filename with the chosen prefix:

```
6152    {%
6153        \HTMLFilename\LWR@isolate{\LWR@sanitized}.html%
6154    }%
6155 }%
6156 \LWR@traceinfo{LWR@htmlsectionfilename Z}%
6157 \endgroup%
6158 }
```

\LWR@htmlrefsectionfilename {⟨*label*⟩}

Prints the filename for the given label

```
6159 \newcommand*{\LWR@htmlrefsectionfilename}[1]{%
6160    \LWR@traceinfo{LWR@htmlrefsectionfilename: !\detokenize{#1}!}%
```

\LWR@nullfonts to allow math in a section name.

```
6161    \begingroup%
6162    \LWR@nullfonts%
6163    \LWR@htmlsectionfilename{\LWR@htmlfileref{#1}}%
6164    \endgroup%
6165    \LWR@traceinfo{LWR@htmlrefsectionfilename: done}%
6166 }
```

```
6167 \end{warpHTML}
```

49   Homepage link

`6168 \begin{warpall}`

\linkhomename Holds the default name for the home link.

```
6169 \newcommand{\linkhomename}{Home}
```

```
6170 \end{warpall}
```

`6171 \begin{warpHTML}`

\LinkHome May be used wherever you wish to place a link back to the homepage. The filename must be detokenized for underscores.

```
6172 \newcommand*{\LinkHome}{%
6173    \LWR@subhyperrefclass{\HomeHTMLFilename.html}{\linkhomename}{linkhome}%
6174 }
```

```
6175 \end{warpHTML}
```

for PRINT output: 6176 `\begin{warpprint}`

\LinkHome May be used wherever you wish to place a link back to the homepage. For print
output, if hyperref is available a hyperlink to the first page is used, named by
\linkhomename. If hyperref is not available, a pageref is used instead.

\BaseJobname is included in the link label in case multiple documents are cross-
referenced.

```
6177 \AtBeginDocument{
6178 \@ifundefined{hyperref}{
6179     \newcommand*{\LinkHome}{%
6180         \linkhomename\ --- page \pageref{\BaseJobname-page-LWRfirstpage}%
6181     }
6182 }{
6183     \newcommand*{\LinkHome}{%
6184         \hyperref[\BaseJobname-page-LWRfirstpage]{\linkhomename}%
6185     }
6186 }
6187 }
6188
6189 \AfterEndPreamble{\label{\BaseJobname-page-LWRfirstpage}}
```

6190 `\end{warpprint}`

for HTML output: 6191 `\begin{warpHTML}`

\LWR@topnavigation Creates a link to the homepage at the top of the page for use when the window is
too narrow for the sideⓉⓄⒸ.

```
6192 \newcommand*{\LWR@topnavigation}{%
6193     \LWR@htmlelementclassline{nav}{topnavigation}{\LinkHome}
6194 }
```

\LWR@botnavigation Creates a link to the homepage at the bottom of the page for use when the window
is too narrow for the sideⓉⓄⒸ.

```
6195 \newcommand*{\LWR@botnavigation}{%
6196     \LWR@htmlelementclassline{nav}{botnavigation}{\LinkHome}
6197 }
```

6198 `\end{warpHTML}`

50 Previous/next navigation links

for HTML & PRINT: 6199 `\begin{warpall}`

\linkpreviousname What to call the link to the previous web page.

6200 `\newcommand*{\linkpreviousname}{Previous}`

\linknextname What to call the link to the next web page.

6201 `\newcommand*{\linknextname}{Next}`

6202 \end{warpall}

6203 \begin{warpprint}

\LinkPrevious   Creates a link to the previous web page if there is one.

6204 \newcommand*{\LinkPrevious}{}

\LinkNext   Creates a link to the next web page if there is one.

6205 \newcommand*{\LinkNext}{}

6206 \end{warpprint}

6207 \begin{warpHTML}

\LinkPrevious   Creates a link to the previous web page if there is one.

The links refer to the LaTeX labels \Basejobname-autofile-*

```
6208 \newcommand*{\LinkPrevious}{%
6209     \ifnumless{\value{LWR@htmlseqfilenumber}}{1}{}{%
6210         \setcounter{LWR@tempcountone}{\value{LWR@htmlseqfilenumber}-1}%
6211         \LWR@subhyperrefclass{%
6212             \LWR@htmlrefsectionfilename{%
6213                 \BaseJobname-autofile-\arabic{LWR@tempcountone}%
6214             }%
6215         }{\linkpreviousname}{linkhome}%
6216     }%
6217 }
```

\LinkNext   Creates a link to the next web page if there is one.

The links refer to the LaTeX labels \Basejobname-autofile-*
and the last is the label \Basejobname-autofile-last

```
6218 \newcommand*{\LinkNext}{%
6219     \ifcsdef{r@\BaseJobname-autofile-last@lwarp}{%
6220         \edef\LWR@tempone{%
6221         \LWR@htmlfileref{\BaseJobname-autofile-\arabic{LWR@htmlseqfilenumber}}%
6222         }%
6223         \edef\LWR@temptwo{%
6224             \LWR@htmlfileref{\BaseJobname-autofile-last}%
6225         }%
6226         \ifdefequal{\LWR@tempone}{\LWR@temptwo}{}{%
6227             \setcounter{LWR@tempcountone}{\value{LWR@htmlseqfilenumber}+1}%
6228             \LWR@subhyperrefclass{%
6229                 \LWR@htmlrefsectionfilename{%
6230                     \BaseJobname-autofile-\arabic{LWR@tempcountone}%
6231                 }%
6232             }{\linknextname}{linkhome}%
6233         }%
6234     }{}%
6235 }
```

6236 \end{warpHTML}

51 `\LWRPrintStack` diagnostic tool

⚠ Diagnostics tool: Prints the LaTeX nesting depth values for the stack levels.
`\LWR@startpars` is used before printing the stack, so that `\LWRPrintStack` may be
called from anywhere in the normal text flow.

for HTML output: 6237 `\begin{warpHTML}`

`\LWRPrintStack` Prints the closedepth stack.

```
6238 \newcommand*{\LWR@subprintstack}{
6239 \LWR@closedepthone\ \LWR@closedepthtwo\ \LWR@closedepththree\
6240 \LWR@closedepthfour\ \LWR@closedepthfive\ \LWR@closedepthsix\
6241 \LWR@closedepthseven\ \LWR@closedeptheight\ \LWR@closedepthnine\
6242 \LWR@closedepthten\ \LWR@closedeptheleven\ \LWR@closedepthtwelve\
6243 \LWR@closedepththirteen\ \LWR@closedepthfourteen\ \LWR@closedepthfifteen\
6244 \LWR@closedepthsixteen\ \LWR@closedepthseventeen\ \LWR@closedeptheighteen\
6245 \LWR@closedepthnineteen\
6246 }
6247
6248 \newcommand*{\LWRPrintStack}{
6249 \LWR@startpars
6250 \LWR@subprintstack
6251 }
```

6252 `\end{warpHTML}`

for PRINT output: 6253 `\begin{warpprint}`

6254 `\newcommand*{\LWRPrintStack}{}`

6255 `\end{warpprint}`

52 Closing stack levels

for HTML output: 6256 `\begin{warpHTML}`

Close one nested level:

```
6257 \newcommand*{\LWR@closeoneprevious}{%
6258
6259 \LWR@closeone
6260
6261 \LWR@popclose
6262 }
```

`\LWR@closeprevious` {⟨*sectintype*⟩} Close everything up to the given depth:

```
6263 \newcommand*{\LWR@closeprevious}[1]{%
6264 \LWR@traceinfo{%
6265     LWR@closeprevious to depth \csuse{LWR@depth#1}, %
6266     depths are \LWR@subprintstack%
6267 }%
```

Close any pending paragraph:

6268 `\LWR@stoppars%`

Close anything nested deeper than the desired depth. First close anything deeper, then at most one of the same level.

```
6269 \whileboolexpr{test{\ifnumcomp{\LWR@closedepthone}{>}{\csuse{LWR@depth#1}}}}%
6270 {%
6271     \LWR@traceinfo{LWR@closeprevious: closing out depth \LWR@closedepthone}%
6272     \LWR@closeoneprevious%
6273 }%
6274 \ifboolexpr{test{\ifnumcomp{\LWR@closedepthone}{=}{\csuse{LWR@depth#1}}}}%
6275 {%
6276     \LWR@traceinfo{LWR@closeprevious: closing out depth \LWR@closedepthone}%
6277     \LWR@closeoneprevious%
6278 }{}%
6279 \LWR@traceinfo{LWR@closeprevious: done, depths are \LWR@subprintstack}%
6280 }
```

6281 `\end{warpHTML}`

53   PDF pages and styles

6282 `\begin{warpHTML}`

`\LWR@forcenewpage` New PDF page a before major environment.

This is used just before major environments, such as verse. Reduces the chance of an environment overflowing the HTML PDF output page.

```
6283 \newcommand{\LWR@forcenewpage}{%
6284 \LWR@traceinfo{LWR@forcenewpage}%
6285 \ifinner\else%
6286     \LWR@traceinfo{LWR@forcenewpage A}%
6287     \LWR@stoppars%
6288     \LWR@traceinfo{LWR@forcenewpage B}%
6289     \LWR@maybe@orignewpage%
6290     \LWR@traceinfo{LWR@forcenewpage C}%
6291     \LWR@startpars%
6292 \fi%
6293 \LWR@traceinfo{LWR@forcenewpage done}%
6294 }
```

`\pagestyle`, etc. are nullified for HTML output.

`\pagestyle` {⟨*style*⟩}

6295 `\renewcommand*{\pagestyle}[1]{}`

`\thispagestyle` {⟨*style*⟩}

6296 `\renewcommand*{\thispagestyle}[1]{}`

\markboth {⟨*left*⟩} {⟨*right*⟩}

6297 \renewcommand*{\markboth}[2]{}

\markright {⟨*right*⟩}

6298 \renewcommand*{\markright}[1]{}

\raggedbottom

6299 \renewcommand*{\raggedbottom}{}

\flushbottom

6300 \renewcommand*{\flushbottom}{}

\sloppy

6301 \renewcommand*{\sloppy}{}

\fussy

6302 \renewcommand*{\fussy}{}

\pagenumbering * {⟨*commands*⟩}

6303 \RenewDocumentCommand{\pagenumbering}{s m}{}

6304 \end{warpHTML}

54   HTML tags, spans, divs, elements

for HTML output: 6305 \begin{warpHTML}

54.1   Mapping LaTeX sections to HTML sections

6306 \newcommand*{\LWR@tagtitle}{h1}
6307 \newcommand*{\LWR@tagtitleend}{/h1}
6308 \newcommand*{\LWR@tagbook}{div class=\textquotedbl{}book\textquotedbl}
6309 \newcommand*{\LWR@tagbookend}{/div}
6310 \newcommand*{\LWR@tagpart}{h2}
6311 \newcommand*{\LWR@tagpartend}{/h2}
6312 \newcommand*{\LWR@tagchapter}{h3}
6313 \newcommand*{\LWR@tagchapterend}{/h3}
6314 \newcommand*{\LWR@tagsection}{h4}
6315 \newcommand*{\LWR@tagsectionend}{/h4}
6316 \newcommand*{\LWR@tagsubsection}{h5}
6317 \newcommand*{\LWR@tagsubsectionend}{/h5}
6318 \newcommand*{\LWR@tagsubsubsection}{h6}
6319 \newcommand*{\LWR@tagsubsubsectionend}{/h6}

```
6320 \newcommand*{\LWR@tagparagraph}{span class=\textquotedbl{}paragraph\textquotedbl}
6321 \newcommand*{\LWR@tagparagraphend}{/span}
6322 \newcommand*{\LWR@tagsubparagraph}{span class=\textquotedbl{}subparagraph\textquotedbl}
6323 \newcommand*{\LWR@tagsubparagraphend}{/span}
6324
6325 \newcommand*{\LWR@tagregularparagraph}{p}
```

54.2   Hook while processing tags

\LWR@hook@processingtags   This is used to disable special text processing while processing HTML tags. Special
(*Hook*) [lwarp]   processing includes that done by babel-french, luavlna, xevlna.

\LWR@hook@processingtags   Disable special text processing while generating tags. Replaces \LWR@FBcancel in
most places.

```
6326 \newcommand*{\LWR@hook@processingtags}{}
```

54.3   Babel-French tag modifications

Adjust babel-french for HTML spaces. So far, this only works for *pdflatex* and
xelatex.

(*Emulates or patches code by* DANIEL FLIPO.)

```
6327 \providecommand*{\LWR@FBcancel}{}
6328
6329 \AtBeginDocument{%
```

In some circumstances, \NoAutoSpacing may be defined when \frenchbsetup is
not.

```
6330 \@ifundefined{NoAutoSpacing}%
6331     {}%
6332     {%
6333         \LetLtxMacro\LWR@FBcancel\NoAutoSpacing%
6334         \appto{\LWR@hook@processingtags}{\LWR@FBcancel}%
6335     }%
6336
6337 \@ifundefined{frenchbsetup}%
6338 {}%
6339 {%
6340     \frenchbsetup{FrenchFootnotes=false}%
6341 %
6342     \renewrobustcmd*{\FBcolonspace}{%
6343         \begingroup%
6344         \LWR@hook@processingtags%
6345         \LWR@origampersand{}nbsp;%
6346         \endgroup%
6347     }%
6348     \renewrobustcmd*{\FBthinspace}{%
6349         \begingroup%
6350         \LWR@hook@processingtags%
6351         \LWR@origampersand\LWR@origpound{}x202f;% \,
6352         \endgroup%
```

```
6353     }%
6354     \renewrobustcmd*{\FBguillspace}{%
6355         \begingroup%
6356         \LWR@hook@processingtags%
6357         \LWR@origampersand{}nbsp;% ~,   for \og xyz \fg{}
6358         \endgroup%
6359     }%
6360     \DeclareDocumentCommand{\FBmedkern}{}{%
6361         \begingroup%
6362         \LWR@hook@processingtags%
6363         \LWR@origampersand\LWR@origpound{}x202f;% \,
6364         \endgroup%
6365     }%
6366     \DeclareDocumentCommand{\FBthickkern}{}{%
6367         \begingroup%
6368         \LWR@hook@processingtags%
6369         \LWR@origampersand{}nbsp;% ~
6370         \endgroup%
6371     }%
6372     \renewrobustcmd*{~}{\HTMLentity{nbsp}}% was overwritten by babel-french
6373     \ifFBunicode%
6374     \else%
6375         \DeclareTextSymbol{\FBtextellipsis}{LY1}{133}%
6376         \DeclareTextCommandDefault{\FBtextellipsis}{\textellipsis\xspace}%
6377     \fi%
6378 }%
6379 }
```

54.4 HTML output formatting

Helps format the output HTML code for human readability.

\LWR@indentHTML Newline and indent the output HTML code.

```
6380 \newcommand*{\LWR@indentHTML}{%
6381     \LWR@orignewline\LWR@origrule{2em}{0pt}%
6382 }
```

\LWR@indentHTMLtwo Newline and indent the output HTML code.

```
6383 \newcommand*{\LWR@indentHTMLtwo}{%
6384     \LWR@orignewline\LWR@origrule{4em}{0pt}%
6385 }
```

54.5 HTML tags

\LWR@htmltagc {⟨tag⟩} Break ligatures and use upright apostrophes in HTML tags.

\protect is in case the tag appears in TOC, LOF, LOT.

```
6386 \newcommand*{\LWR@htmltagc}[1]{%
6387     \LWR@traceinfo{LWR@htmltagc !\detokenize{#1}!}%
6388     \begingroup%
6389     \LWR@hook@processingtags%
```

```
6390        \LWR@fontfortags{LWR@htmltagc}{\detokenize{#1}}%
6391        \protect\LWR@origtextless%
6392        \LWR@traceinfo{LWR@htmltagc B}%
6393        \LWR@isolate{#1}%
6394        \LWR@traceinfo{LWR@htmltagc C}%
6395        \protect\LWR@origtextgreater%
6396        \endgroup%
6397        \LWR@traceinfo{LWR@htmltagc done}%
6398 }
```

\LWR@spanwarnformat {⟨*object*⟩}

        Warns if the given object is used inside a span.

```
6399 \newcommand*{\LWR@spanwarnformat}[1]{%
6400        \ifnumcomp{\value{LWR@spandepth}}{>}{0}{%
6401            \PackageWarning{lwarp}{%
6402                A #1 is being used inside a span.\MessageBreak
6403                Formatting may be lost,%
6404            }%
6405        }{}%
6406 }
```

\LWR@spanwarninvalid {⟨*object*⟩}

        Warns if the given object is used inside a span.

```
6407 \newcommand*{\LWR@spanwarninvalid}[1]{%
6408        \ifnumcomp{\value{LWR@spandepth}}{>}{0}{%
6409            \PackageWarning{lwarp}{%
6410                A #1 is being used inside a span.\MessageBreak
6411                This generates invalid HTML,%
6412            }%
6413        }{}%
6414 }
```

LWR@nestspan (*env.*) Disable minipage, \parbox, and HTML <div>s inside a .

⚠   \begin{LWR@nestspan} must follow the opening tag to allow a paragraph to start if the span is at the beginning of a new paragraph.

⚠   \end{LWR@nestspan} must follow the or a <p> may appear inside the span.

```
6415 \newcommand*{\LWR@nestspanitem}{%
6416        \if@newlist\else{
6417            \LWR@htmltagc{br /}%
6418            \LWR@orignewline%
6419        }\fi%
6420        \LWR@origitem%
6421 }
6422
6423 \newenvironment*{LWR@nestspan}
6424 {%
6425        \LWR@traceinfo{LWR@nestspan starting}%
6426        \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
6427        {%
6428            \LWR@traceinfo{LWR@nestspan: inside a lateximage}%
```

```
6429      }%
6430      {% not in a lateximage
6431          \LWR@traceinfo{LWR@nestspan: NOT inside a lateximage}%
6432          \addtocounter{LWR@spandepth}{1}%
```

Nullify several objects inside the span:

```
6433          \RenewDocumentEnvironment{minipage}{O{t} o O{t} m}%
6434              {\LWR@spanwarnformat{minipage or \protect\parbox}}%
6435              {}%
6436          \RenewDocumentEnvironment{BlockClass}{o D(){} m}%
6437              {\LWR@spanwarnformat{multi-paragraph object}}%
6438              {}%
6439          \RenewDocumentEnvironment{LWR@BlockClassWP}{m m D(){} m}%
6440              {\LWR@spanwarnformat{multi-paragraph object}}%
6441              {}%
6442          \renewcommand{\BlockClassSingle}[2]{%
6443              {\LWR@spanwarnformat{multi-paragraph object}}%
6444              ##2%
6445          }%
6446          \renewcommand{\LWR@forcenewpage}{}%
6447          \renewcommand{\LWR@liststart}{\LetLtxMacro\item\LWR@nestspanitem}%
6448          \renewcommand{\LWR@listend}{\leavevmode}%
6449          \renewenvironment{quote}{\LWR@htmltagc{br /}}{\LWR@htmltagc{br /}}%
6450        \renewenvironment{quotation}{\LWR@htmltagc{br /}}{\LWR@htmltagc{br /}}%
6451      }% not in a lateximage
6452      \LWR@traceinfo{LWR@nestspan starting: done}%
6453 }% starting env
6454 {% ending env
6455      \LWR@traceinfo{LWR@nestspan ending}%
6456      \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
6457      {}%
6458      {\addtocounter{LWR@spandepth}{-1}}%
6459      \LWR@traceinfo{LWR@nestspan ending: done}%
6460 }
```

\LWR@htmlspan {⟨*tag*⟩} {⟨*text*⟩}

⚠️   \LWR@spandepth is used to ensure that paragraph tags are not generated inside a span. The exact sequence of when to add and subtract the counter is important to correctly handle the paragraph tags before and after the span.

```
6461 \NewDocumentCommand{\LWR@htmlspan}{m +m}{%
6462      \LWR@ensuredoingapar%
6463      \LWR@htmltagc{#1}%
6464      \begin{LWR@nestspan}%
6465      #2%
6466      \LWR@htmltagc{/#1}%
6467      \end{LWR@nestspan}%
6468 }
```

\LWR@htmlspanclass [⟨*style*⟩] (⟨*aria role*⟩) {⟨*class*⟩} {⟨*text*⟩}

```
6469 \NewDocumentCommand{\LWR@htmlspanclass}{o D(){} m +m}{%
6470      \LWR@traceinfo{LWR@htmlspanclass #3}%
6471      \LWR@ensuredoingapar%
6472      \ifblank{#2}%
6473          {\LWR@subhtmlelementclass{span}[#1]{#3}}%
```

```
6474          {\LWR@subhtmlelementclass{span}[#1](#2){#3}}%
6475      \begin{LWR@nestspan}%
6476      #4%
6477      \LWR@htmltagc{/span}%
6478      \end{LWR@nestspan}%
6479      \LWR@traceinfo{LWR@htmlspanclass done}%
6480 }
```

\LWR@htmltag *{⟨*tag*⟩}*

> Print an HTML tag: <tag>

```
6481 \newcommand*{\LWR@htmltag}[1]{%
6482      \LWR@htmltagc{#1}%
6483 }
```

54.6   Block tags and comments

In the following, \origttfamily breaks ligatures, which may not be used for HTML codes:

\LWR@htmlopencomment
\LWR@htmlclosecomment

```
6484 \newcommand*{\LWR@htmlopencomment}{%
6485 % \LWR@traceinfo{LWR@htmlopencomment}%
6486      \begingroup%
6487      \LWR@hook@processingtags%
6488      \LWR@fontfortags{LWR@htmlopencomment}{}%
6489      \LWR@print@mbox{\LWR@origtextless{}!-\/-}%
6490      \endgroup%
6491 }
6492
6493 \newcommand*{\LWR@htmlclosecomment}{%
6494 % \LWR@traceinfo{LWR@htmlclosecomment}%
6495      \begingroup%
6496      \LWR@hook@processingtags%
6497      \LWR@fontfortags{LWr@htmlclosecomment}{}%
6498      \LWR@print@mbox{-\/-\LWR@origtextgreater}%
6499      \endgroup%
6500 }
```

\LWR@htmlcomment *{⟨*comment*⟩}*

```
6501 \newcommand{\LWR@htmlcomment}[1]{%
6502      \ifmmode%
6503      \else%
6504          \LWR@htmlopencomment{}%
6505          {%
6506              \LWR@print@normalfont%
6507              \LWR@origttfamily% break ligatures
6508              #1%
6509          }%
6510          \LWR@htmlclosecomment{}%
6511      \fi%
6512 }
```

\LWR@htmlblockcomment {⟨*comment*⟩}

```
6513 \newcommand{\LWR@htmlblockcomment}[1]
6514     {\LWR@stoppars\LWR@htmlcomment{#1}\LWR@startpars}
```

\LWR@htmlblocktag {⟨*tag*⟩} print a stand-alone HTML tag

```
6515 \newcommand*{\LWR@htmlblocktag}[1]{%
6516     \LWR@stoppars%
6517     \LWR@htmltag{#1}%
6518     \LWR@startpars%
6519 }
```

54.7   Div class and element class

\LWR@subhtmlelementclass {⟨*element*⟩} [⟨*style*⟩] (⟨*aria role*⟩) {⟨*class*⟩}

Factored and reused in several places.

The trailing spaces allow more places for a line break.

The use of \textquotedbl instead of " provides improved compatibility with xeCJK.

```
6520 \NewDocumentCommand{\LWR@subhtmlelementclass}{m O{} D(){} m}{%
6521     \LWR@traceinfo{LWR@subhtmlelementclass !#1!#4!}%
6522     \ifblank{#2}%
6523     {% empty style
6524         \LWR@htmltag{%
6525             #1%
6526             \ifblank{#3}{}{ role=\textquotedbl#3\textquotedbl}% spaces
6527             \ifblank{#4}{}{ class=\textquotedbl#4\textquotedbl}% spaces
6528         }%
6529     }%
6530     {% non-empty style
6531         \LWR@htmltag{%
6532             #1\LWR@indentHTML%
6533          \ifblank{#3}{}{role=\textquotedbl#3\textquotedbl\LWR@indentHTML}%
6534         \ifblank{#4}{}{class=\textquotedbl#4\textquotedbl\LWR@indentHTML}%
6535             style=\textquotedbl#2\textquotedbl\LWR@orignewline%
6536         }%
6537     }%
6538     \LWR@traceinfo{LWR@subhtmlelementclass done}%
6539 }
```

\LWR@htmlelementclass {⟨*element*⟩} [⟨*style*⟩] (⟨*aria role*⟩) {⟨*class*⟩}

```
6540 \NewDocumentCommand{\LWR@htmlelementclass}{m o D(){} m}{%
6541     \LWR@stoppars%
6542     \LWR@forceemptyline%
6543     \ifblank{#3}%
6544         {\LWR@subhtmlelementclass{#1}[#2]{#4}}%
6545         {\LWR@subhtmlelementclass{#1}[#2](#3){#4}}%
6546     \LWR@startpars%
6547 }
```

\LWR@htmlelementclassend {⟨*element*⟩} {⟨*class*⟩}

```
6548 \newcommand*{\LWR@htmlelementclassend}[2]{%
6549     \LWR@stoppars%
6550     \LWR@htmltag{/#1}%
6551     \ifbool{HTMLDebugComments}{%
6552         \ifblank{#2}%
6553             {\LWR@htmlcomment{End of #1}}%
6554             {\LWR@htmlcomment{End of #1 ``#2''}}%
6555     }{}%
6556     \LWR@startpars%
6557 }
```

\LWR@htmldivclass [⟨*style*⟩] (⟨*aria role*⟩) {⟨*class*⟩}

```
6558 \NewDocumentCommand{\LWR@htmldivclass}{o D(){} m}{%
6559     \ifblank{#2}
6560         {\LWR@htmlelementclass{div}[#1]{#3}}%
6561         {\LWR@htmlelementclass{div}[#1](#2){#3}}%
6562 }
```

\LWR@htmldivclassend {⟨*class*⟩}

```
6563 \newcommand*{\LWR@htmldivclassend}[1]{%
6564     \LWR@htmlelementclassend{div}{#1}%
6565 }
```

54.8   Single-line elements

A single-line element, without a paragraph tag for the line of text:

\LWR@htmlelementclassline {⟨*element*⟩} [⟨*style*⟩] {⟨*class*⟩} {⟨*text*⟩}

```
6566 \NewDocumentCommand{\LWR@htmlelementclassline}{m o m +m}{%
6567     \LWR@stoppars
6568     \LWR@forceemptyline%
6569     \LWR@subhtmlelementclass{#1}[#2]{#3}%
6570     #4%
6571     \LWR@htmltag{/#1}
6572     \LWR@startpars
6573 }
```

54.9   HTML5 semantic elements

\LWR@htmlelement {⟨*element*⟩}

```
6574 \newcommand*{\LWR@htmlelement}[1]{%
6575     \LWR@htmlblocktag{#1}
6576 }
```

\LWR@htmlelementend {⟨*element*⟩}

```
6577 \newcommand*{\LWR@htmlelementend}[1]{%
6578     \LWR@stoppars
6579     \LWR@htmltag{/#1}
6580     \LWR@startpars
6581 }
6582
6583 \end{warpHTML}
```

54.10   High-level block and inline classes

These are high-level commands which allow the creation of arbitrary block or inline sections which may be formatted with css.

Nullified versions are provided for print mode.

For other direct-formatting commands, see section 97.

BlockClass (*env.*) [⟨*style*⟩] (⟨*aria role*⟩) {⟨*class*⟩}   High-level interface for <div> classes.

Ex: \begin{BlockClass}{class} text \end{BlockClass}

for HTML & PRINT:
```
6584 \begin{warpall}
6585 \NewDocumentEnvironment{BlockClass}{o D(){} m}{}{}
6586 \end{warpall}
```

for HTML output:
```
6587 \begin{warpHTML}
6588
6589 \NewDocumentEnvironment{LWR@HTML@BlockClass}{o D(){} m}%
6590     {\LWR@htmldivclass[#1](#2){#3}}%
6591     {\LWR@htmldivclassend{#3}}
6592
6593 \LWR@formattedenv{BlockClass}
6594 \end{warpHTML}
```

\BlockClassSingle {⟨*class*⟩} {⟨*text*⟩}   A single-line <div>, without a paragraph tag for the line of text.

for HTML & PRINT:
```
6595 \begin{warpall}
6596 \newcommand{\BlockClassSingle}[2]{#2}
6597 \end{warpall}
```

for HTML output:
```
6598 \begin{warpHTML}
6599 \newcommand{\LWR@HTML@BlockClassSingle}[2]{%
6600     \LWR@htmlelementclassline{div}{#1}{#2}%
6601 }
6602
6603 \LWR@formatted{BlockClassSingle}
6604 \end{warpHTML}
```

\InlineClass (⟨*WP style*⟩) [⟨*style*⟩] {⟨*class*⟩} {⟨*text*⟩}

High-level interface for inline span classes.

(⟨*WP style*⟩) is css styling to add when formatting for a word processor import.

[⟨*style*⟩] is the css styling to add when not formatting for a word processor.

for HTML & PRINT:
```
6605 \begin{warpall}
6606 \NewDocumentCommand{\InlineClass}{D(}{)}{}{} o m +m}{#4}%
6607 \end{warpall}
```

for HTML output:
```
6608 \begin{warpHTML}
6609 \NewDocumentCommand{\LWR@HTML@InlineClass}{D(}{)}{}{} o m +m}{%
6610     \LWR@traceinfo{LWR@HTML@InlineClass #3}%
6611     \ifbool{FormatWP}{%
6612         \LWR@traceinfo{LWR@HTML@InlineClass: FormatWP}%
6613         \LWR@htmlspanclass[#1]{#3}{#4}%
6614     }{%
6615         \LWR@traceinfo{LWR@HTML@InlineClass: not FormatWP}%
6616         \LWR@htmlspanclass[#2]{#3}{#4}%
6617     }%
6618     \LWR@traceinfo{LWR@HTML@InlineClass: done}%
6619 }
6620
6621 \LWR@formatted{InlineClass}
6622 \end{warpHTML}
```

LWR@BlockClassWP (*env.*) {⟨*WPstyle*⟩} {⟨*HTMLstyle*⟩} (⟨*aria role*⟩) {⟨*class*⟩}   Low-level interface for <div> classes with an automatic float ID. These are often used when \ifbool{FormatWP}.

The use of \textquotedbl instead of " provides improved compatibility with xeCJK.

for HTML & PRINT:
```
6623 \begin{warpall}
6624 \NewDocumentEnvironment{LWR@BlockClassWP}{m m D(){} m}{}{}
6625 \end{warpall}
```

for HTML output:
```
6626 \begin{warpHTML}
6627 \NewDocumentEnvironment{LWR@HTML@LWR@BlockClassWP}{m m D(){} m}%
6628     {%
6629         \LWR@stoppars%
6630         \ifbool{FormatWP}%
6631         {%
6632             \addtocounter{LWR@thisautoidWP}{1}%

6633             \LWR@htmltag{%
6634                 div class=\textquotedbl#4\textquotedbl\ % space
6635                 id=\textquotedbl%
6636                     \LWR@print@mbox{autoidWP-\arabic{LWR@thisautoidWP}}%
6637                 \textquotedbl%
6638                 \ifblank{#3}{}{ role=\textquotedbl#3\textquotedbl}%
6639                 \ifblank{#1}{}{ style=\textquotedbl#1\textquotedbl}%
6640             }%
6641         }% FormatWP
6642         {% not FormatWP
6643             \LWR@htmltag{%
6644                 div class=\textquotedbl#4\textquotedbl%
6645                 \ifblank{#3}{}{ role=\textquotedbl#3\textquotedbl}%
6646                 \ifblank{#2}{}{ style=\textquotedbl#2\textquotedbl}%
6647             }%
6648         }% not FormatWP
6649         \LWR@startpars%
6650     }
6651     {\LWR@htmldivclassend{#4}}
6652
```

```
6653 \LWR@formattedenv{LWR@BlockClassWP}
6654 \end{warpHTML}
```

54.11   Closing HTML tags

`6655 \begin{warpHTML}`

Sections H1, H2, etc. do not need a closing HTML tag, but we add a comment for readability:

```
6656 \newcommand*{\LWR@printclosebook}
6657     {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing book}}{}}
6658 \newcommand*{\LWR@printclosepart}
6659     {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing part}}{}}
6660 \newcommand*{\LWR@printclosechapter}
6661     {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing chapter}}{}}
6662 \newcommand*{\LWR@printclosesection}
6663     {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing section}}{}}
6664 \newcommand*{\LWR@printclosesubsection}
6665     {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing subsection}}{}}
6666 \newcommand*{\LWR@printclosesubsubsection}
6667     {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing subsubsection}}{}}
6668 \newcommand*{\LWR@printcloseparagraph}
6669     {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing paragraph}}{}}
6670 \newcommand*{\LWR@printclosesubparagraph}
6671     {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing subparagraph}}{}}
```

Lists require closing HTML tags:

```
6672 \newcommand*{\LWR@printcloselistitem}
6673     {\LWR@htmltag{/li}}
6674 \newcommand*{\LWR@printclosedescitem}
6675     {\LWR@htmltag{/dd}}
6676 \newcommand*{\LWR@printcloseitemize}
6677     {\LWR@htmltag{/ul}}
6678 \newcommand*{\LWR@printcloseenumerate}
6679     {\LWR@htmltag{/ol}}
6680 \newcommand*{\LWR@printclosedescription}
6681     {\LWR@htmltag{/dl}}
```

```
6682 \end{warpHTML}
```

55   Paragraph handling

These commands generate the HTML paragraph tags when allowed and required.

Paragraph tags are or are not allowed depending on many conditions. Section 56 has high-level commands which allow paragraph-tag generation to start/stop. Even when allowed (LWR@doingstartpars), tags are not generated until a LATEX paragraph is being used (LWR@doingapar). LWR@lateximagedepth is used to prevent nesting tags inside a lateximage. LWR@spandepth is used to prevent nesting paragraph tags inside a paragraph, which became important inside \fbox commands and other spans.

The LaTeX paragraph hooks are used to manage tag creation.

for HTML output: 6683 `\begin{warpHTML}`

LWR@spandepth (*Ctr*) Do not create paragraph tags inside of an HTML span.

6684 `\newcounter{LWR@spandepth}`
6685 `\setcounter{LWR@spandepth}{0}`

LWR@doingparhooks (*bool*) Tells whether the lwarp paragraph hooks are to be active.

6686 `\newbool{LWR@doingparhooks}`
6687 `\boolfalse{LWR@doingparhooks}`

LWR@in@multirow@par (*bool*) Tells whether to generate break instead of paragraph tags inside a `\multirow`.

6688 `\newbool{LWR@in@multirow@par}`
6689 `\boolfalse{LWR@in@multirow@par}`

LWR@starting@fancybox (*bool*) Suppresses `<br>` if beginning a fancybox environment.

6690 `\newbool{LWR@starting@fancybox}`
6691 `\boolfalse{LWR@starting@fancybox}`

LWR@doingstartpars (*bool*) Tells whether paragraphs may be generated.

6692 `\newbool{LWR@doingstartpars}`
6693 `\boolfalse{LWR@doingstartpars}`

LWR@doingapar (*bool*) Tells whether have actually generated and are currently processing paragraph text.

6694 `\newbool{LWR@doingapar}`
6695 `\global\boolfalse{LWR@doingapar}`

LWR@algocf@dopars (*bool*) Tells whether algorithm2e has patched paragraph handling using `\everypar`. If so, the open paragraph tags are generated by algorithm2e's `\algocf@everypar` instead of `\LWR@openparagraph`.

6696 `\newbool{LWR@algocf@dopars}`
6697 `\boolfalse{LWR@algocf@dopars}`

\PN@parnotes@auto Redefined by parnotes to print paragraph notes at the end of each paragraph.

6698 `\def\PN@parnotes@auto{}%`

\LWR@ensuredoingapar These were different in older versions of lwarp, but are now the same thing.
\LWR@openparagraph
6699 `\newcommand*{\LWR@openparagraph}`
6700 `{%`

See if paragraph handling is enabled:

6701 `    \ifboolexpr{`
6702 `        bool{LWR@doingparhooks} and`
6703 `        bool{LWR@doingstartpars}`
6704 `    }%`
6705 `    {% handling pars`

See if have already started a lateximage or a <span>. If so, do not generate nested paragraph tags.

```
6706          \ifboolexpr{
6707              test {\ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}} or
6708              test {\ifnumcomp{\value{LWR@spandepth}}{>}{0}}
6709          }% nested par tags?
```

If so: Do nothing if already started a lateximage page. Cannot nest a lateximage. Also do nothing if already inside a <span>. Do not nest paragraph tags inside a <span>.

```
6710          {}% no nested par tags
```

Else: No lateximage or <span> has been started yet, so it's OK to generate paragraph tags.

```
6711          {% yes nest par tags
6712              \ifbool{LWR@doingapar}{}{%
```

If parnotes is used, paragraph notes are inserted before starting the next paragraph:

```
6713                  \PN@parnotes@auto%
```

Set flag before creating the tag, so that the tag itself does not trigger a new paragraph:

```
6714                  \global\booltrue{LWR@doingapar}%
```

The opening paragraph tag. Do not create tag if doing algorithm2e handling or inside a \multirow.

```
6715                  \ifbool{LWR@algocf@dopars}{}{%
6716                      \ifbool{LWR@in@multirow@par}%
6717                          {}%
6718                      {\LWR@htmltagc{\LWR@tagregularparagraph}\LWR@orignewline}%
6719                  }%
6720              }%
6721          }% end of yes nest par tags
6722      }% end of handling pars
6723      {}% not handling pars
6724 }
6725
6726 \let\LWR@ensuredoingapar\LWR@openparagraph
```

\LWR@closeparagraph@br Add an HTML break if in a span, and not in a lateximage, and not in tabular metadata. Factored from \LWR@closeparagraph.

```
6727 \newcommand*{\LWR@closeparagraph@br}
6728 {%
6729      \ifboolexpr{
6730          test {\ifnumcomp{\value{LWR@spandepth}}{>}{0}} and
6731          test {\ifnumcomp{\value{LWR@lateximagedepth}}{=}{0}} and
6732          not bool {LWR@starting@fancybox} and
6733          not bool {LWR@intabularmetadata} or
6734          bool {LWR@in@multirow@par}
```

```
6735        }%
6736            {\unskip\LWR@htmltagc{br /}}%
6737            {}%
6738 }
```

\LWR@closeparagraph

```
6739 \newcommand*{\LWR@closeparagraph}
6740 {%
6741 % \LWR@traceinfo{LWR@closeparagraph}%
```

See if paragraph handling is enabled:

```
6742    \ifbool{LWR@doingparhooks}{%
6743        \ifbool{LWR@doingapar}%
```

If currently in paragraph mode:

```
6744        {% handling pars
```

See if already started a lateximage or a <span>:

```
6745            \ifboolexpr{
6746                test {\ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}} or
6747                test {\ifnumcomp{\value{LWR@spandepth}}{>}{0}} or
6748                bool{LWR@in@multirow@par}
6749            }%
```

Add a parbreak if in a span, not in a lateximage, and not in table metadata.

```
6750            {% no nested par tags
6751                \LWR@closeparagraph@br%
6752            }% no nested par tags
```

If have not already started a lateximage or a <span>:

```
6753            {% yes nest par tags
```

Print a closing tag.

(The fill seems to be required to force the caption package to create flush left caption text in the HTML.)

```
6754                \@hspacer{\fill}% \hspace*{\fill}
6755                \leavevmode\LWR@orignewline%
6756                \LWR@htmltagc{/\LWR@tagregularparagraph}%
```

No longer doing a paragraph:

```
6757                \global\boolfalse{LWR@doingapar}%
```

Disable the special minipage & \hspace interaction until a new minipage is found:

```
6758                \global\boolfalse{LWR@minipagethispar}%
```

If parnotes is used, paragraph notes are inserted after ending the previous paragraph:

```
6759               \PN@parnotes@auto%

6760           }% end of yes nest par tags
6761       }% LWR@doingapar: end of handling pars
```

Add a parbreak if in a span, not in a lateximage, and not in table metadata.

```
6762       {% not LWR@doingapar: not handling pars
6763           \LWR@closeparagraph@br%
6764       }% not handling pars
```

In most cases, finish with a LaTeX \par, but in the case of paragraphs between lines in a tabular fetch the next token instead. Required for \multicolumn.

```
6765       \ifboolexpr{%
6766           not bool {LWR@doingapar} and
6767           test {\ifnumcomp{\value{LWR@tabulardepth}}{>}{0}} and
6768           test {
6769       \ifnumcomp{\value{LWR@tabulardepth}}{=}{\value{LWR@tabularpardepth}}
6770           } and
6771           bool {LWR@intabularmetadata} and
6772           not bool {LWR@tableparcell} and
6773           test {\ifnumcomp{\value{LWR@lateximagedepth}}{=}{0}}
6774       }%
6775           {\LWR@getmynexttoken}%
6776           {}%
6777     }% LWR@doingparhooks
6778     {}% not LWR@doingparhooks
6779 % Do not place anything here, due to the above \LWR@getmynexttoken.
6780 }
```

55.1   Paragraph Hooks

para/begin (*Hook*) [LaTeX]

```
6781 \AddToHook{para/begin}[lwarp]{\LWR@openparagraph}
```

para/end (*Hook*) [LaTeX]

```
6782 \AddToHook{para/end}[lwarp]{\LWR@closeparagraph}
```

```
6783 \end{warpHTML}
```

56   Paragraph start/stop handling

These commands allow/disallow the generation of HTML paragraph tags.

Section 55 has the commands which actually generate the tags.

The LaTeX paragraph hooks are used to generate the opening and closing paragraph tags.

for HTML output: `6784 \begin{warpHTML}`

\LWR@startpars Begin handling HTML paragraphs. This allows an HTML paragraph to start, but one
has not yet begun.

```
6785 \newcommand*{\LWR@startpars}%
6786 {%
```

Ignore if inside a lateximage or <span>:

```
6787     \ifboolexpr{
6788         test {\ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}} or
6789         test {\ifnumcomp{\value{LWR@spandepth}}{>}{0}}
6790     }%
6791         {}% nesting
6792         {% not nesting
```

The LaTeX paragraph hook controls tag generation for the start and end of para-
graphs.

See if currently handling HTML paragraphs:

```
6793         \ifboolexpr {bool{LWR@doingparhooks} and bool{LWR@doingstartpars}}%
```

If already in paragraph mode, do nothing.

```
6794             {}%
```

If not currently in paragraph mode:

```
6795             {\par}%
```

Are now handling paragraphs, but have not yet actually started one:

```
6796             \global\booltrue{LWR@doingstartpars}%
```

No <par> tag yet to undo:

```
6797             \global\boolfalse{LWR@doingapar}%
6798         }% not nesting
6799 }
```

\LWR@stoppars Stop handling HTML paragraphs. Any currently open HTML paragraph is closed,
and no more will be opened.

```
6800 \newcommand*{\LWR@stoppars}%
6801 {%
```

Ignore if inside a lateximage or <span>:

```
6802     \ifboolexpr{
6803         test {\ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}} or
6804         test {\ifnumcomp{\value{LWR@spandepth}}{>}{0}}
6805     }%
6806         {}% nesting
6807         {% not nesting
```

See if currently handling HTML paragraphs:

```
6808             \ifboolexpr{bool{LWR@doingparhooks} and bool{LWR@doingapar}}%
```

if currently in an HTML paragraph:

```
6809                {%
```

Print a closing tag:

```
6810                \leavevmode\LWR@orignewline%
6811                \LWR@htmltagc{/\LWR@tagregularparagraph}%
6812                \LWR@orignewline%
```

No longer have an open HTML paragraph:

```
6813                \global\boolfalse{LWR@doingapar}%
```

Disable the special minipage & \hspace interaction until a new minipage is found:

```
6814                \global\boolfalse{LWR@minipagethispar}%
6815            }%
```

If was not in an HTML paragraph:

```
6816            {}%
```

No longer in paragraph mode:

```
6817            \global\setbool{LWR@doingstartpars}{false}%
```

No <p> tag to undo:

```
6818            \global\boolfalse{LWR@doingapar}%
6819        }% not nesting
6820 }
```

```
6821 \end{warpHTML}
```

57 Indentfirst

indentfirst (*Pkg*)   indentfirst redefines \@afterindentfalse to be \@afterindenttrue. This is re-
versed \AtBeginDocument here.

for HTML output: 6822 \begin{warpHTML}

```
6823 \AtBeginDocument{
6824    \def\@afterindentfalse{\let\if@afterindent\iffalse}
6825    \@afterindentfalse
6826 }
6827 \let\LWR@afterindent@syntaxhighlight\fi% syntax highlighting
```

```
6828 \end{warpHTML}
```

58 Page headers and footers

6829 \begin{warpall}

In the following, catcode is manually changed back and forth without groups, since new macros are being defined which must not be contained within the groups.

```
6830 \newcommand{\LWR@firstpagetop}{} % for the home page alone
6831 \newcommand{\LWR@firstpagebottom}{} % for the home page alone
6832 \newcommand{\LWR@pagetop}{}    % for all other pages
6833 \newcommand{\LWR@pagebottom}{}
```

```
6834 \newcommand{\LWR@HTMLmeta}{}
```

\HTMLFirstPageTop {⟨*text and logos*⟩}

```
6835 \newcommand{\HTMLFirstPageTop}[1]{%
6836    \renewcommand{\LWR@firstpagetop}{#1}%
6837 }
```

\HTMLFirstPageBottom {⟨*text and logos*⟩}

```
6838 \newcommand{\HTMLFirstPageBottom}[1]{%
6839    \renewcommand{\LWR@firstpagebottom}{#1}%
6840 }
```

\HTMLPageTop {⟨*text and logos*⟩}

```
6841 \newcommand{\HTMLPageTop}[1]{%
6842    \renewcommand{\LWR@pagetop}{#1}%
6843 }
```

\HTMLPageBottom {⟨*text and logos*⟩}

```
6844 \newcommand{\HTMLPageBottom}[1]{%
6845    \renewcommand{\LWR@pagebottom}{#1}%
6846 }
```

\HTMLMeta {⟨*name*⟩} {⟨*content*⟩}

Sets a custom meta tag for the following pages.

```
6847 \newcommand{\HTMLMeta}[2]{%
6848    \renewcommand{\LWR@HTMLmeta}{%
6849       \LWR@htmltag{%
6850        meta name=\LWR@orig@textquotedbl{}#1\LWR@orig@textquotedbl\ % space
6851           content=\LWR@orig@textquotedbl{}#2\LWR@orig@textquotedbl\ /%
6852       }\LWR@orignewline%
6853    }{}{}%
6854 }
```

\HTMLAddMeta {⟨*name*⟩} {⟨*content*⟩}

Adds to the custom meta tags for the following pages.

```
6855 \newcommand{\HTMLAddMeta}[2]{%
6856     \apptocmd{\LWR@HTMLmeta}{%
6857         \LWR@htmltag{%
6858          meta name=\LWR@orig@textquotedbl{}#1\LWR@orig@textquotedbl\ % space
6859             content=\LWR@orig@textquotedbl{}#2\LWR@orig@textquotedbl\ /%
6860         }\LWR@orignewline%
6861     }{}{}%
6862 }
```

```
6863 \end{warpall}
```

59   css

`6864 \begin{warpHTML}`

\LWR@currentcss   The css filename to use. This may be changed mid-document using \CSSFilename, allowing different css files to be used for different sections of the document.

`6865 \newcommand*{\LWR@currentcss}{lwarp.css}`

\CSSFilename   {⟨*new-css-filename.css*⟩}   Assigns the css file to be used by the following HTML pages.

```
6866 \newcommand*{\CSSFilename}[1]{%
6867     \renewcommand*{\LWR@currentcss}{#1}%
6868     \@onelevel@sanitize\LWR@currentcss%
6869 }
6870
6871 \end{warpHTML}
```

`6872 \begin{warpprint}`
```
6873 \newcommand*{\CSSFilename}[1]{}
6874 \end{warpprint}
```

60   MATHJAX script

`6875 \begin{warpHTML}`
Default: lwarp_mathjax.txt

\LWR@mathjaxfilename   The MATHJAX script filename to use. This file is copied into the head of each HTML page. This may be changed mid-document using \MathJaxFilename, allowing the use of a custom MATHJAX script, such as for a local repository, or different MATHJAX script files to be used for different sections of the document.

`6876 \newcommand*{\LWR@mathjaxfilename}{lwarp_mathjax.txt}`

\MathJaxFilename   {⟨*filename*⟩}   Assigns the MATHJAX script file to be used by the following HTML pages.

```
6877 \newcommand*{\MathJaxFilename}[1]{%
6878     \renewcommand*{\LWR@mathjaxfilename}{#1}%
6879     \@onelevel@sanitize\LWR@mathjaxfilename%
6880 }
6881
6882 \end{warpHTML}
```

for PRINT output:
```
6883 \begin{warpprint}
6884 \newcommand*{\MathJaxFilename}[1]{}
6885 \end{warpprint}
```

61   Title, HTML meta author, HTML meta description

for HTML output:
```
6886 \begin{warpHTML}
```

\title {⟨*title*⟩} Modified to remember \thetitle, which is used to set the HTML page titles.

```
6887 \let\LWR@origtitle\title
6888
6889 \renewcommand*{\title}[1]{%
6890     \LWR@origtitle{#1}%
6891     \begingroup%
6892         \renewcommand{\thanks}[1]{}%
6893         \protected@xdef\thetitle{#1}%
6894     \endgroup%
6895 }
6896 \end{warpHTML}
```

for HTML & PRINT:
```
6897 \begin{warpall}
```

\HTMLTitle {⟨*Titlename*⟩}   The Title to place into an HTML meta tag. The default is to use the document \title's setting.

```
6898 \providecommand{\thetitle}{\BaseJobname}
6899
6900 \newcommand{\theHTMLTitle}{\thetitle}
6901
6902 \newcommand{\HTMLTitle}[1]{\renewcommand{\theHTMLTitle}{#1}}
```

\HTMLAuthor {⟨*authorname*⟩}   The author to place into an HTML meta tag. If none given, the default is \theauthor, which is empty unless the titling package is used.

```
6903 \providecommand{\theauthor}{}
6904
6905 \newcommand{\theHTMLAuthor}{\theauthor}
6906
6907 \newcommand{\HTMLAuthor}[1]{\renewcommand{\theHTMLAuthor}{#1}}
```

This is placed inside an HTML meta tag at the start of each file. This may be changed mid-document using \HTMLAuthor, allowing different HTML authors to be used for different sections of the document.

⚠ HTML author Do not use double quotes, and do not exceed 150 characters.

\HTMLDescription {⟨*New html meta description.*⟩}   Assigns the HTML file's description meta tag.

```
6908 \newcommand{\LWR@currentHTMLDescription}{}
6909
6910 \newcommand{\HTMLDescription}[1]{%
6911 \renewcommand{\LWR@currentHTMLDescription}{#1}
6912 }
```

\HTMLKeywords {⟨*New html meta keywords.*⟩}   Assigns the HTML file's keywords meta tag.

```
6913 \newcommand{\LWR@currentHTMLKeywords}{}
6914
6915 \newcommand{\HTMLKeywords}[1]{%
6916 \renewcommand{\LWR@currentHTMLKeywords}{#1}
6917 }
6918
6919 \end{warpall}
```

62 Footnotes

lwarp uses native LaTeX footnote code, although with its own \box to avoid the LaTeX output routine. The usual functions mostly work as-is.

footnote numbering   To have footnote numbers reset each time footnotes are printed:

> \setcounter{footnoteReset}{1}

For bigfoot, manyfoot, or perpage:

> \MakePerPage{footnoteX}
> — or —
> \MakeSortedPerPage{footnoteX}

The footnotes are reset when they are printed, according to section level as set by FootnoteDepth, which is not necessarily by HTML page. This is recommended for \alph, \Alph, or \fnsymbol footnotes, due to the limited number of symbols which are available.

MATHJAX   Also for MATHJAX, \footnotename is used for a \footnotemark if the actual footnote number is not known. To redefine it, provide it before loading lwarp:

> \providecommand{\footnotename}{something}
> \usepackage{lwarp}

Similar for sidenotes. For endnotes:

> \def\endnotename{something}% \def allows name to start with
> "end"

For the pagenote package, there is no \pagenotename to define, since there is no \pagenotemark command.

footmisc   The footmisc stable option is emulated by lwarp.

⚠ sectioning commands   When using footnotes in sectioning commands, to generate consistent results between print and HTML, use the footmisc package with the stable option, provide a short TOC entry, and \protect the \footnote:

```
\usepackage[stable]{footmisc}
. . .
\subsection[Subsection Name]
    {Subsection Name\protect\footnote{A footnote.}}
```

memoir with footmisc ⚠ memoir

If using **memoir** class, with which **lwarp** preloads **footmisc**, the stable option must be declared before **lwarp** is loaded:

```
\PassOptionsToPackage{stable}{footmisc}
\usepackage{lwarp}
. . .
```

Do not use a starred sectioning command. As an alternative, it may be possible to adjust \secnumdepth instead.

Several kinds of footnotes are used: in a regular page, in a minipage, or as thanks in the titlepage. Each of these is handle differently.

62.1   Regular page footnotes

In HTML documents, footnotes are placed at the bottom of the web page or the section, depending on FootnoteDepth, using the LaTeX box \LWR@footnotebox. Using this instead of the original \footins box avoids having footnotes be printed by the output routine, since footnotes should be printed per HTML page instead of per PDF page.

See section 62.4 for the implementation.

62.2   Minipage footnotes

See section 62.5 for how minipage footnotes are gathered. See section 96.4 for how minipage footnotes are placed into the document.

62.3   Titlepage thanks

See section 71.7 for titlepage footnotes.

62.4   Regular page footnote implementation

for HTML & PRINT: 6920 `\begin{warpall}`

FootnoteDepth (*Ctr*)   Determines how deeply to place footnotes in the HTML files, similar to tocdepth.
Default: 3   The default of 3 places footnotes before each \subsubsection or higher. See table 12 for a table of LaTeX section headings.

6921 `\newcounter{FootnoteDepth}`
6922 `\setcounter{FootnoteDepth}{3}`

footnoteReset (*Ctr*)   If non-zero, the footnote counter is reset to this value each time the footnotes
Default: 0   are printed, as controlled by FootnoteDepth. For the **manyfoot** and **bigfoot** packages, additional counters such as footnote<suffix>Reset will be defined

as well. These counters may be set non-zero by the user, and are also set if the **perpage**'s \MakePerPage or \MakeSortedPerPage macros are used for the footnote or footnote<suffix> counters.

(The name is not capitalized because it is made from the counter's name with "Reset" appended.)

```
6923 \newcounter{footnoteReset}
6924 \setcounter{footnoteReset}{0}
```

```
6925 \end{warpall}
```

6926 \begin{warpHTML}

Required for footnotes inside description or **amstheorem** square braces:

```
6927 \AtBeginDocument{
6928 \robustify{\footnote}
6929 \robustify{\footnotemark}
6930 }
```

\LWR@footnotebox Patch LATEX footnotes to use a new \box instead of an insert for **lwarp** footnotes. This avoids having the original \footins appear at the bottom of a lateximage, which is on its own new page.

```
6931 \newbox\LWR@footnotebox
```

LWR@spewingnotes (*bool*) Used with the **footnote** package to suppress paragraph tags before and after \spewnotes.

```
6932 \newbool{LWR@spewingnotes}% For the footnote package.
```

Much of the following has unneeded print-mode formatting removed.

\@makefntext {⟨*text*⟩}

```
6933 \long\def\@makefntext#1{\textsuperscript{\@thefnmark}~{#1}}
```

\@makefnmark

```
6934 \def\@makefnmark{%
6935     \textsuperscript{\@thefnmark}%
6936 }
```

Footnotes may be in regular text, in which case paragraphs are tagged, or in a table data cell or lateximage, in which case paragraph tags must be added manually.

In a lateximage during HTML output, the lateximage is placed inside a print-mode minipage, but the footnotes are broken out by:

```
\def\@mpfn{footnote}
\def\thempfn{\thefootnote}
\let\@footnotetext\LWR@footnotetext
```

\LWR@@footnotetext {⟨*text*⟩} {⟨*footnote box name*⟩}

Factored to allow multiple footnote boxes for manyfoot.

```
6937 \long\def\LWR@@footnotetext#1#2{%
6938 \LWR@traceinfo{LWR@footnotetext}%
```

Perhaps generate an autopage in the text to link a citation backreference closer to its usage.

```
6939 \LWR@newautopagelabel{page}%
6940 \LWR@ensuredoingapar%
```

Locally disable auto page labels inside the footnote text. Footnotes are accumulated in the current page before finally being placed in a potentially later page, so the autopages would be incorrect.

```
6941 \begingroup%
6942 \let\LWR@newautopagelabel\LWR@null@newautopagelabel%
```

Take the existing footnote box and add the new content:

```
6943 \global\setbox\csname #2\endcsname=\vbox{%
6944     \unvbox\csname #2\endcsname%
```

Remember the footnote number for \ref:

```
6945     \def\@currentcounter{footnote}%
6946     \protected@edef\@currentlabel{%
6947         \csname p@footnote\endcsname\@thefnmark%
6948     }% @currentlabel
```

Open a group:

```
6949     \color@begingroup%
```

Disable CJK xpinyin while generating footnotes.

```
6950     \LWR@disablepinyin%
```

Use HTML superscripts in the footnote even when the main text is inside a lateximage, because the footnote will be in HTML:

```
6951     \renewrobustcmd{\textsuperscript}[1]{\LWR@htmlspan{sup}{##1}}%
```

Use paragraph tags if in a tabular data cell or a lateximage:

```
6952     \ifbool{LWR@spewingnotes}{}{%
6953         \LWR@htmltagc{\LWR@tagregularparagraph}\LWR@orignewline%
6954     }%
```

Append the footnote to the list:

```
6955     \@makefntext{#1}%
```

Closing paragraph tag:

```
6956    \ifbool{LWR@spewingnotes}{}{%
6957        \LWR@orignobreakspace\LWR@orignewline%
6958        \LWR@htmltagc{/\LWR@tagregularparagraph}%
6959        \LWR@orignewline%
6960    }%
```

Close the group:

```
6961    \color@endgroup%
6962 }% vbox
6963 \endgroup%
6964 }%
```

\LWR@footnotetext {⟨*text*⟩}

```
6965 \long\def\LWR@footnotetext#1{\LWR@@footnotetext{#1}{LWR@footnotebox}}%
```

\@footnotetext {⟨*text*⟩}

```
6966 \LetLtxMacro\@footnotetext\LWR@footnotetext
```

62.5   Minipage footnote implementation

Patch LATEX minipage footnotes to use a new \box instead of an insert for lwarp
minipage footnotes. This avoids having the original \@mpfootins appear at the
bottom of a lateximage, which is on its own new page.

```
6967 \newbox\LWR@mpfootnotes
```

\@mpfootnotetext {⟨*text*⟩}

```
6968 \long\def\@mpfootnotetext#1{%
6969 \LWR@traceinfo{@mpfootnotetext}%
6970 \LWR@ensuredoingapar%
6971 \global\setbox\LWR@mpfootnotes\vbox{%
6972    \unvbox\LWR@mpfootnotes%
6973    \reset@font\footnotesize%
6974    \hsize\columnwidth%
6975    \@parboxrestore%

6976    \def\@currentcounter{mpfootnote}%

6977    \protected@edef\@currentlabel%
6978        {\csname p@mpfootnote\endcsname\@thefnmark}%
6979    \color@begingroup%
```

Add paragraph tag:

```
6980    \LWR@htmltagc{\LWR@tagregularparagraph}\LWR@orignewline%

6981    \@makefntext{%
6982        \ignorespaces#1%
6983    }%
```

Add the closing paragraph tag:

```
6984    \leavevmode\LWR@orignewline%
6985    \LWR@htmltagc{/\LWR@tagregularparagraph}%

6986    \color@endgroup%
6987 }% vbox
```

Paragraph handling:

```
6988 \LWR@ensuredoingapar%
6989 \LWR@traceinfo{@mpfootnotetext: done}%
6990 }
```

\thempfootnote Redefined to remove the \itshape, which caused an obscure compiling error in some situations.

```
6991 \AtBeginDocument{
6992    \def\thempfootnote{\@alph\c@mpfootnote}
6993 }
```

62.6   Printing pending footnotes

\LWR@@printpendingfootnotes {⟨*footnote counter name*⟩}

```
6994 \newcommand*{\LWR@@printpendingfootnotes}[1]{%
6995 \expandafter\ifvoid\csname LWR@#1box\endcsname\else
6996    \LWR@forcenewpage
6997    \begin{BlockClass}(note){footnotes}%
```

Create a new autopage in case citation back references occur inside the footnotes:

```
6998    \LWR@newautopagelabel{page}%

6999    \null
7000    \unvbox\csuse{LWR@#1box}
7001    \setbox\csuse{LWR@#1box}=\vbox{}
7002    \end{BlockClass}
7003    \ifltxcounter{#1Reset}{%
7004        \ifnumgreater{\value{#1Reset}}{0}{%
7005            \setcounter{#1}{\value{#1Reset}}%
7006            \addtocounter{#1}{-1}%
7007        }{}%
7008    }{}%
7009 \fi
7010 }
```

\LWR@printpendingfootnotes Enclose the footnotes in a class, print, then clear. For manynotes, new footnotes may be added via \appto.

```
7011 \newcommand*{\LWR@printpendingfootnotes}{%
7012    \LWR@@printpendingfootnotes{footnote}%
7013 }
```

\LWR@maybeprintpendingfootnotes {⟨*depth*⟩} Used to print footnotes before sections only if formatting for an EPUB or word processor:

```
7014 \newcommand*{\LWR@maybeprintpendingfootnotes}[1]{%
7015 \ifboolexpr{
7016     not test{\ifnumcomp{#1}{>}{\value{FootnoteDepth}}} or
7017     bool{FormatEPUB} or
7018     bool{FormatWP}
7019 }%
7020 {\LWR@printpendingfootnotes}%
7021 {}%
7022 }
```

\LWR@printpendingmpfootnotes Enclose the minipage footnotes in a class, print, then clear.

```
7023 \newcommand*{\LWR@printpendingmpfootnotes}{%
7024 \ifvoid\LWR@mpfootnotes\else
7025     \LWR@forcenewpage
7026     \begin{BlockClass}(note){footnotes}%
7027     \null
7028     \unvbox\LWR@mpfootnotes
7029     \setbox\LWR@mpfootnotes=\vbox{}
7030     \end{BlockClass}
7031 \fi
7032 }
```

\LWR@nullifyfootnotes Cancels footnotes, such as inside an HTML comment or a \nameref.

```
7033 \newcommand*{\LWR@nullifyfootnotes}{%
7034     \renewcommand{\footnote}[2][]{}%
7035     \renewcommand{\footnotemark}[1][]{}%
7036 }
```

```
7037 \end{warpHTML}
```

63 Marginpars

\marginpar [⟨*left*⟩] {⟨*right*⟩}   \marginpar may contains paragraphs, but in order to remain inline with the surrounding text lwarp nullifies block-related macros inside the \marginpar. Paragraph breaks are converted to <br /> tags.

\marginparBlock [⟨*left*⟩] {⟨*right*⟩}   To include block-related macros, use \marginparBlock, which takes the same arguments but creates a <div> instead of a <span>. A line break will occur in the text where the \marginBlock occurs.

for HTML output: 7038 \begin{warpHTML}

\marginpar [⟨*left*⟩] {⟨*right*⟩}

```
7039 \renewcommand{\marginpar}[2][]{%
7040 \ifbool{FormatWP}%
7041 {%
7042     \begin{LWR@BlockClassWP}%
7043         {width:2in; float:right; margin:10pt}{}(note){marginblock}%
```

```
7044     #2
7045     \end{LWR@BlockClassWP}%
7046 }%
7047 {%
7048     \LWR@htmlspanclass(note){marginpar}{#2}%
7049 }%
7050 }
```

\marginparBlock [⟨*left*⟩] {⟨*right*⟩}

> For use when the marginpar will be more than one paragraph, and/or contains more than simple text.
>
> HTML version.

```
7051 \newcommand{\marginparBlock}[2][]{%
7052     \LWR@stoppars%
7053     \ifbool{FormatWP}%
7054     {%
7055         \begin{LWR@BlockClassWP}%
7056             {width:2in; float:right; margin:10pt}{}%
7057             (note){marginblock}%
7058         #2
7059         \end{LWR@BlockClassWP}
7060     }%
7061     {%
7062         \begin{BlockClass}[width:2in; float:right; margin:10pt]%
7063             (note){marginparblock}%
7064         #2
7065         \end{BlockClass}
7066     }%
7067     \LWR@startpars%
7068 }
```

\reversemarginpar

```
7069 \renewcommand*{\reversemarginpar}{}
```

\normalmarginpar

```
7070 \renewcommand*{\normalmarginpar}{}
```

```
7071 \end{warpHTML}
```

for PRINT output: `7072 \begin{warpprint}`

\marginparBlock [⟨*left*⟩] {⟨*right*⟩}

> For use when the marginpar will be more than one paragraph, and/or contains more than simple text.
>
> Print version.

```
7073 \LetLtxMacro\marginparBlock\marginpar
```

```
7074 \end{warpprint}
```

64 Tracking internal cross references

Cross references are generated using the PDF file's page number during LaTeX compilation. Internal labels are generated which include these page numbers in the label.

*_html.aux (*file*) A new entry in the *_html.aux file is used to help cross-references:

> \newlabel{autopage-<nnn>}{{<x>}{<y>}}

LWR@currentautosecpage (*Ctr*) Records the page number when the section was created. (If a math expression is included in the section name, and SVG math is used, the corresponding lateximage will cause the page number to change by the time the following autosec label is created, thus the initial page number is recorded here.) LWR@currentautosecfloatpage is updated more often than LWR@currentautosecpage.

```
7075 \newcounter{LWR@currentautosecpage}
7076 \setcounter{LWR@currentautosecpage}{1}
```

LWR@currentautosecfloatpage (*Ctr*) The HTML output's PDF page number at the start of a new HTML file, section, or float. Updated more often than LWR@currentautosecpage, such as when a new float occurs. Used only for table of contents, list of figures, list of tables, but not for general cross references such as \label, citation backlinks, etc.

\LWRsetnextfloat is written with this and the autoid by the modified \addcontentsline just before each float's entry.

```
7077 \newcounter{LWR@currentautosecfloatpage}
7078 \setcounter{LWR@currentautosecfloatpage}{1}
```

LWR@previousautopagelabel (*Ctr*) Remembers which autopage label was most recently generated. Used to avoid duplicates.

```
7079 \newcounter{LWR@previousautopagelabel}
7080 \setcounter{LWR@previousautopagelabel}{-1}
```

\LWR@newautopagelabel {⟨*pagenumber counter*⟩}

\BaseJobname is added to the label in case xr or xr-hyper are used.

```
7081 \newcommand*{\LWR@newautopagelabel}[1]{%
```

No action if this autopage label has already been defined:

```
7082 \ifnumequal{\value{LWR@previousautopagelabel}}{\value{page}}%
7083     {}%
```

If the PDF page has changed, create a label using the desired counter.

If the counter is LWR@currentautosecpage, that was the page number when the section generation began, but the current PDF page may be different by now if the section name had an SVG image, such as SVG math. To allow the cross-reference to point just after the section heading, the label must be made after the section heading is complete, which may have generated a new PDF page. Thus, the label

is made with the given counter, which may be the PDF page number where the section heading began, then if the PDF page number has changed, another label is made for the current page number.

```
7084    {%
7085        \label{\BaseJobname-autopage-\csuse{the#1}}%
```

If there are intervening pages, such as an SVG image, define another label for the new page:

```
7086        \ifnumequal{\value{#1}}{\value{page}}%
7087            {}%
7088            {\label{\BaseJobname-autopage-\csuse{thepage}}}%
```

Remember the latest autopage label:

```
7089        \setcounter{LWR@previousautopagelabel}{\value{page}}%
7090    }%
7091 }
```

\LWR@null@newautopagelabel {⟨*pagenumber counter*⟩}

Inside a footnote, the page numbers will be incorrect, so this is nullified.

```
7092 \newcommand*{\LWR@null@newautopagelabel}[1]{}
```

65   Splitting HTML files

- Files are split according to FileDepth and CombineHigherDepths.

- Filenames are sanitized by \LWR@filenamenoblanks.

- \LWR@newhtmlfile finishes an HTML page, adds a comment to tell where and how to split the file, then starts a new HTML page.

for HTML & PRINT: ```7093 \begin{warpall}```

FileDepth (*Ctr*)   {⟨*section depth*⟩} determines how deeply to break into new HTML files, similar to tocdepth. The default of -5 produces one large HTML file.

```
7094 \newcounter{FileDepth}
7095 \setcounter{FileDepth}{-5}
```

CombineHigherDepths (*bool*)   Combile higher-level sections together into one file?

```
7096 \newbool{CombineHigherDepths}
7097 \booltrue{CombineHigherDepths}
```

\FilenameLimit   Maximum length of the generated filenames.

```
7098 \newcommand*{\FilenameLimit}{80}
```

```
7099 \end{warpall}
```

for HTML output: 7100 `\begin{warpHTML}`

`\LWR@thisfilename` The currently-active filename or number. At first, this is the homepage.

```
7101 \AtBeginDocument{
7102 \ifbool{FileSectionNames}%
7103    {\newcommand*{\LWR@thisfilename}{\HomeHTMLFilename}}
7104    {\newcommand*{\LWR@thisfilename}{0}}
7105 }
```

`\LWR@thisnewfilename` The filename being sanitized.

```
7106 \newcommand*{\LWR@thisnewfilename}{}
```

`\LWR@simplifyname` * {⟨*expression*⟩}   Simplify `\LWR@thisnewfilename`.

If starred, detokenizes the input expression. If found, changes the expression to a single detokenized dash.

```
7107 \NewDocumentCommand{\LWR@simplifyname}{s m}{%
7108 \IfBooleanTF{#1}{%
7109    \StrSubstitute{\LWR@thisnewfilename}%
7110       {\detokenize{#2}}%
7111       {\detokenize{-}}[\LWR@thisnewfilename]%
7112 }{%
7113    \StrSubstitute{\LWR@thisnewfilename}%
7114       {#2}%
7115       {\detokenize{-}}[\LWR@thisnewfilename]%
7116 }
7117 }
```

`\LWR@simplifycustom` User-defined filename simplifications. Redefine with `\newcommand`.

```
7118 \newcommand*{\LWR@simplifycustom}{}
```

`\FilenameSimplify` * {⟨*phrase*⟩} Assign a user-defined filename simplification. Appends to `\LWR@simplifycustom`.

```
7119 \NewDocumentCommand{\FilenameSimplify}{s m}{%
7120 \IfBooleanTF{#1}{%
7121    \appto{\LWR@simplifycustom}{%
7122       \LWR@simplifyname*{#2}%
7123    }%
7124 }{%
7125    \appto{\LWR@simplifycustom}{%
7126       \LWR@simplifyname{#2}%
7127    }%
7128 }%
7129 }
```

`\LWR@avoiddupfilenames`   Instructions for how to avoid duplicate filenames. This is used in a warning in `\LWR@filenamenoblanks`, and in an error in `\LWR@newhtmlfile`.

```
7130 \newcommand*{\LWR@avoiddupfilenames}{%
7131    To avoid duplicate filenames, use the optional\MessageBreak
```

```
7132    short Table of Contents entry:\MessageBreak
7133    \space\space\protect\section[Unique name, no math]{Name with math}%
7134        \MessageBreak
7135    or use \protect\texorpdfstring, from the hyperref package:\MessageBreak
7136    \space\space%
7137        \protect\section{\MessageBreak
7138            \space\space\space\space\protect\texorpdfstring\MessageBreak
7139                \space\space\space\space\space\space%
7140                {Name with math}{Unique name, no math}\MessageBreak
7141        \space\space}
7142 }
```

\LWR@filenamenoblanks {⟨*filename*⟩}

> Convert blanks into dashes, removes short words, store result in
> \LWR@thisfilename.

> Also see \LWR@nullfonts for nullified macros.

```
7143 \newcommand*{\LWR@filenamenoblanks}[1]{%
7144 \begingroup
```

> Locally temporarily disable direct-formatting commands, not used in filenames:

```
7145 \LWR@nullfonts%
7146 \renewcommand*{\LWR@htmltagc}[1]{}%
```

```
7147 \edef\LWR@thisnewfilename{#1}%
```

> Replaces common macros with hyphens. (\& is done by \LWR@nullfonts.)

```
7148 \RenewDocumentCommand{\LWR@subsingledollar}{s m m m}{}%
7149 \LWR@simplifyname{\_}
7150 \LWR@simplifyname{\#}
7151 \LWR@simplifyname{\textbackslash}
7152 \LWR@simplifyname{\protect}
7153 \LWR@simplifyname{\ }
7154 \LWR@simplifyname{\textless}
7155 \LWR@simplifyname{\textgreater}
```

```
7156 \edef\LWR@thisnewfilename{\detokenize\expandafter{\LWR@thisnewfilename}}%
```

> Warn if there is dollar math in the section name:

```
7157 \ifbool{FileSectionNames}{%
7158     \IfSubStr{\LWR@thisnewfilename}{\LWRdollar}{%
7159         \PackageWarning{lwarp}
7160         {%
7161             This section name:\MessageBreak
7162             \space\space``\detokenize\expandafter{#1}''\MessageBreak
7163             at the line number listed below,\MessageBreak
7164             is using $dollar-delimited math$,
7165             which generates\MessageBreak
7166             complicated file names.  It is better to use\MessageBreak
7167             \space\space%
7168             \protect\section{Name with \protect\(parenthesis math\protect\)}%
7169                 \MessageBreak
```

```
7170          The math then will be removed from the file name.\MessageBreak
7171          \MessageBreak
7172          \LWR@avoiddupfilenames%
7173          \MessageBreak
7174          This section is found before or%
7175      }
7176   }{}%
7177 }{}
```

```
7178 \LWR@traceinfo{LWR@filenamenoblanks edef: !\LWR@thisnewfilename!}%
7179 \fullexpandarg%
```

Convert spaces into hyphens:

```
7180 \LWR@simplifyname*{ }
```

Convert punctutation into hyphens:

```
7181 \LWR@simplifyname*{*}
7182 \LWR@simplifyname*{(}
7183 \LWR@simplifyname*{)}
7184 \LWR@simplifyname*{.}
7185 \LWR@simplifyname*{!}
7186 \LWR@simplifyname*{,}
7187 \LWR@simplifyname*{'}
7188 \LWR@simplifyname*{+}
7189 \LWR@simplifyname*{/}
7190 \LWR@simplifyname*{:}
7191 \LWR@simplifyname*{;}
7192 \LWR@simplifyname*{=}
7193 \LWR@simplifyname*{?}
7194 \LWR@simplifyname*{@}
7195 \LWR@simplifyname*{^}
7196 \LWR@simplifyname*{&}
7197 \LWR@simplifyname*{"}
7198 \LWR@simplifyname*{<}
7199 \LWR@simplifyname*{>}
```

```
7200 \LWR@simplifyname{\LWRbackslash}
```

Braces are removed entirely to avoid extra dashes in the result.

```
7201 \StrSubstitute{\LWR@thisnewfilename}%
7202     {\LWRleftbrace}{}[\LWR@thisnewfilename]%
7203 \StrSubstitute{\LWR@thisnewfilename}%
7204     {\LWRrightbrace}{}[\LWR@thisnewfilename]%
```

```
7205 \LWR@simplifyname{\LWRpercent}
7206 \LWR@simplifyname{\LWRdollar}
```

```
7207 \LWR@simplifyname*{|}
7208 \LWR@simplifyname*{^}
7209 \LWR@simplifyname*{~}
7210 \LWR@simplifyname*{[}
7211 \LWR@simplifyname*{]}
7212 \LWR@simplifyname*{`}
```

Convert short words:

```
7213 \LWR@simplifyname*{-s-}
7214 \LWR@simplifyname*{-S-}
7215 \LWR@simplifyname*{-a-}
7216 \LWR@simplifyname*{-A-}
7217 \LWR@simplifyname*{-an-}
7218 \LWR@simplifyname*{-AN-}
7219 \LWR@simplifyname*{-to-}
7220 \LWR@simplifyname*{-TO-}
7221 \LWR@simplifyname*{-by-}
7222 \LWR@simplifyname*{-BY-}
7223 \LWR@simplifyname*{-of-}
7224 \LWR@simplifyname*{-OF-}
7225 \LWR@simplifyname*{-and-}
7226 \LWR@simplifyname*{-AND-}
7227 \LWR@simplifyname*{-for-}
7228 \LWR@simplifyname*{-FOR-}
7229 \LWR@simplifyname*{-the-}
7230 \LWR@simplifyname*{-THE-}
```

Convert custom words:

```
7231 \LWR@simplifycustom%
```

If PDF LATEX and not `utf8` encoding, don't try to convert emdash, endash:

```
7232 \ifPDFTeX% pdflatex or dvi latex
7233 \ifdefstring{\inputencodingname}{utf8}{%
7234     \LWR@simplifyname*{—}
7235 %     emdash
7236     \LWR@simplifyname*{–}
7237 %     endash
7238 }{}%
7239 \else% not PDFTeX
7240     \LWR@simplifyname*{—}
7241     \LWR@simplifyname*{–}
7242 \fi%
```

Convert multiple hyphens:

```
7243 \LWR@simplifyname*{-----}
7244 \LWR@simplifyname*{----}
7245 \LWR@simplifyname*{---}
7246 \LWR@simplifyname*{--}
```

If starts with a dash, remove the leading dash:

```
7247 \IfBeginWith{\LWR@thisnewfilename}{\detokenize{-}}{%
7248     \StrGobbleLeft{\LWR@thisnewfilename}{1}[\LWR@thisnewfilename]%
7249 }{}%
```

If ends with a dash, remove the trailing dash:

```
7250 \IfEndWith{\LWR@thisnewfilename}{\detokenize{-}}{%
7251     \StrGobbleRight{\LWR@thisnewfilename}{1}[\LWR@thisnewfilename]%
7252 }{}%
```

Limits the length of the filename:

7253 \StrLeft{\LWR@thisnewfilename}{\FilenameLimit}[\LWR@thisnewfilename]%

Return the global result:

7254 \global\let\LWR@thisfilename\LWR@thisnewfilename%
7255 \endgroup%
7256 \LWR@traceinfo{LWR@filenameoblanks: result is \LWR@thisfilename}%
7257 }

65.1   Sanitizing expressions for HTML

Math expressions are converted to lateximages, and some math environments
may contain &, <, or >, which should not be allowed inside an HTML <alt> tag, so
must convert them to HTML entities.

\LWR@replacestrings {⟨*search*⟩} {⟨*replace*⟩}

Replaces strings inside \tmpb.

Modified from the original, by PETR OLSAK, from the opmac package.

7258 \bgroup
7259 \catcode`!=3 \catcode`?=3
7260
7261 \long\gdef\LWR@replacestrings@addto#1#2{%
7262   \expandafter\def\expandafter#1\expandafter{#1#2}%
7263 }
7264
7265 \gdef\LWR@replacestrings#1#2{%
7266   \long\def\LWR@replacestringsA##1#1{\def\tmpb{##1}\LWR@replacestringsB}%
7267   \long\def\LWR@replacestringsB##1#1{%
7268   \ifx!##1\relax \else\LWR@replacestrings@addto\tmpb{#2##1}%
7269   \expandafter\LWR@replacestringsB\fi%
7270   }%   improved version <May 2016> inspired
7271   \expandafter\LWR@replacestringsA\tmpb?#1!#1% from pysyntax.tex by Petr Krajnik
7272   \long\def\LWR@replacestringsA##1?{%
7273   \def\tmpb{##1}%
7274   }\expandafter\LWR@replacestringsA\tmpb%
7275 }
7276 \egroup

LWR@HTMLsanitize@tmpb@enable   Allow to disable sanitization while inside a lateximage, or while using minted.
(*bool*)   HTML sanitizaiton was occurring too early, and minted would then colorize the
⚠   sanitized results, breaking the HTML entities in lwarp's HTML output.

7277 \newbool{LWR@HTMLsanitize@tmpb@enable}
7278 \booltrue{LWR@HTMLsanitize@tmpb@enable}

LWR@HTMLsanitize@tmpb@removebackslash
(*bool*)
    Allow to enable / disable sanitization of the macros \%, \#, \%. This is usually
    enabled to allow the user to enter these macros in URLs, for example, but is disabled

for ALT tags and MATHJAX output where the literal LATEX source must be preserved.

```
7279 \newbool{LWR@HTMLsanitize@tmpb@removebackslash}
7280 \booltrue{LWR@HTMLsanitize@tmpb@removebackslash}
```

LWR@MathJax@silentquotes   If true, double quotes (\" and ") are removed (used for mathspec). This unfortu-
            (*bool*)   nately includes double quotes used inside \text with MATHJAX. If false, double
                       quotes are escaped.

```
7281 \newbool{LWR@MathJax@silentquotes}
7282 \boolfalse{LWR@MathJax@silentquotes}
```

\LWR@eolspace   The end of line character generated by detokenizing \<space> when at the end of
                a line.

```
7283 \expandafter\def\expandafter\LWR@eolspace\expandafter{\detokenize{\
7284 }}% there is a trailing space in the previous line
```

\LWR@HTMLsanitize@tmpb   Sanitizes HTML for \tmpb. These characters may be interpreted by the browser,
                         causing false HTML code.

To allow sanitization, expand anything \detokenized before assigning to \tmpb.

```
7285 \catcode`\&=12
7286 \newcommand*{\LWR@HTMLsanitize@tmpb}{%
7287     \ifbool{LWR@HTMLsanitize@tmpb@enable}%
7288     {%
```

HTML entities:

At the end of a line, \<space> is turned in to an end of line character, which is
now converted to a regular \<space>.

```
7289         \expandafter\LWR@replacestrings\expandafter{\LWR@eolspace}{\ }%
```

&amp must be first because & is used for lt, gt, etc.

```
7290         \LWR@replacestrings{&}{&amp;}% Must be before the following:
7291         \LWR@replacestrings{<}{&lt;}%
7292         \LWR@replacestrings{>}{&gt;}%
7293         \LWR@replacestrings{'}{&apos;}%
7294         \LWR@replacestrings{`}{&grave;}%
7295         \LWR@replacestrings{_}{\detokenize{_}}%
```

Neutralize \%, \#, \& in case used by the author.

\@tempa is built to be

```
\LWR@replacestrings{\#}{#}
```

---

and similar for % and &.

```
7296         \ifbool{LWR@HTMLsanitize@tmpb@removebackslash}%
```

```
7297          {%
7298              \def\@tempa{\LWR@replacestrings}%
7299          \expandafter\appto\expandafter\@tempa\expandafter{\expandafter{\detokenize{\#}}}%
7300          \expandafter\appto\expandafter\@tempa\expandafter{\expandafter{\LWRhash}}%
7301              \@tempa%
7302              \def\@tempa{\LWR@replacestrings}%
7303          \expandafter\appto\expandafter\@tempa\expandafter{\expandafter{\detokenize{\%}}}%
7304          \expandafter\appto\expandafter\@tempa\expandafter{\expandafter{\LWRpercent}}%
7305              \@tempa%
7306              \def\@tempa{\LWR@replacestrings}%
7307          \expandafter\appto\expandafter\@tempa\expandafter{\expandafter{\detokenize{\&}}}%
7308          \expandafter\appto\expandafter\@tempa\expandafter{\expandafter{\LWRamp}}%
7309              \@tempa%
7310          }%
7311          {}%
```

The quotes occasionally causes problems. For MathJax, also allow neutralization
of \" and the " character.

```
7312              \ifbool{LWR@MathJax@silentquotes}
7313                  {%
7314                      \expandafter\LWR@replacestrings\expandafter{\LWRbackslash"}{}%
7315                      \LWR@replacestrings{"}{}%
7316                  }%
7317                  {%
7318                      \LWR@replacestrings{"}{&quot;}%
7319                  }%
7320          }% sanitizing enabled
7321          {}% sanitizing not enabled
7322 }
7323 \catcode`\&=4
```

\LWR@HTMLsanitize@use@tmpb {⟨*text*⟩}

Sanitizes via \LWR@HTMLsanitize@tmpb and then immediately uses the result.

```
7324 \newcommand{\LWR@HTMLsanitize@use@tmpb}[1]{%
7325     \ifbool{LWR@HTMLsanitize@tmpb@enable}%
7326         {%
7327             \def\tmpb{#1}%
7328             \LWR@HTMLsanitize@tmpb%
7329             \tmpb%
7330         }%
7331         {#1}%
7332 }
```

\LWR@subHTMLsanitize   \LWR@strresult must first be set by \LWR@HTMLsanitizedetokenized, \LWR@HTMLsanitizeexpanded,
                       or \CustomizeMathJax.

```
7333 \catcode`\#=12
7334 \newcommand{\LWR@subHTMLsanitize}{%
7335     \edef\tmpb{\detokenize\expandafter{\LWR@strresult}}%
7336     \LWR@HTMLsanitize@tmpb%
```

MATHJAX allows expressions to be defined with \newcommand. These expressions
would appear with ## for each argument, and each must be changed to a single

#. This must be done after all the above changes. Attempting another conversion after this causes an error upon further expansion.

```
7337        \LWR@replacestrings{##}{#}%
7338        \edef\LWR@strresult{\detokenize\expandafter{\tmpb}}%
7339 }
7340 \catcode`\#=6
```

\LWR@HTMLsanitizedetokenized {⟨*detokenized text*⟩}

Prints the sanitized text, already detokenized.

```
7341 \newrobustcmd{\LWR@HTMLsanitizedetokenized}[1]{%
7342        \LWR@traceinfo{LWR@HTMLsanitizedetokenized !#1!}%
```

Cancel French babel character handling, and fully expand the strings:

```
7343        \begingroup%
7344        \LWR@hook@processingtags%
7345        \edef\LWR@strresult{#1}%
7346        \LWR@subHTMLsanitize%
7347        \LWR@strresult%
7348        \endgroup%
7349        \LWR@traceinfo{LWR@HTMLsanitize done}%
7350 }
```

\LWR@HTMLsanitizeexpanded {⟨*text*⟩}

This version must be given the detokenized and expanded text. This is only used for adding math to MathJax expressions or lateximage alt tags.

```
7351 \edef\LWR@beginspaceleftbrace{begin \LWRleftbrace}
7352 \edef\LWR@beginspaceleftbrace{\detokenize\expandafter{\LWR@beginspaceleftbrace}}
7353 \edef\LWR@beginleftbrace{begin\LWRleftbrace}
7354 \edef\LWR@beginleftbrace{\detokenize\expandafter{\LWR@beginleftbrace}}
7355
7356 \edef\LWR@endspacerightbrace{end \LWRrightbrace}
7357 \edef\LWR@endspacerightbrace{\detokenize\expandafter{\LWR@endspacerightbrace}}
7358 \edef\LWR@endrightbrace{end\LWRrightbrace}
7359 \edef\LWR@endrightbrace{\detokenize\expandafter{\LWR@endrightbrace}}
7360
7361 \newrobustcmd{\LWR@HTMLsanitizeexpanded}[1]{%
```

Cancel French babel character handling, and fully expand the strings:

```
7362        \begingroup%
7363        \LWR@hook@processingtags%
```

```
7364        \edef\LWR@strresult{#1}%
```

The math expression may includes spaces between tokens, but MathJax does not want a space between \begin or \end and the following brace. This space is removed here.

```
7365        \protect\StrSubstitute{\LWR@strresult}%
7366            {\LWR@beginspaceleftbrace}{\LWR@beginleftbrace}[\LWR@strresult]%
```

```
7367        \protect\StrSubstitute{\LWR@strresult}%
7368            {\LWR@endspacerightbrace}{\LWR@endrightbrace}[\LWR@strresult]%

7369        \LWR@subHTMLsanitize%
7370        \LWR@strresult%
7371        \endgroup%
7372 }
```

65.2 Customizing MᴀᴛʜJᴀx

\LWR@customizedMathJax Additional MᴀᴛʜJᴀx definitions to be added to the start of each ʜᴛᴍʟ page.

```
7373 \newcommand*{\LWR@customizedMathJax}{}
```

LWR@warnedcustomizemathjax Used to issue only one warning about using a \CustomizeMathJax per macro.
(*bool*)
```
7374 \newbool{LWR@warnedcustomizemathjax}
7375 \boolfalse{LWR@warnedcustomizemathjax}
```

\LWR@subcustomizedmathjax {⟨*macro definition*⟩}

```
7376 \newcommand*{\LWR@subcustomizedmathjax}[1]{%
7377     \begingroup%
7378     \LWR@hook@processingtags%
7379     \boolfalse{LWR@HTMLsanitize@tmpb@removebackslash}%
7380     \edef\LWR@strresult{\detokenize{#1}}%
7381     \LWR@subHTMLsanitize%
7382     \xdef\LWR@customizedMathJax{%
7383         \LWR@customizedMathJax%
7384             \LWR@strresult%
7385     }%
7386     \endgroup%
7387 }
7388 \@onlypreamble\LWR@subcustomizedmathjax
```

\CustomizeMathJax {⟨*macro definition*⟩}

A warning is issued if a very long argument is given.

```
7389 \newcommand*{\CustomizeMathJax}[1]{%
7390     \ifbool{LWR@warnedcustomizemathjax}{}{%
7391         \StrLen{\detokenize{#1}}[\LWR@tempone]%
7392         \ifnumgreater{\LWR@tempone}{350}{%
7393             \AtEndDocument{%
7394                 \PackageNoteNoLine{lwarp}{%
7395                 To ensure faster MathJax compilation, place each\MessageBreak
7396                 custom macro in its own \protect\CustomizeMathJax.\MessageBreak
7397                 See the Lwarp documentation regarding customizing\MessageBreak
7398                     MathJax%
7399                 }%
7400             }%
7401             \booltrue{LWR@warnedcustomizemathjax}%
7402         }{}%
7403     }%
7404     \appto\LWR@customizedMathJax{\LWRbackslash(}%
```

```
7405        \LWR@subcustomizedmathjax{#1}%
7406        \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
7407 }
7408 \@onlypreamble\CustomizeMathJax
```

\LWR@infoprocessingmathjax {⟨*package name*⟩}

```
7409 \newcommand*{\LWR@infoprocessingmathjax}[1]{%
7410 \typeout{---}
7411 \typeout{Package lwarp: Processing MathJax customizations for #1.}
7412 \typeout{\space\space This may take a moment.}
7413 \typeout{---}
7414 }
```

defaults   Default customizations:

In the MATHJAX code, footnotes are only referenced. For equations, they are also
generated in the HTML when the LATEX math is generated inside the HTML comment.
For other math environments, the \footnotemark / \footnotetext method must
⚠ \footnotemark be used. See section 8.5.4 regarding \footnotemark.

For footnotes, \footnotename is used in most cases, however for equation the
footnote is picked up from LATEXin \LWR@doendequation.

First, \footnotename for MATHJAX is copied from LATEX.

```
7415 \providecommand{\footnotename}{footnote}
7416
7417 % due to warpMathJax:
7418 \end{warpHTML}
7419
7420 \begin{warpMathJax}
7421 \xdef\LWR@customizedMathJax{\LWR@customizedMathJax%
7422        \LWRbackslash(%
7423        \LWRbackslash{}newcommand%
7424        \{\LWRbackslash{}footnotename\}%
7425        \{\footnotename\}%
7426        \LWRbackslash)\par%
7427 }
7428 \end{warpMathJax}
```

\LWRfootnote is set per equation if a footnote is detected in the equation's math
expression, otherwise it defaults to \footnotename.

```
7429 \begin{warpMathJax}
7430 \CustomizeMathJax{\def\LWRfootnote{1}}
7431 \CustomizeMathJax{\newcommand{\footnote}[2][\LWRfootnote]{{}^{\mathrm{#1}}}}
7432 \CustomizeMathJax{\newcommand{\footnotemark}[1][\LWRfootnote]{{}^{\mathrm{#1}}}}
```

\hspace is modified to accept and ignore a star:

```
7433 \CustomizeMathJax{\let\LWRorighspace\hspace}
7434 \CustomizeMathJax{\renewcommand{\hspace}{\ifstar\LWRorighspace\LWRorighspace}}
```

Various other customizations:

```
7435 \CustomizeMathJax{\newcommand{\TextOrMath}[2]{#2}}
7436 \CustomizeMathJax{\newcommand{\mathnormal}[1]{{#1}}}
```

```
7437 \CustomizeMathJax{\newcommand\ensuremath[1]{#1}}
7438 \CustomizeMathJax{% absorb two optional arguments
7439     \newcommand{\LWRframebox}[2][]{\fbox{#2}}
7440     \newcommand{\framebox}[1][]{\LWRframebox}
7441 }
7442 \CustomizeMathJax{\newcommand{\setlength}[2]{}}
7443 \CustomizeMathJax{\newcommand{\addtolength}[2]{}}
7444 \CustomizeMathJax{\newcommand{\setcounter}[2]{}}
7445 \CustomizeMathJax{\newcommand{\addtocounter}[2]{}}
7446 \CustomizeMathJax{\newcommand{\arabic}[1]{}}
7447 \CustomizeMathJax{\newcommand{\number}[1]{}}
7448 \CustomizeMathJax{\newcommand{\noalign}[1]{\text{#1}\notag \\}}
7449 \CustomizeMathJax{\newcommand{\cline}[1]{}}
7450 \CustomizeMathJax{\newcommand{\directlua}[1]{\text{(directlua)}}}
7451 \CustomizeMathJax{\newcommand{\luatexdirectlua}[1]{\text{(directlua)}}}
```

\protect, \mathchar, and \delimiter are silently discarded; and \mathcode and \delcode are ignored.

```
7452 \CustomizeMathJax{\newcommand{\protect}{}}
7453 \CustomizeMathJax{\def\LWRabsorbnumber#1 {}}
7454 \CustomizeMathJax{\def\LWRabsorbquotenumber"#1 {}}
7455 \CustomizeMathJax{\newcommand{\LWRabsorboption}[1][]{}}
7456 \CustomizeMathJax{\newcommand{\LWRabsorbtwooptions}[1][]{\LWRabsorboption}}
7457 \CustomizeMathJax{\def\mathchar{\ifnextchar"\LWRabsorbquotenumber\LWRabsorbnumber}}
7458 \CustomizeMathJax{\def\mathcode#1={\mathchar}}
7459 \CustomizeMathJax{\let\delcode\mathcode}
7460 \CustomizeMathJax{\let\delimiter\mathchar}
```

Some text symbols missing from MATHJAX:

```
7461 \CustomizeMathJax{\def\oe{\unicode{x0153}}}
7462 \CustomizeMathJax{\def\OE{\unicode{x0152}}}
7463 \CustomizeMathJax{\def\ae{\unicode{x00E6}}}
7464 \CustomizeMathJax{\def\AE{\unicode{x00C6}}}
7465 \CustomizeMathJax{\def\aa{\unicode{x00E5}}}
7466 \CustomizeMathJax{\def\AA{\unicode{x00C5}}}
7467 \CustomizeMathJax{\def\o{\unicode{x00F8}}}
7468 \CustomizeMathJax{\def\O{\unicode{x00D8}}}
7469 \CustomizeMathJax{\def\l{\unicode{x0142}}}
7470 \CustomizeMathJax{\def\L{\unicode{x0141}}}
7471 \CustomizeMathJax{\def\ss{\unicode{x00DF}}}
7472 \CustomizeMathJax{\def\SS{\unicode{x1E9E}}}
7473 \CustomizeMathJax{\def\dag{\unicode{x2020}}}
7474 \CustomizeMathJax{\def\ddag{\unicode{x2021}}}
7475 \CustomizeMathJax{\def\P{\unicode{x00B6}}}
7476 \CustomizeMathJax{\def\copyright{\unicode{x00A9}}}
7477 \CustomizeMathJax{\def\pounds{\unicode{x00A3}}}
7478 \end{warpMathJax}
7479
7480
7481 \begin{warpHTML}% due to warpMathJax
```

\LWR@customizeMathJax Prints MATHJAX commands to the HTML output.

```
7482 \newcommand{\LWR@customizeMathJax}{%
7483 \ifbool{mathjax}{
```

```
7484 \LWR@stoppars
7485 \LWR@htmlcomment{MathJax customizations:}


7486 \LWR@htmlelementclass{div data-nosnippet}[display:none]{}
7487 \LWR@stoppars
```

Avoid ligatures while printing MATHJAX customizations:

```
7488 {
7489     \LWR@print@ttfamily
7490     \LWR@customizedMathJax
7491 }
7492 \LWR@htmlelementclassend{div}{}
7493 }{}
7494 }


7495 \end{warpHTML}
```

for PRINT output:
```
7496 \begin{warpprint}
```

`\CustomizeMathJax` The print-mode version:

```
7497 \newcommand*{\CustomizeMathJax}[1]{}
```

`\FilenameSimplify` * {⟨*expression*⟩}

```
7498 \NewDocumentCommand{\FilenameSimplify}{s m}{}


7499 \end{warpprint}
```

for HTML output:
```
7500 \begin{warpHTML}
```

`\LWR@createfooter` If specified, create the first or later web page footer.

```
7501 \newcommand*{\LWR@createfooter}{%
7502     \ifnumless{\value{LWR@htmlseqfilenumber}}{1}{%
7503         \ifdefempty{\LWR@firstpagebottom}{}{%
7504             \LWR@htmlelement{footer}
7505
7506             \LWR@firstpagebottom
7507
7508             \LWR@htmlelementend{footer}
7509         }%
7510     }{%
7511         \ifdefempty{\LWR@pagebottom}{}{%
7512             \LWR@htmlelement{footer}
7513
7514             \LWR@pagebottom
7515
7516             \LWR@htmlelementend{footer}
7517         }%
7518     }%
7519 }
```

`\LWR@newhtmlfile` {⟨*section name*⟩}

Finishes the current HTML page with footnotes, footer, navigation, then starts a new HTML page with an HTML comment telling where to split the page and what the new filename and CSS are, then adds navigation, side TOC, header, and starts the text body.

```
7520 \newcommand*{\LWR@newhtmlfile}[1]{
7521 \LWR@traceinfo{LWR@newhtmlfile}
```

At the bottom of the ending file:

```
7522 \LWR@htmlelementclassend{section}{textbody}
7523 \LWR@htmlelementclassend{main}{bodycontainer}
7524 \LWR@htmlelementclassend{div}{bodyandsidetoc}
7525
7526 \LWR@printpendingfootnotes
7527
```

No footer between files if EPUB:

```
7528 \ifbool{FormatEPUB}{}{\LWR@createfooter}
```

No bottom navigation if are finishing the home page or formatting for EPUB or a word-processor.

```
7529 \ifthenelse{\boolean{FormatEPUB}\OR\boolean{FormatWP}}
7530     {}
7531     {\ifnumcomp{\value{LWR@htmlfilenumber}}{>}{0}{\LWR@botnavigation}{}}
```

End of this HTML file:

```
7532 \LWR@stoppars
7533 \LWR@htmltag{/body}\LWR@orignewline
7534 \LWR@htmltag{/html}\LWR@orignewline
7535 \LWR@traceinfo{LWR@newhtmlfile: about to LWR@orignewpage}
7536 \LWR@maybe@orignewpage
7537
7538 \addtocounter{LWR@htmlfilenumber}{1}%
7538 \addtocounter{LWR@htmlseqfilenumber}{1}%
```

If using a filename based on section name, create a version without blanks. The filename without blanks will be placed into \LWR@thisfilename. Duplicates will be detected using MD5 hashes.

If not using a filename, the file number will be used instead.

```
7539 \ifbool{FileSectionNames}%
7540 {%
```

Convert the section name to a filename with blanks and common words removed. The resulting filename is in \LWR@thisfilename.

```
7541     \LWR@filenamenoblanks{#1}%
```

Create a macro name from the MD5 hash of the file name, to detect duplicates:

```
7542     \edef\LWR@hashedname{\LWR@mdfive{\LWR@thisfilename}}%
```

If the macro name is not yet defined, this filename is unique. If the filename is unique, create an empty macro using the hashed name, to be used to test for additional duplicates in the future. The first time a duplicate is found, a warning is issued and this macro then contains a 1. For further duplicates of the same name, no additional warning is issed.

```
7543      \ifcsundef{LWR@filename\LWR@hashedname}{%
7544          \csdef{LWR@filename\LWR@hashedname}{}%
7545      }{%
```

If the filename is not unique, create a warning if the first duplicate, and modify the filename by appending a unique file number.

```
7546          \ifcsempty{LWR@filename\LWR@hashedname}%
7547          {% first instance
7548              \PackageWarning{lwarp}
7549                  {%
7550                      The section name:\MessageBreak
7551                      ``#1'',\MessageBreak
7552                      at the line number listed below,\MessageBreak
7553                      generates the filename\MessageBreak
7554                      ``\LWR@thisfilename'',\MessageBreak
7555                      which appears to be a duplicate.  There is a\MessageBreak
7556              previous section with an identical or similar name.\MessageBreak
7557              A unique file number has been appended to the file name\MessageBreak
7558                      of this and any further similar files.\MessageBreak
7559              This number may change as sections are added/removed,\MessageBreak
7560                      and obsolete HTML files may result.\MessageBreak
7561                      To remove these files, use lwarpmk cleanall.\MessageBreak
7562              (While generating file names, Lwarp sanitizes math,\MessageBreak
7563                      most symbols, and a few common short words,\MessageBreak
7564                      and this may cause a conflict.)\MessageBreak
7565                      \LWR@avoiddupfilenames%
7566                  }%
7567              \csdef{LWR@filename\LWR@hashedname}{1}%
7568          }% first instance
7569          {}% repeated instances
7570          \edef\LWR@thisfilename{\LWR@thisfilename-\arabic{LWR@htmlseqfilenumber}}%
7571      }%
7572 }%
```

If using file numbers instead of names, the name is set to the next file number.

```
7573 {\renewcommand*{\LWR@thisfilename}{\arabic{LWR@htmlfilenumber}}}
```

Include an HTML comment to instruct lwarpmk where to split the files apart. Uses pipe-separated fields for split_html.gawk. Uses monospaced font with ligatures disabled for everything except the title.

```
7574 \LWR@traceinfo{LWR@newhtmlfile: about to print start file}%
```

\LWR@nullfonts to allow math in a section name.

```
7575 \begingroup%
7576 \LWR@nullfonts%
7577 \LWR@htmlblockcomment{%
7578 |Start file|%
7579 \LWR@htmlsectionfilename{\LWR@thisfilename}|%
```

```
7580 }
7581 \endgroup%
```

At the top of the starting file:

```
7582 \LWR@stoppars
7583
```

Start a new file with the given section name:

```
7584 \LWR@filestart[#1]
7585
```

Track the PDF page numbers of the HTML output. This is updated more frequently than LWR@currentautosecpage.

```
7586 \setcounter{LWR@currentautosecfloatpage}{\value{page}}%
7587 \LWR@newautopagelabel{LWR@currentautosecfloatpage}%
```

No navigation between files if formatting for an EPUB or word processor:

```
7588 \ifthenelse{\boolean{FormatEPUB}\OR\boolean{FormatWP}}
7589     {}
7590     {\LWR@topnavigation}
7591
```

No header if between files if formatting for an EPUB or word processor:

```
7592 \ifthenelse{\boolean{FormatEPUB}\OR\boolean{FormatWP}}
7593     {}
7594     {
7595         \ifdefempty{\LWR@pagetop}{}{
7596             \LWR@htmlelement{header}
7597
7598             \LWR@pagetop
7599
7600             \LWR@htmlelementend{header}
7601         }
7602     }
7603
```

The container for the sidetoc and text body:

```
7604 \LWR@htmlelementclass{div}{bodyandsidetoc}
```

No sideTOC if formatting for an EPUB or word processor:

```
7605 \ifthenelse{\boolean{FormatEPUB}\OR\boolean{FormatWP}}
7606     {}
7607     {\LWR@sidetoc}
7608
```

Start of the <textbody>:

```
7609 \LWR@htmlelementclass{main}{bodycontainer}
7610 \LWR@htmlelementclass{section}{textbody}
```

Not yet found a new section in this file. Once one is found, a label will be placed for previous/next links.

```
7611 \boolfalse{LWR@setseqfilelabel}
```

Print title only if there is one. Skip if formatting for an EPUB or word processor:

```
7612 \ifthenelse{\boolean{FormatEPUB}\OR\boolean{FormatWP}}%
7613     {}%
7614     {%
7615         \ifcsvoid{thetitle}{}{%
7616             \LWR@printthetitle%
7617         }%
7618     }%
```

Keep paragraph tags disabled for now:

```
7619 \LWR@stoppars
7620
```

If using MathJax, print the customizations here.

```
7621 \LWR@customizeMathJax
```

```
7622 \LWR@traceinfo{LWR@newhtmlfile: done}
7623 }
```

```
7624 \end{warpHTML}
```

66 Sectioning

Sectioning and cross-references have been emulated from scratch, rather than try to patch several layers of existing LaTeX code and packages. Formatting is handled by CSS, so the emulated code has much less work to do than the print versions.

Unicode Section names and the resulting filenames with accented characters are partially supported, depending on the ability of *pdflatex* to generate characters and *pdftotext*

⚠ accents in filenames to read them. If extra symbols appear in the text, it may be that *pdflatex* is actually producing a symbol over or under a character, resulting in *pdftotext* picking up the accent symbol separately.

XƎLaTeX and LuaLaTeX directly support accented section and file names, but it may be necessary to use LaTeX accents instead of native Unicode accents. LaTeX accents will have the accents stripped when creating file names, whereas using Unicode accents will create filenames which include accents, which may cause issues with some operating systems.

for HTML output: `7625 \begin{warpHTML}`

66.1 User-level starred section commands

\ForceHTMLPage For HTML output, forces the next section to be on its own HTML page, if FileDepth allows, even if starred. For use with \printindex and others which generate a starred section which should be on its own HTML page. Also see \ForceHTMLTOC.

For print output, no effect.

```
7626 \newbool{LWR@forcinghtmlpage}
7627 \boolfalse{LWR@forcinghtmlpage}
7628
7629 \newcommand*{\ForceHTMLPage}{%
7630 \global\booltrue{LWR@forcinghtmlpage}%
7631 }
```

\ForceHTMLTOC   For HTML output, forces the next section to have a TOC entry, even if starred. For
use with \printindex and others which generate a starred section which should
be in the TOC so that it may be accessed via HTML. Not necessary if used with
tocbibind. Also see \ForceHTMLPage.

For print output, no effect.

```
7632 \newbool{LWR@forcinghtmltoc}
7633 \boolfalse{LWR@forcinghtmltoc}
7634
7635 \newcommand*{\ForceHTMLTOC}{%
7636 \global\booltrue{LWR@forcinghtmltoc}%
7637 }
```

```
7638 \end{warpHTML}
```

for PRINT output:
```
7639 \begin{warpprint}
7640 \newcommand*{\ForceHTMLPage}{}
7641 \newcommand*{\ForceHTMLTOC}{}
7642 \end{warpprint}
```

for HTML output:
```
7643 \begin{warpHTML}
```

66.2   Book class commands

\mainmatter   Declare the main matter section of the document. Does not reset the page number,
⚠   which must be consecutive arabic numbers for the HTML conversion.

```
7644 \newbool{LWR@mainmatter}
7645 \DeclareDocumentCommand{\mainmatter}{}{%
7646 \booltrue{LWR@mainmatter}%
7647 }
```

\frontmatter   Declare the front matter section of the document, using arabic numbering for the
internal numbering. Does not reset the page number.

```
7648 \DeclareDocumentCommand{\frontmatter}{}{%
7649 \boolfalse{LWR@mainmatter}%
7650 }
```

\backmatter   Declare the back matter section of the document. Does not reset the page number.

```
7651 \DeclareDocumentCommand{\backmatter}{}{%
7652 \boolfalse{LWR@mainmatter}
7653 }
```

66.3 Sectioning support macros

\LWR@sectionumber {⟨*section type*⟩}

Typeset a section number and its trailing space with css formatting:

```
7654 \newcommand*{\LWR@sectionnumber}[1]{%
7655 \InlineClass{sectionnumber}{#1}%
7656 }
```

autosec A tag used by the TOC and index.

\LWR@createautosec {⟨*section type*⟩}

Create an autosection tag.

The use of \textquotedbl instead of " provides improved compatibility with xeCJK.

```
7657 \newcommand*{\LWR@createautosec}[1]{%
7658 \LWR@htmltag{%
7659     #1 % space
7660     id=\textquotedbl\LWR@print@mbox{autosec-\arabic{page}}\textquotedbl%
7661 }%
7662 }
```

\LWR@pushoneclose {⟨*sectiontype*⟩} Stacks the new sectioning level's closing tag, to be used when this section is closed some time later.

⚠ \LWR@stoppars must be executed first.

```
7663 \NewDocumentCommand{\LWR@pushoneclose}{m}{%
7664 \LWR@traceinfo{LWR@pushoneclose #1}%
7665     \LWR@pushclose{#1}%
7666 }
```

\LWR@startnewdepth {⟨*sectiontype*⟩}

Closes currently stacked tags of a lesser level, then opens the new nesting level by saving this new sectioning level's closing tag for later use.

⚠ \LWR@stoppars must be executed first.

```
7667 \NewDocumentCommand{\LWR@startnewdepth}{m}{%
```

Close any stacked sections up to this new one.

```
7668 \LWR@closeprevious{#1}%
```

Push a new section depth:

```
7669 \LWR@pushoneclose{#1}%
7670 }
```

LWR@prevFileDepth (*Ctr*) Remembers the previous LWR@FileDepth.

Initialized to a deep level so that any section will trigger a new HTML page after the home page.

```
7671 \newcounter{LWR@prevFileDepth}
7672 \setcounter{LWR@prevFileDepth}{\LWR@depthsubparagraph}
```

\@seccntformat {⟨*sectiontype*⟩}

```
7673 \def\@seccntformat#1{\csname the#1\endcsname\quad}
```

\simplechapterdelim Used by tocbibind and anonchap.

```
7674 \newcommand*{\simplechapterdelim}{}
```

\@chapcntformat {⟨*sectiontype*⟩}

\let to \@seccntformat by default, but may be redefined by \simplechapter and \restorechapter from tocbibind or anonchap.

```
7675 \let\@chapcntformat\@seccntformat
```

\@partcntformat {⟨*sectiontype*⟩}

\let to \@seccntformat by default, but may be redefined by ctex.

```
7676 \let\@partcntformat\@seccntformat
```

\@partnameformat Prints "Part" for part sections.

Nullified by ctex.

```
7677 \newcommand*{\@partnameformat}{\LWR@isolate{\partname}~}%
```

\LWR@printchaptername Print \chaptername in most cases, but this is nullified for ctexbook, komascript, ujt* classes.

```
7678 \newcommand*{\LWR@printchaptername}{%
7679     \ifdefvoid{\chaptername}{}{\chaptername~}%
7680 }
```

\LWR@section * [⟨*TOC name*⟩] {⟨*name*⟩} {⟨*sectiontype*⟩}

The common actions for the high-level sectioning commands.

```
7681 \DeclareDocumentCommand{\LWR@section}{m m m m}{%
7682 \IfValueTF{#2}%
7683     {\LWR@traceinfo{LWR@section: starting #4 #2}}%
7684     {\LWR@traceinfo{LWR@section: starting #4 #3}}%
```

Warn if starting a section inside a <span>:

```
7685     \LWR@spanwarninvalid{section}%
```

```
7686 \LWR@maybeprintpendingfootnotes{\csuse{LWR@depth#4}}%
7687 \LWR@stoppars%
7688 \LWR@startnewdepth{#4}%
```

Cancel special minipage horizontal space interaction:

```
7689 \global\boolfalse{LWR@minipagethispar}%
```

Start a new HTML file unless starred, and if is a shallow sectioning depth.

Exception: Also start a new HTML file for \part*, for appendix.

Generate a new LaTeX page so that TOC and index page number points to the section:

```
7690 \LWR@traceinfo{LWR@section: testing whether to start a new HTML file}%
7691 \IfBooleanT{#1}{\LWR@traceinfo{LWR@section: starred}}%
7692 \ifbool{LWR@forcinghtmlpage}{\LWR@traceinfo{LWR@section: forcinghtmlpage}}{}%
7693 \ifthenelse{%
7694     \(%
7695         \(\NOT\equal{#1}{\BooleanTrue}\)\OR%
7696         \(\cnttest{\@nameuse{LWR@depth#4}}{=}{\LWR@depthpart}\)\OR%
7697         \(\boolean{LWR@forcinghtmlpage}\)%
7698     \)%
7699     \AND%
7700     \cnttest{\@nameuse{LWR@depth#4}}{<=}{\value{FileDepth}}%
7701     \AND%
7702     \(%
7703         \NOT\boolean{CombineHigherDepths}\OR%
7704         \cnttest{\@nameuse{LWR@depth#4}}{<=}{\value{LWR@prevFileDepth}}%
7705     \)%
7706     \AND%


7707     \(% phantomsection
7708         \NOT\isempty{#3}%
7709         \OR%
7710         \(\NOT\equal{#1}{\BooleanTrue}\)%
7711     \)%
7712 }%
```

If so: start a new HTML file:

```
7713 {% new file
7714     \LWR@traceinfo{LWR@section: new HTML file}%
```

See if there was an optional TOC name entry:

```
7715     \IfNoValueTF{#2}%
```

If no optional entry

```
7716         {\LWR@newhtmlfile{#3}}%
```

If yes an optional entry

```
7717         {\LWR@newhtmlfile{#2}}%
7718 }% new file
```

Else: No new HTML file:

```
7719 {% not new file
```

Generate a new LaTeX page so that TOC and index page number points to the section:

```
7720     \LWR@traceinfo{LWR@section: not a new HTML file, about to LWR@orignewpage}%
7721     \LWR@stoppars%
7722     \LWR@maybe@orignewpage%
7723 }% not new file
```

```
7724 %
7725 % Remember this section's name for \cs{nameref}:
7726 %     \begin{macrocode}
7727 \IfValueT{#3}{%
7728     \LWR@traceinfo{LWR@section: about to LWR@setlatestname}%
7729     \IfValueTF{#2}{\LWR@setlatestname{#2}}{\LWR@setlatestname{#3}}%
7730 }%
```

Print an opening comment with the level and the name; ex: "section" "Introduction" Footnotes may be used in section names, which would also appear in the HTML section opening comments, so the short TOC entry is used if possible, and a limited opening comment is made if the sectional unit is starred.

Avoid extra <par> tag:

```
7731 \LWR@stoppars%
```

Form a sectioning comment:

```
7732 \begingroup%
7733 \LWR@nullfonts%
7734 \LWR@nullifyfootnotes%
7735 \LWR@htmlcomment{%
7736     \LWR@orignewline%
7737     \IfValueTF{#2}%
7738         {...... #4 #2 ......}%
7739         {...... #4 #3 ......}%
7740     \LWR@orignewline%
7741 }%
7742 \LWR@orignewline%
7743 \ifbool{HTMLDebugComments}%
7744     {%
7745         \IfBooleanTF{#1}% starred
7746             {%
7747                 \IfNoValueTF{#2}% short TOC
7748                     {\LWR@htmlcomment{Opening #4* ``#3''}}%
7749                     {\LWR@htmlcomment{Opening #4* ``#2''}}%
7750             }%
7751             {%
7752                 \IfNoValueTF{#2}% short TOC
7753                     {\LWR@htmlcomment{Opening #4 ``#3''}}%
7754                     {\LWR@htmlcomment{Opening #4 ``#2''}}%
7755             }%
7756         \LWR@orignewline%
7757     }%
7758     {}%
7759 \endgroup%
```

For inline sections paragraph and subparagraph, start a new paragraph now:

```
7760 \ifthenelse{%
7761     \cnttest{\@nameuse{LWR@depth#4}}{>=}{\LWR@depthparagraph}%
7762 }%
7763     {\LWR@startpars}%
7764     {}%
```

Create the opening tag with an autosec:

```
7765 \LWR@traceinfo{LWR@section: about to LWR@createautosec}%
7766 \LWR@createautosec{\@nameuse{LWR@tag#4}}%
```

```
7767 \setcounter{LWR@currentautosecpage}{\value{page}}%
```

Check if starred:

```
7768 \IfBooleanTF{#1}%
7769 {%
7770     \LWR@traceinfo{LWR@section: starred}%
```

Starred, but also forcing a TOC entry, so add unnumbered TOC name or regular name:

```
7771     \ifbool{LWR@forcinghtmltoc}%
7772     {%
7773         \addcontentsline{toc}{#4}{%
7774             \IfValueTF{#2}{\LWR@isolate{#2}}{\LWR@isolate{#3}}%
7775         }%
7776     }%
7777     {}%
7778 }% starred
```

Not starred, so step counter and add to TOC:

```
7779 {% not starred
```

Only add a numbered TOC entry if section number is not too deep:

```
7780     \ifthenelse{%
7781         \cnttest{\@nameuse{LWR@depth#4}}{<=}{\value{secnumdepth}}%
7782     }%
7783     {% if secnumdepth
```

If in the main matter, step the counter and add the TOC entry. For article class, lwarp assumes that all is mainmatter.

```
7784         \LWR@traceinfo{LWR@section: about to test main matter}%
7785         \ifbool{LWR@mainmatter}%
7786         {%
7787             \LWR@traceinfo{LWR@section: yes mainmatter}%
7788             \refstepcounter{#4}%
```

Add main matter numbered TOC entry with the TOC name or the regular name:

```
7789             \LWR@traceinfo{LWR@section: about to addcontentsline}%
7790             \addcontentsline{toc}{#4}%
```

```
7791            {%
7792                \protect\numberline{%
7793                    \@nameuse{pre#4name}%
7794                    \@nameuse{the#4}%
7795                    \@nameuse{post#4name}%
7796                }%
7797                {%
7798                    \ignorespaces%
7799                \IfValueTF{#2}{\LWR@isolate{#2}}{\LWR@isolate{#3}}\protect\relax%
7800                }%
7801            }%
7802            \LWR@traceinfo{LWR@section: finished addcontentsline}%
7803        }% end of if main matter
```

If not main matter, add unnumbered TOC name or regular name:

```
7804        {% not main matter
7805            \LWR@traceinfo{LWR@section: no main matter}%
7806            \addcontentsline{toc}{#4}{%
7807                \IfValueTF{#2}{\LWR@isolate{#2}}{\LWR@isolate{#3}}%
7808            }%
7809        }% end of not main matter
7810    }% end of secnumdepth
```

Deeper than secnumdepth, so add an unnumbered TOC entry:

```
7811    {%
7812        \addcontentsline{toc}{#4}{%
7813            \IfValueTF{#2}{\LWR@isolate{#2}}{\LWR@isolate{#3}}%
7814        }%
7815    }%
```

For part, print "Part":

```
7816    \ifbool{LWR@mainmatter}%
7817    {%
7818        \ifthenelse{%
7819            \(\cnttest{\@nameuse{LWR@depth#4}}{<=}%
7820                {\value{secnumdepth}}\) \AND%
7821            \(\cnttest{\@nameuse{LWR@depth#4}}{=}{\LWR@depthpart}\)%
7822        }%
7823            {\@partnameformat}%
7824            {}%
```

Print the section number:

```
7825        \LWR@traceinfo{LWR@section: about to print section number}%
7826        \ifthenelse{%
7827            \cnttest{\@nameuse{LWR@depth#4}}{<=}{\value{secnumdepth}}%
7828        }%
7829            {%
7830                \ifstrequal{#4}{part}%
7831                {\protect\LWR@sectionnumber{\@partcntformat{#4}}}%
7832                {%
7833                    \ifstrequal{#4}{chapter}%
7834                        {%

7835                            \LWR@printchaptername%
7836                            \protect\LWR@sectionnumber{\@chapcntformat{#4}}%
```

```
7837                              }%
7838                                 {\protect\LWR@sectionnumber{\@seccntformat{#4}}}%
7839                       }%
7840                    }%
7841                 {}%
7842          \LWR@traceinfo{LWR@section: finished print section number}%
7843      }{}%
7844 }% not starred
```

Print the section name:

```
7845 \LWR@traceinfo{LWR@section: about to print the section name}%
7846 \LWR@isolate{#3}%
```

Close the heading tag, such as /H2:

```
7847 \LWR@traceinfo{LWR@section: about to close the heading tag}%
7848 \LWR@htmltag{\@nameuse{LWR@tag#4end}}%
7849 \LWR@orignewline%
```

Generate a LaTeX label.

Track the PDF page numbers of the HTML output. A new autopage label may be generated for LWR@currentautosecpage for the start of the section, and also for the current page if it is different due to an SVG image in the section name. Also, the final page after the section has been created is updated in LWR@currentautosecfloatpage.

```
7850 \LWR@traceinfo{LWR@section: about to create the LaTeX label}%
7851 \setcounter{LWR@currentautosecfloatpage}{\value{page}}%
7852 \LWR@newautopagelabel{LWR@currentautosecpage}\LWR@orignewline%
```

If this is the first section found in this file, create a label for prevous/next links:

```
7853 \ifbool{LWR@setseqfilelabel}{}{%
7854    \label{\BaseJobname-autofile-\arabic{LWR@htmlseqfilenumber}}}%
7855    \booltrue{LWR@setseqfilelabel}%
7856 }%
```

Start paragraph handing unless is an inline paragraph or subparagraph:

```
7857 \ifthenelse{%
7858    \cnttest{\@nameuse{LWR@depth#4}}{<}{\LWR@depthparagraph}%
7859 }%
7860    {\LWR@startpars}%
7861    {}%
```

If not starred, remember the previous depth to possibly trigger a new HTML page.

HOWEVER, allow a \part* to start a new HTML page. This is used by appendix.

A starred section does not trigger a new HTML page at the beginning of this macro, so it should not affect it here at the end either. This became an issue when a \listoftables was tested in the middle of the document. The \chapter* for the list was not allowing a new HTML page for the section following it while CombineHigherDepths was true.

```
7862 \ifthenelse{%
7863     \NOT\equal{#1}{\BooleanTrue}\OR%
7864     \cnttest{\@nameuse{LWR@depth#4}}{=}{\LWR@depthpart}%
7865 }%
7866     {% not starred
7867         \setcounter{LWR@prevFileDepth}{\@nameuse{LWR@depth#4}}%
7868     }% not starred
7869     {}%
```

Reset to defaults if not a phantomsection:

```
7870 \ifstrempty{#3}%
7871     {}%
7872     {%
7873         \global\boolfalse{LWR@forcinghtmlpage}%
7874         \global\boolfalse{LWR@forcinghtmltoc}%
7875     }%
7876 %
7877 \LWR@traceinfo{LWR@section: done}%
7878 }
```

66.4   Pre- and post- sectioning names

\prebookname   Usually null, but is used by uj* and ut* Japanese classes.
\postbookname
```
7879 \providecommand*{\prebookname}{}
7880 \providecommand*{\postbookname}{}
```

\prepartname   Usually null, but is used by uj* and ut* Japanese classes.
\postpartname
```
7881 \providecommand*{\prepartname}{}
7882 \providecommand*{\postpartname}{}
```

\prechaptername   Usually null, but is used by uj* and ut* Japanese classes.
\postchaptername
```
7883 \providecommand*{\prechaptername}{}
7884 \providecommand*{\postchaptername}{}
```

\presectionname   Always null, but provided here for algorithmic simplicity in \LWR@section.
\postsectionname
```
7885 \providecommand*{\presectionname}{}
7886 \let\postsectionname\presectionname
7887
7888 \let\presubsectionname\presectionname
7889 \let\postsubsectionname\postsectionname
7890
7891 \let\presubsubsectionname\presectionname
7892 \let\postsubsubsectionname\postsectionname
7893
7894 \let\preparagraphname\presectionname
7895 \let\postparagraphname\postsectionname
7896
7897 \let\presubparagraphname\presectionname
7898 \let\postsubparagraphname\postsectionname
```

66.5 \section and friends

For memoir, a second optional argument is allowed.

For hypbmsec, a second optional argument or either parenthesis argument is allowed.

Each of these additional arguments are for headers or PDF bookmarks, and are ignored for HTML output.

\part * (⟨*2:PDF name*⟩) [⟨*3:TOC name*⟩] [⟨*4:PDF name*⟩] (⟨*5:PDF name*⟩) {⟨*6:name*⟩}

```
7899 \newcommand{\part@preamble}{}% for koma-script
7900
7901 \DeclareDocumentCommand{\part}{s d() o o d() m}{%
7902     \LWR@section{#1}{#3}{#6}{part}%
7903
7904     \part@preamble% for koma-script
7905     \renewcommand{\part@preamble}{}%
7906 }
```

\chapter * (⟨*2:PDF name*⟩) [⟨*3:TOC name*⟩] [⟨*4:PDF name*⟩] (⟨*5:PDF name*⟩) {⟨*6:name*⟩}

```
7907 \let\@printcites\relax% for quotchap package
7908
7909 \newcommand{\chapter@preamble}{}% for koma-script
7910
7911 \@ifundefined{chapter}
7912 {}
7913 {%
7914     \DeclareDocumentCommand{\chapter}{s d() o o d() m}{%
7915         \LWR@section{#1}{#3}{#6}{chapter}%
7916
7917         \@printcites% for quotchap package
7918
7919         \chapter@preamble% for koma-script
7920         \renewcommand{\chapter@preamble}{}%
7921     }
7922 }
```

\section * (⟨*2:PDF name*⟩) [⟨*3:TOC name*⟩] [⟨*4:PDF name*⟩] (⟨*5:PDF name*⟩) {⟨*6:name*⟩}

```
7923 \DeclareDocumentCommand{\section}{s d() o o d() m}{%
7924     \LWR@section{#1}{#3}{#6}{section}%
7925 }
```

\subsection * (⟨*2:PDF name*⟩) [⟨*3:TOC name*⟩] [⟨*4:PDF name*⟩] (⟨*5:PDF name*⟩) {⟨*6:name*⟩}

```
7926 \DeclareDocumentCommand{\subsection}{s d() o o d() m}{%
7927     \LWR@section{#1}{#3}{#6}{subsection}%
7928 }
```

\subsubsection * (⟨*2:PDF name*⟩) [⟨*3:TOC name*⟩] [⟨*4:PDF name*⟩] (⟨*5:PDF name*⟩) {⟨*6:name*⟩}

```
7929 \DeclareDocumentCommand{\subsubsection}{s d() o o d() m}{%
7930     \LWR@section{#1}{#3}{#6}{subsubsection}%
7931 }
```

\paragraph * (⟨*2:PDF name*⟩) [⟨*3:TOC name*⟩] [⟨*4:PDF name*⟩] (⟨*5:PDF name*⟩) {⟨*6:name*⟩}

```
7932 \DeclareDocumentCommand{\paragraph}{s d() o o d() m}{%
7933     \LWR@section{#1}{#3}{#6}{paragraph}%
7934 }
```

\subparagraph * (⟨*2:PDF name*⟩) [⟨*3:TOC name*⟩] [⟨*4:PDF name*⟩] (⟨*5:PDF name*⟩) {⟨*6:name*⟩}

```
7935 \DeclareDocumentCommand{\subparagraph}{s d() o o d() m}{%
7936     \LWR@section{#1}{#3}{#6}{subparagraph}%
7937 }
```

```
7938 \end{warpHTML}
```

67   Starting a new file

`7939 \begin{warpall}`

\HTMLLanguage Default language for the HTML lang tag.

```
7940 \newcommand*{\LWR@currentHTMLLanguage}{en-US}
7941
7942 \newcommand*{\HTMLLanguage}[1]{%
7943     \renewcommand*{\LWR@currentHTMLLanguage}{#1}%
7944 }
```

\theHTMLTitleSeparator May be used inside \theHTMLTitleSection to separate the website's overall HTML title and the particular page's section name.

```
7945 \ifPDFTeX% pdflatex or dvi latex
7946     \ifdefstring{\inputencodingname}{utf8}{%
7947         \newcommand*{\theHTMLTitleSeparator}{ —\ }% EMdash
7948     }{%
7949         \newcommand*{\theHTMLTitleSeparator}{ -\ }% hyphen
7950     }%
7951 \else%
7952     \ifpTeX
7953         \newcommand*{\theHTMLTitleSeparator}{ -\ }% hyphen
7954     \else
7955         \newcommand*{\theHTMLTitleSeparator}{ —\ }% EMdash
7956     \fi%
7957 \fi%
```

\HTMLTitleBeforeSection Sets the HTML page's meta title tag to show the website title before the section
name.

```
7958 \newcommand*{\HTMLTitleBeforeSection}{%
7959     \def\theHTMLTitleSection{%
7960         \theHTMLTitle\theHTMLTitleSeparator\theHTMLSection%
7961     }%
7962 }
```

\HTMLTitleAfterSection Sets the HTML page's meta title tag to show the section name before the website
title.

```
7963 \newcommand*{\HTMLTitleAfterSection}{%
7964     \def\theHTMLTitleSection{%
7965         \theHTMLSection\theHTMLTitleSeparator\theHTMLTitle%
7966     }%
7967 }
```

\theHTMLTitleSection Forms the HTML page's meta title tag. The default is to show the website title
before the section name.

```
7968 \HTMLTitleBeforeSection
```

\theHTMLSection The section name is passed to \LWR@filestart, which then sets \theHTMLSection
for use inside \theHTMLTitleSection to create an HTML meta title tag.

```
7969 \newcommand*{\theHTMLSection}{}
```

```
7970 \end{warpall}
```

for HTML output: ```7971 \begin{warpHTML}```

\LWR@filestart [⟨*section name*⟩]   Creates the opening HTML tags.

```
7972 \newcommand*{\LWR@filestart}[1][]{%
7973 \LWR@traceinfo{LWR@filestart !#1!}%
```

Locally temporarily disable direct-formatting commands:

```
7974 \begingroup%
7975 \LWR@nullfonts%
```

Save the section name for use while creating the HTML meta title tag:

```
7976 \edef\theHTMLSection{#1}%
```

Remove extra material:

```
7977 \StrSubstitute{\theHTMLSection}{\protect}{\detokenize{-}}[\theHTMLSection]%
7978 \StrSubstitute{\theHTMLSection}%
7979     {\detokenize{-----}}{\detokenize{-}}[\theHTMLSection]%
7980 \StrSubstitute{\theHTMLSection}%
7981     {\detokenize{---}}{\detokenize{-}}[\theHTMLSection]%
7982 \StrSubstitute{\theHTMLSection}%
7983     {\detokenize{---}}{\detokenize{-}}[\theHTMLSection]%
```

```
7984 \StrSubstitute{\theHTMLSection}%
7985     {\detokenize{--}}{\detokenize{-}}[\theHTMLSection]%
```

If starts with a dash, remove the leading dash:

```
7986 \IfBeginWith{\theHTMLSection}{\detokenize{-}}{%
7987     \StrGobbleLeft{\theHTMLSection}{1}[\theHTMLSection]%
7988 }{}%
```

Create the page's HTML header:

```
7989 \LWR@htmltag{!DOCTYPE html}\LWR@orignewline
```

The language is user-adjustable:

NOTE: \LWR@orig@textquotedbl is used here because \textquotedbl is nullified by \LWR@nullfonts while starting the new file.

```
7990 \LWR@htmltag{%
7991     html lang=\LWR@orig@textquotedbl\LWR@currentHTMLLanguage\LWR@orig@textquotedbl%
7992 }\LWR@orignewline
```

Start of the meta data:

```
7993 \LWR@htmltag{head}\LWR@orignewline
```

Charset is fixed at UTF-8:

```
7994 \LWR@htmltag{%
7995     meta charset=\LWR@orig@textquotedbl{}UTF-8\LWR@orig@textquotedbl\ /%
7996 }\LWR@orignewline
```

Author:

```
7997 \ifthenelse{\equal{\theHTMLAuthor}{}}%
7998     {}%
7999     {%
8000         \LWR@htmltag{%
8001         meta name=\LWR@orig@textquotedbl{}author\LWR@orig@textquotedbl\ % space
8002         content=\LWR@orig@textquotedbl\theHTMLAuthor\LWR@orig@textquotedbl\ /%
8003         }\LWR@orignewline%
8004     }%
```

lwarp is the generator:

```
8005 \LWR@htmltag{%
8006     meta % space
8007     name=\LWR@orig@textquotedbl{}generator\LWR@orig@textquotedbl\ % space
8008     content=\LWR@orig@textquotedbl{}LaTeX Lwarp package\LWR@orig@textquotedbl\ /%
8009 }\LWR@orignewline%
```

If there is a description, add it now:

```
8010 \ifdefempty{\LWR@currentHTMLDescription}{}{%
8011     \LWR@htmltag{%
8012         meta name=%
8013             \LWR@orig@textquotedbl{}description\LWR@orig@textquotedbl\ % space
8014         content=%
```

```
8015        \LWR@orig@textquotedbl\LWR@currentHTMLDescription\LWR@orig@textquotedbl\ /%
8016    }\LWR@orignewline
8017 }%
```

If there are keywords, add it now:

```
8018 \ifdefempty{\LWR@currentHTMLKeywords}{}{%
8019    \LWR@htmltag{%
8020        meta name=%
8021            \LWR@orig@textquotedbl{}keywords\LWR@orig@textquotedbl\ % space
8022        content=%
8023        \LWR@orig@textquotedbl\LWR@currentHTMLKeywords\LWR@orig@textquotedbl\ /%
8024    }\LWR@orignewline
8025 }%
```

Mobile-friendly viewport:

```
8026 \LWR@htmltag{%
8027    meta % space
8028    name=\LWR@orig@textquotedbl{}viewport\LWR@orig@textquotedbl\ % space
8029    content=%
8030        \LWR@orig@textquotedbl{}width=device-width, initial-scale=1.0\LWR@orig@textquotedbl\ /%
8031 }\LWR@orignewline
```

Custom HTML meta tags:

```
8032 \LWR@HTMLmeta
```

The page's title, if there is one. A section name is also added if given.

```
8033 \ifthenelse{\equal{\theHTMLTitle}{}}%
8034    {}%
8035    {%
8036        \LWR@htmltag{title}%
8037        \ifdefempty{\theHTMLSection}%
8038            {\theHTMLTitle}%
8039            {\theHTMLTitleSection}%
8040        \LWR@htmltag{/title}\LWR@orignewline%
8041    }%
```

The page's stylesheet:

```
8042 \LWR@htmltag{%
8043    link % space
8044    rel=\LWR@orig@textquotedbl{}stylesheet\LWR@orig@textquotedbl\ % space
8045    type=\LWR@orig@textquotedbl{}text/css\LWR@orig@textquotedbl\ % space
8046    href=\LWR@orig@textquotedbl\LWR@currentcss\LWR@orig@textquotedbl\ /%
8047 }%
8048 \LWR@orignewline
```

Optional MATHJAX support. The HTML tags must be turned off during the verbatim input, and the paragraph handling which was turned on at the end of verbatim input must be immediately turned off again.

```
8049 \ifbool{mathjax}%
8050 {%
8051    \begingroup%
8052    \LWR@restoreoriglists%
```

```
8053      \boolfalse{LWR@verbtags}%
8054      \boolfalse{LWR@HTMLsanitize@tmpb@removebackslash}%

8055        \IfFileExists{\LWR@mathjaxfilename}%
8056            {\verbatiminput{\LWR@mathjaxfilename}}%
8057            {%
8058                \PackageError{lwarp}%
8059                    {%
8060                    \protect\MathJaxFilename\space specified the file\MessageBreak
8061                        \space\space\LWR@mathjaxfilename\MessageBreak
8062                        which does not exist%
8063                    }%
8064            {Specify an existing file, or remove \protect\MathJaxFilename.}%
8065            }%

8066      \endgroup%
8067      \LWR@stoppars%
8068 }% end of mathjax
8069 {}%
```

End of the header:

```
8070 \LWR@htmltag{/head}\LWR@orignewline
```

Start of the body:

```
8071 \LWR@htmltag{body}\LWR@orignewline

8072 \endgroup%
8073 \LWR@traceinfo{LWR@filestart: done}%
8074 }

8075 \end{warpHTML}
```

68   Starting HTML output

`8076 \begin{warpHTML}`

\LWR@LwarpStart Executed at the beginning of the entire document.

The use of \textquotedbl instead of " improves compatibility with xeCJK.

```
8077 \catcode`\$=\active
8078 \newcommand*{\LWR@LwarpStart}
8079 {%
8080 \LWR@traceinfo{LWR@lwarpStart}
```

If formatting for a word processor, force filedepth to single-file only, force HTML debug comments off.

```
8081 \ifbool{FormatWP}{%
8082      \setcounter{FileDepth}{-5}%
8083      \boolfalse{HTMLDebugComments}%
8084 }{}
```

Expand and detokenize \HomeHTMLFilename and \HTMLFilename:

```
8085 \edef\LWR@strresult{\HomeHTMLFilename}
8086 \edef\HomeHTMLFilename{\detokenize\expandafter{\LWR@strresult}}
8087 \edef\LWR@strresult{\HTMLFilename}
8088 \edef\HTMLFilename{\detokenize\expandafter{\LWR@strresult}}
```

Force onecolumn and empty page style:

```
8089 \LWR@origonecolumn%
8090 \LWR@origpagestyle{empty}%
```

No black box for overfull lines:

```
8091 \overfullrule=0pt
```

Reduce chance of line overflow when HTML tags are added:

```
8092 \LWR@print@footnotesize%
```

In PDF output, don't allow line breaks to interfere with HTML tags:

```
8093 \LWR@print@raggedright%
8094 \LetLtxMacro{\\}{\LWR@endofline}%
```

Spread the lines for *pdftotext* to read them well:

```
8095 \linespread{1.3}%
```

For *pdftotext* to reliably identify paragraph splits:

```
8096 \setlength{\parindent}{0pt}
8097 \setlength{\parskip}{2ex}
```

For the lateximage record file:

```
8098 \immediate\openout\LWR@lateximagesfile=\BaseJobname-images.txt
```

Removes space around the caption in the HTML:

```
8099 \setlength{\belowcaptionskip}{0ex}
8100 \setlength{\abovecaptionskip}{0ex}
```

Redefine the plain page style to be empty when used by index pages:

```
8101 \renewcommand{\ps@plain}{}
```

Plug in some new actions. This is done just before the document start so that they won't be over-written by some other package.

Float captions:

```
8102 \let\LWR@origcaption\caption
```

Not yet started any paragraph handling:

```
8103 \global\booltrue{LWR@doingparhooks}
```

```
8104 \global\boolfalse{LWR@doingapar}
8105 \global\boolfalse{LWR@doingstartpars}
```

\color@endgroup's \endgraf was conflicting with lwarp's paragraph handling.

```
8106 \let\color@endgroup\endgroup
```

Document and page settings:

```
8107 \mainmatter
8108 \LWR@origpagenumbering{arabic}
```

Start a new HTML file and a header:

```
8109 \LWR@traceinfo{LWR@lwarpStart: Starting new file.}
8110 \LWR@filestart%
```

Tell *lwarpmk* that the lwarp package is in use. This allows *lwarpmk* to warn if usepackage{lwarp} was somehow disabled.

```
8111 \begingroup%
8112 \LWR@nullfonts%
8113 \LWR@htmlblockcomment{%
8114 |Using lwarp|%
8115 \LWR@htmlsectionfilename{\LWR@thisfilename}|%
8116 }
8117 \endgroup%
```

```
8118 \LWR@traceinfo{LWR@lwarpStart: Generating first header.}
8119 \ifdefempty{\LWR@firstpagetop}{}{%
8120     \LWR@htmltag{header}\LWR@orignewline
8121     \LWR@startpars
8122     \LWR@firstpagetop
8123     \LWR@stoppars
8124     \LWR@htmltag{/header}\LWR@orignewline
8125 }%
```

```
8126 \LWR@htmlelementclass{div}{bodywithoutsidetoc}
8127 \LWR@htmlelementclass{main}{bodycontainer}
8128 \LWR@traceinfo{LWR@lwarpStart: Generating textbody.}
8129 \LWR@htmlelementclass{section}{textbody}
```

Create a label for previous/next links, and remember it has been done:

```
8130 \booltrue{LWR@setseqfilelabel}%
8131 \label{\BaseJobname-autofile-\arabic{LWR@htmlseqfilenumber}}
```

Patch the itemize, enumerate, and description environments and \item. This works with the native LaTeX environments, as well as those provided by enumitem, enumerate, and paralist.

```
8132 \LWR@patchlists
```

Ensure that math mode is active to call lwarp's patches:

```
8133 \catcode`\$=\active
```

Required for \nameref to work with sᴠɢ math:

```
8134 \immediate\write\@mainaux{\catcode`\string$\active}%
8135 \LetLtxMacro\LWR@syntaxhighlightone$% balance for editor syntax highlighting
```

Allow ʜᴛᴍʟ paragraphs to begin:

```
8136 \LWR@startpars
```

If using MᴀᴛʜJᴀx, disable \ensuremath by printing a nullified definition at the start of each file, and add further customizations:

```
8137 \ifbool{mathjax}{
8138     \typeout{---}
8139     \typeout{Package lwarp:}
8140     \typeout{Processing MathJax customizations for the first HTML page.}
8141     \typeout{Later HTML pages will take the same amount of time.}
8142   \typeout{If this takes too long, see the Lwarp manual regarding customizing MathJax.}
8143 }{}
8144
8145 \LWR@customizeMathJax
8146
8147 \ifbool{mathjax}{
8148     \typeout{Done.}
8149     \typeout{---}
8150 }{}
```

First autopage label in case a figure occurs early before the first section: A new autopage label may be generated for LWR@currentautosecpage for the start of the section, and also for the current page if it is different due to an sᴠɢ image in the section name. Also, the final page after the section has been created is updated in LWR@currentautosecfloatpage.

```
8151 \setcounter{LWR@currentautosecfloatpage}{\value{page}}%
8152 \LWR@newautopagelabel{LWR@currentautosecpage}%
```

```
8153 \LWR@traceinfo{LWR@lwarpStart: done}
8154 }
8155 \catcode`\$=3% math shift until lwarp starts
```

```
8156 \end{warpHTML}
```

69   Ending ʜᴛᴍʟ output

8157 \begin{warpHTML}

\LWR@requesttoc {⟨*boolean*⟩} {⟨*suffix*⟩} Requests that a ᴛᴏᴄ, ʟᴏꜰ, or ʟᴏᴛbe generated.

```
8158 \newcommand*\LWR@requesttoc[2]{%
8159 \ifbool{#1}
8160 {
8161     \expandafter\newwrite\@nameuse{tf@#2}
8162     \immediate\openout \@nameuse{tf@#2} \jobname.#2\relax
8163 }{}
8164 }
```

\LWR@LwarpEnd   Final stop of all HTML output:

```
8165 \newcommand*{\LWR@LwarpEnd}
8166 {
8167 \LWR@stoppars
8168 \LWR@closeprevious{finished}
```

At the bottom of the ending file:

Print any pending footnotes:

```
8169 \LWR@printpendingfootnotes
```

Close the textbody.

(The \LWR@orignobreakspace is in case no autopage is required for the label, which would not print anything, and something must be printed before the newline.)

```
8170 \label{\BaseJobname-autofile-last}\LWR@orignobreakspace\LWR@orignewline
```

```
8171 \LWR@htmlelementclassend{section}{textbody}
8172 \LWR@htmlelementclassend{main}{bodycontainer}
8173 \LWR@htmlelementclassend{div}{bodyandsidetoc}
```

Create the footer if not EPUB

```
8174 \ifbool{FormatEPUB}{}{\LWR@createfooter}
```

No bottom navigation if are finishing the home page, or if formatting for an EPUB or word processor.

Presumably has a table-of-contents.

```
8175 \ifthenelse{\boolean{FormatEPUB}\OR\boolean{FormatWP}}
8176     {}
8177     {
8178         \ifnumcomp{\value{LWR@htmlfilenumber}}{>}{0}{\LWR@botnavigation}{}
8179     }
```

```
8180 \LWR@stoppars% final stop of all paragraphs
```

Finish the HTML file:

```
8181 \LWR@htmltag{/body}\LWR@orignewline
8182 \LWR@htmltag{/html}\LWR@orignewline
```

Seems to be required sometimes:

```
8183 \LWR@maybe@orignewpage
8184 }
```

enddocument/info (*Hook*)   Used to close the *-images.txt file.
        [LaTeX]

   \enddocument   If labels have not changed, mark successful completion of the lateximages.txt file. Executed as everything is being shut down.

For the newer kernel hooks, see **texdoc lthooks-doc** and **texdoc ltshipout-doc**.

```
8185 \ifdef{\AddToHook}{% newer kernel
8186     \AddToHook{enddocument/info}{%
8187         \if@filesw
8188         \ifx \@multiplelabels \relax
8189          \if@tempswa
```

This is where warnings of duplicate labels would appear.

```
8190          \else
```

No duplicate labels, so safe to create images.

```
8191             \immediate\write\LWR@lateximagesfile{%
8192                 |end|end|end|%
8193             }%
8194          \fi
8195        \fi\fi
8196     }
8197 }% newer kernel
8198 {% older kernel
8199     \xpatchcmd{\enddocument}
8200        {%
8201         \if@tempswa
8202         \@latex@warning@no@line{Label(s) may have changed.
8203         Rerun to get cross-references right}%
8204         \fi
8205        }
8206        {%
8207         \if@tempswa
8208             \@latex@warning@no@line{Label(s) may have changed.
8209             Rerun to get cross-references right}%
8210         \else
```

No duplicate labels, so safe to create images.

```
8211             \immediate\write\LWR@lateximagesfile{%
8212                 |end|end|end|%
8213             }%
8214         \fi
8215        }
8216        {}
8217        {
8218         \AtEndDocument{
8219             \PackageWarningNoLine{lwarp}
8220             {%
8221                 Could not patch \protect\enddocument.\MessageBreak
8222             If labels have changed, be sure to recompile before\MessageBreak
8223                 creating lateximages with\MessageBreak
8224                 \space\space lwarpmk limages,\MessageBreak
8225                 or the images may be corrupt%
8226             }
8227         }
8228        }
8229 }% older kernel
```

70 Nullifying foreground/background hooks

See **texdoc lthooks-doc** and **textdoc ltshipout-doc**.

shipout/background (*Hook*)
[LaTeX]

Nullified.

shipout/foreground (*Hook*)
[LaTeX]

Nullified.

```
8230 \ifdef{\RemoveFromHook}{
8231   \AfterEndPreamble{
8232     \IfHookEmptyTF{shipout/background}{}{
8233       \PackageInfo{lwarp}{Removing background hook}
8234       \RemoveFromHook{shipout/background}[*]
8235     }
8236     \IfHookEmptyTF{shipout/foreground}{}{
8237       \PackageInfo{lwarp}{Removing foreground hook}
8238       \RemoveFromHook{shipout/foreground}[*]
8239     }
8240   }
8241 }{}
```

```
8242 \end{warpHTML}
```

71 Title page

package support
⚠ load order

lwarp supports the native LaTeX titling commands, and also supports the packages authblk and titling. If both are used, authblk should be loaded before titling.

\published and \subtitle

If using the titling package, additional titlepage fields for \published and \subtitle may be added by using \AddSubtitlePublished in the preamble. See section 71.8.

affiliation

lwarp provides for the \author macro an additional \affiliation macro to provide an affiliation and other additional information for each author in the title page. The affiliation information is removed when using titlingpage's \theauthor in the main text.

reusing titlepage information

The titling package maintains the definitions of \thetitle, \theauthor, etc., after the title has been typeset. These commands are to be used to refer to the document's title and author, etc., in the main text. These definitions have the \thanks and \affiliation removed, and for \author the \and is replaced to generate a simple inline list of authors separated by commas. Note: \theauthor does not

⚠ \theauthor, authblk

work well with authblk unless the traditional LaTeX syntax is used.

custom titlepages

\printtitle, \printauthor, etc., are provided for use inside a custom titlepage or titlingpage environment, and these retain the \thanks and \affiliation.

\printthanks

\printthanks has been added to force the printing of thanks inside a titlingpage environment when \maketitle is not used.

⚠ \thanks

Inside a \titlepage or \titlingpage environment, use \thanks instead of \footnote for acknowledgements, etc.

71.1   Setting the title, etc.

The following provide setting commands for both HTML and print outputs.

\author
\and
{⟨*author*⟩} While using \maketitle and print mode, the author is treated as a single-column tabular and the \and feature finishes the current tabular then starts a new one for the next author. Each author thus is placed into its own tabular, and an affiliation may be placed on its own line such as

\author{Name \\ Affiliation \and Second Name \\ Second Affiliation}

For HTML, the entire author block is placed inside a <div> of class author, and each individual author is inside a <div> of class oneauthor.

\@title
\@author
\@date
\@title, \@author, and \@date store the values as originally assigned, including any \thanks, \and, or \affiliation. These are low-level macros intended to be used by other macros only inside a titlepage or titlingpage, and are used by \maketitle. The author is printed inside a single-column tabular, which becomes multiple single-column tabulars if multiples authors are included. For HTML, these tabulars become side-by-side <div>s of class oneauthor, all of which are combined into one <div> of class author.

\printtitle
\printauthor
\printdate
\printtitle, etc. are user-level macros intended to be used in custom titlepage or titlingpage environments in cases where \maketitle is not desired. These commands preserve the \thanks, etc., and should not be used in the main text.

\thetitle
\theauthor
\thedate
\HTMLPageBottom
\thetitle, \theauthor, and \thedate are available if **titling** has been loaded, and are sanitized user-level versions from which have been removed the \thanks and \affiliation, and \and is changed for inline text usage. The author is printed inline without \affiliation or \thanks, with \and placing commas between multiple authors. Thus, these commands are to be used in the main text whenever the user wishes to refer to the document's title and such. One practical use for this is to place the authors at the bottom of each HTML page, such as:

```
\HTMLPageBottom{
  \begin{center}\textcopyright~20xx \theauthor\end{center}
}
```

⚠ \theauthor, authblk   \theauthor does not work well if **authblk** is used. If \theauthor is important, it is recommended to use the standard LATEX syntax for \author, optionally with **lwarp**'s \affiliation macro as well.

⚠   affiliations   After \maketitle has completed, \theauthor retains the definition of the author, but \and is changed to become a comma and a space, intending to print the authors names separated by spaces. This fails when affiliations are included on their own table rows.

\affiliation   A solution, provide here, is to define a macro \affiliation which, during \maketitle, starts a new row and adds the affiliation, but after \maketitle is finished \affiliation is re-defined to discard its argument, thus printing only the author names when \author is later used inline.

71.2   \if@titlepage

for HTML & PRINT: 8243 \begin{warpall}

\if@titlepage Some classes do not provide \if@titlepage. In this case, provide it and force it
false.

```
8244 \ifcsvoid{@titlepagefalse}{
8245     \newif\if@titlepage
8246     \@titlepagefalse
8247 }{}
```

```
8248 \end{warpall}
```

71.3 Changes for \affiliation

\affiliation {⟨*text*⟩}

Adds the affiliation to the author for use in \maketitle.

Inside titlepage, this macro prints its argument. Outside, it is null.

for HTML & PRINT:
```
8249 \begin{warpall}
8250 \providerobustcmd{\affiliation}[1]{}
8251 \end{warpall}
```

for PRINT output:
```
8252 \begin{warpprint}
```

```
8253 \AtBeginEnvironment{titlepage}{
8254 \renewrobustcmd{\affiliation}[1]{\\ \textsc{\small#1}}
8255 }
8256
8257 \AtBeginDocument{
8258 \IfPackageLoadedTF{titling}{
8259 \AtBeginEnvironment{titlingpage}{
8260 \renewrobustcmd{\affiliation}[1]{\\ \textsc{\small#1}}
8261 }
8262 }{}% titling loaded
8263 }% AtBeginDocument
```

```
8264 \end{warpprint}
```

for HTML output:
```
8265 \begin{warpHTML}
```

titlepage (*env.*) Sets up a <div> of class titlepage. Provided even for memoir class, since it is
used by \maketitle.

```
8266 \DeclareDocumentEnvironment{titlepage}{}
8267 {%
8268     \renewrobustcmd{\affiliation}[1]{\\ \InlineClass{affiliation}{##1}}%
8269     \LWR@printpendingfootnotes
8270     \LWR@forcenewpage
8271     \BlockClass{titlepage}
8272 }
8273 {
8274     \endBlockClass
8275     \LWR@printpendingfootnotes
8276 }
```

```
8277 \end{warpHTML}
```

71.4 Printing the thanks

\printthanks Forces the \thanks to be printed. This is necessary in a titlingpage environment when \maketitle was not used.

for PRINT output:
```
8278 \begin{warpprint}
8279 \newcommand*{\printthanks}{\@thanks}
8280 \end{warpprint}
```

for HTML output:
```
8281 \begin{warpHTML}
8282 \newcommand*{\printthanks}{\LWR@stoppars\@thanks\LWR@startpars}
8283 \end{warpHTML}
```

71.5 Printing the title, etc. in HTML

The following are for printing the title, etc. in a titlepage or a titlingpage in HTML:

for HTML output:
```
8284 \begin{warpHTML}
```

\printtitle

```
8285 \newcommand*{\printtitle}
8286 {%
8287    \LWR@stoppars%
8288    \LWR@htmltag{\LWR@tagtitle}%
8289    \@title%
8290    \LWR@htmltag{\LWR@tagtitleend}%
8291    \LWR@startpars%
8292 }
```

\LWR@printthetitle A private version which prints the title without footnotes, used to title each HTML page.

```
8293 \newcommand*{\LWR@printthetitle}
8294 {%
8295    \LWR@stoppars%
8296    \LWR@htmltag{\LWR@tagtitle}%
8297    \thetitle%
8298    \LWR@htmltag{\LWR@tagtitleend}%
8299    \LWR@startpars%
8300 }
```

\printauthor HTML version.

```
8301 \newcommand*{\printauthor}{
```

The entire author block is contained in a <div> named author:

```
8302 \begin{BlockClass}{author}
```

\and finishes one author and starts the next:

```
8303 \renewcommand{\and}{%
```

```
8304 \end{BlockClass}
8305 \begin{BlockClass}{oneauthor}
8306 }
```

Individual authors are contained in a <div> named oneauthor:

```
8307 \begin{BlockClass}{oneauthor}
8308 \@author
8309 \end{BlockClass}
8310 \end{BlockClass}
8311 }
```

\printdate

```
8312 \newcommand*{\printdate}{%
8313 \begin{BlockClass}{titledate}
8314 \@date
8315 \end{BlockClass}
8316 }
```

```
8317 \end{warpHTML}
```

71.6   Printing the title, etc. in print form

The following are for printing the title, etc. in a titlepage or a titlingpage in print form:

for PRINT output: `8318 \begin{warpprint}`

\printtitle

```
8319 \newcommand*{\printtitle}{{\Huge\@title}}
```

\printauthor   Print mode.

```
8320 \newcommand*{\printauthor}
8321     {{\large\begin{tabular}[t]{c}\@author\end{tabular}}}
```

\printdate

```
8322 \newcommand*{\printdate}{{\small\textit{\@date}}}
```

```
8323 \end{warpprint}
```

71.7   \maketitle for HTML output

An HTML <div> of class titlepage is used.

\thanks are a form of footnotes used in the title page. See section 62 for other kinds of footnotes.

See \thanksmarkseries{series}, below, to set the style of the footnote marks.

for HTML output: `8324 \begin{warpHTML}`

```
8325 \IfClassLoadedTF{memoir}
8326 {
8327 \newcommand{\LWR@setfootnoteseries}{%
8328     \renewcommand\thefootnote{\@arabic\c@footnote}%
8329 }
8330 }{% not memoir
8331 \if@titlepage
8332 \newcommand{\LWR@setfootnoteseries}{%
8333     \renewcommand\thefootnote{\@arabic\c@footnote}%
8334 }
8335 \else
8336 \newcommand{\LWR@setfootnoteseries}{%
8337     \renewcommand\thefootnote{\@fnsymbol\c@footnote}%
8338 }
8339 \fi
8340 }% not memoir
```

\LWR@maketitlesetup   Patches \thanks macros.

```
8341 \newcommand*{\LWR@maketitlesetup}{%
```

Redefine the footnote mark:

```
8342 \LWR@setfootnoteseries%
8343 \def\@makefnmark{%
8344     \textsuperscript{\thefootnote}%
8345 }
```

$$\thefootnote \Rightarrow \nameuse{arabic}{footnote}, \text{ or}$$
$$\thefootnote \Rightarrow \nameuse{fnsymbol}{footnote}$$

Redefine the footnote text:

```
8346 \long\def\@makefntext##1{%
```

Make the footnote mark and some extra horizontal space for the tags:

```
8347 \textsuperscript{\@thefnmark}~%
```

$$\makethanksmark \Rightarrow \thanksfootmark \Rightarrow \tamark \Rightarrow$$
$$\@thefnmark \Rightarrow \itshape a \text{ (or similar)}$$

Print the text:

```
8348 {##1}%
8349 }%
8350 }
```

\@fnsymbol {⟨*counter*⟩}

Re-defined to use an HTML entity for the double vertical bar symbol. The original definition used \| which was not being seen by *pdftotext*.

```
8351 \def\LWR@HTML@@fnsymbol#1{%
```

```
8352     \ifcase#1\or *\or
8353     \HTMLentity{dagger}\or
8354     \HTMLentity{Dagger}\or
8355     \HTMLentity{sect}\or
8356     \HTMLentity{para}\or
8357     \HTMLunicode{2016}\or
8358     **\or
8359     \HTMLentity{dagger}\HTMLentity{dagger} \or
8360     \HTMLentity{Dagger}\HTMLentity{Dagger} \else
8361     \@ctrerr\fi%
8362 }
8363 \LWR@formatted{@fnsymbol}
```

\maketitle HTML mode. Creates an HTML titlepage div and typesets the title, etc.

Code from the titling package is adapted, simplified, and modified for HTML output.

The name \LWR@maketitle is used to preserve its definition in case a later package overwrites \maketitle.

```
8364 \newcommand*{\LWR@maketitle}{%
```

An HTML titlepage <div> is used for all classes.

```
8365 \begin{titlepage}
```

Set up special patches:

```
8366 \LWR@maketitlesetup
```

Typeset the title, etc:

```
8367 \@maketitle
```

Immediately generate any \thanks footnotes:

```
8368 \LWR@stoppars\@thanks\LWR@startpars
```

Close the HTML titlepage div and cleanup:

```
8369 \end{titlepage}
8370 \setcounter{footnote}{0}%
8371 \global\let\thanks\relax
8372 \global\let\maketitle\relax
8373 \global\let\@maketitle\relax
8374 \global\let\@thanks\@empty
8375 \global\let\@author\@empty
8376 \global\let\@date\@empty
8377 \global\let\@title\@empty
8378 \global\let\title\relax
8379 \global\let\author\relax
8380 \global\let\date\relax
8381 \global\let\and\relax
8382 }
8383
8384 \LetLtxMacro\maketitle\LWR@maketitle
```

\@maketitle HTML mode. Typesets the title, etc.:

```
8385 \providecommand*{\@maketitle}{}
8386 \renewrobustcmd{\@maketitle}{%
8387     \LWR@stoppars%
8388     \LWR@htmltag{\LWR@tagtitle}%
8389     \@title%
8390     \LWR@htmltag{\LWR@tagtitleend}%
8391     \LWR@startpars%
8392     \begin{BlockClass}{author}%
```

For IEEEtran class:

```
8393     \renewcommand*{\cr}{}%
8394     \renewcommand*{\crcr}{}%
8395     \renewcommand*{\noalign}{}%

8396         \renewcommand{\and}{%
8397             \end{BlockClass}%
8398             \begin{BlockClass}{oneauthor}%
8399         }%
8400         \begin{BlockClass}{oneauthor}%
8401             \@author%
8402         \end{BlockClass}%
8403     \end{BlockClass}%
8404     \begin{BlockClass}{titledate}%
8405     \@date%
8406     \end{BlockClass}%
8407 }
```

\LWR@titlingmaketitle \maketitle for use inside an HTML titlingpage environment.

```
8408 \newcommand*{\LWR@titlingmaketitle}{%
```

Keep pending footnotes out of the title block:

```
8409 \LWR@stoppars\@thanks\LWR@startpars
```

Set up special patches:

```
8410 \LWR@maketitlesetup
```

Typeset the title, etc:

```
8411 \@maketitle
```

Immediately generate any \thanks footnotes:

```
8412 \LWR@stoppars\@thanks\LWR@startpars
8413 }
```

```
8414 \end{warpHTML}
```

71.8 \published and \subtitle

\subtitle and \published To add \subtitle and \published to the titlepage, load the titling package and

use \AddSubtitlePublished in the preamble.

The default lwarp.css has definitions for the published and subtitle classes.

If titling is loaded, \AddSubtitlePublished creates a number of additional macros, and also assigns some of the titling hooks. If titling is not loaded, \AddSubtitlePublished creates null macros.

⚠ titling hooks   Do not use \AddSubtitlePublished if the user has patched the titling hooks for some other reason. Portions are marked \warpprintonly to reduce extra tags in HTML. Similarly, BlockClass has no effect in print mode. Thus, the following may be marked warpall.

for HTML & PRINT:
```
8415 \begin{warpall}
```

\AddSubtitlePublished Adds \published and \subtitle, and related.

```
8416 \newcommand*{\AddSubtitlePublished}{%
8417 \IfPackageLoadedTF{titling}{% yes titling package
8418     \newcommand{\@published}{}%
8419     \newcommand{\published}[1]{\gdef\@published{##1}}%
8420     \renewcommand*{\maketitlehooka}{\printpublished}%
8421     \newcommand*{\printpublished}{%
8422         \warpprintonly{\begin{center}\unskip}%
8423         \begin{BlockClass}{published}%
8424         \warpprintonly{\large\itshape}%
8425         \@published%
8426         \end{BlockClass}%
8427         \warpprintonly{\end{center}}%
8428     }%
8429     \newcommand{\@subtitle}{}%
8430     \newcommand{\subtitle}[1]{\gdef\@subtitle{##1}}%
8431     \renewcommand*{\maketitlehookb}{\printsubtitle}%
8432     \newcommand*{\printsubtitle}{%
8433         \warpprintonly{\begin{center}\unskip}%
8434         \begin{BlockClass}{subtitle}%
8435         \warpprintonly{\Large\itshape}%
8436         \@subtitle%
8437         \end{BlockClass}%
8438         \warpprintonly{\end{center}}%
8439     }%
8440 }% yes titling package
8441 {% no titling package
8442     \def\@published{}%
8443     \DeclareDocumentCommand{\published}{m}{\gdef\@published{##1}}%
8444     \DeclareDocumentCommand{\printpublished}{}{}%
8445     \def\@subtitle{}%
8446     \DeclareDocumentCommand{\subtitle}{m}{\gdef\@subtitle{##1}}%
8447     \DeclareDocumentCommand{\printsubtitle}{}{}%
8448 }% no titling package
8449 }% \AddSubtitlePublished
```

```
8450 \end{warpall}
```

72 Abstract

The following code replaces the LaTeX default, and will itself be replaced later if the abstract package is loaded.

for HTML output: 8451 \begin{warpHTML}

\abstractname User-redefinable title for the abstract.

Also over-written by the babel package.

8452 \providecommand*{\abstractname}{Abstract}

Some classes allow an optional name, so it is allowed here.

abstract (*env.*)

8453 \DeclareDocumentEnvironment{abstract}{O{\abstractname}}
8454 {
8455   \LWR@forcenewpage
8456   \BlockClass{abstract}
8457   \BlockClassSingle{abstracttitle}{#1}
8458 }
8459 {
8460   \endBlockClass
8461 }

8462 \end{warpHTML}

73 Quote and verse

73.1 Attributions

\attribution {⟨*name*⟩}

For use with quote, quotation, verse:

Ex: "A quotation." \attribution{\textsc{Author Name}\\\textsl{Book Title}}

for HTML & PRINT: 8463 \begin{warpall}
8464 \newcommand{\attribution}[1]{
8465   \begin{flushright}
8466   \unskip
8467   #1
8468   \end{flushright}%
8469 }
8470 \end{warpall}

for HTML output: 8471 \begin{warpHTML}
8472 \newcommand{\LWR@HTML@attribution}[1]{%
8473   \LWR@stoppars%
8474   \begin{BlockClass}{attribution}
8475   #1

```
8476        \end{BlockClass}
8477        \LWR@startpars%
8478 }
8479 \LWR@formatted{attribution}
8480 \end{warpHTML}
```

73.2   Quotes, quotations

8481 `\begin{warpHTML}`

quote (*env.*)

```
8482 \newenvironment*{LWR@HTML@quote}
8483 {
8484        \LWR@forcenewpage
8485        \LWR@htmlblocktag{blockquote}
8486 }
8487 {\LWR@htmlblocktag{/blockquote}}
8488
8489 \LWR@formattedenv{quote}
```

quotation (*env.*)

```
8490 \newenvironment*{LWR@HTML@quotation}
8491 {
8492        \LWR@forcenewpage
8493        \LWR@htmlblocktag{blockquote}
8494 }
8495 {\LWR@htmlblocktag{/blockquote}}
8496
8497 \LWR@formattedenv{quotation}
```

```
8498 \end{warpHTML}
```

73.3   Verse

When using verse or memoir, always place a \\ after each line.

\attrib   The documentation for the verse and memoir packages suggest defining an
\attrib command, which may already exist in current documents, but it will
only work for print output. lwarp provides \attribution, which works for both
print and HTML output. To combine the two so that \attrib is used for print and
\attribution is used for HTML:

```
\begin{warpHTML}
\let\attrib\attribution
\end{warpHTML}
```

\vleftskip (*Len*)   These lengths are used by verse and memoir to control the left margin, and they
\vleftmargini (*Len*)   may already be set by the user for print output. New lengths \HTMLvleftskip and
\HTMLvleftskip (*Len*)   \HTMLleftmargini are provided to control the margins in HTML output. These
\HTMLleftmargini (*Len*)   new lengths may be set by the user before any verse environment, and persist
                            until they are manually changed again. One reason to change \HTMLleftmargini
                            is if there is a wide \flagverse in use, such as the word "Chorus", in which case

the value of \HTMLleftmargini should be set to a wide enough length to contain "Chorus". The default is wide enough for a stanza number.

⚠ spacing   Horizontal spacing relies on *pdftotext*'s ability to discern the layout (-layout option) of the text in the HTML-tagged PDF output. For some settings of \HTMLleftmargini or \HTMLleftskip the horizontal alignment may not work out exactly, in which
⚠ verse margin   case a label may be shifted by one space. During translation to HTML, the stanza numbers are kept out of the left margin, which would have caused *pdftotext* to shift everything over.

73.3.1   LaTeX core verse environment

for HTML output: 8499 \begin{warpHTML}

verse (*env.*)

```
8500 \newenvironment{LWR@HTML@verse}
8501                {\let\\\newline% lwarp
8502                 \list{}{\itemsep        \z@
8503                         \itemindent   -1.5em%
8504                         \listparindent\itemindent
8505                         \rightmargin  \leftmargin
8506                         \advance\leftmargin 1.5em}%
8507                 \item\relax}
8508                {\endlist}
8509
8510 \LWR@formattedenv{verse}
8511 \end{warpHTML}
```

for HTML & PRINT: 8512 \begin{warpall}

73.3.2   verse and memoir

The following lengths are used by verse and memoir. They may be set in either print or HTML output, but are only used in HTML. This allows the user to set \vleftskip and \leftmargini for print output, and optionally select different values for HTML.

\HTMLvleftskip (*Len*)   Sets \vleftskip inside a verse environment in HTML.

```
8513 \newlength{\HTMLvleftskip}
8514 \setlength{\HTMLvleftskip}{1em}
```

\HTMLleftmargini (*Len*)   Sets \leftmargini inside a verse environment in HTML.

```
8515 \newlength{\HTMLleftmargini}
8516 \setlength{\HTMLleftmargini}{4.5em}
```

```
8517 \end{warpall}
```

74   Verbatim and tabbing

for HTML & PRINT: 8518 `\begin{warpall}`

\VerbatimHTMLWidth (*Len*)   Width to use in HTML Verbatim environment.

This width is used when placing line numbers to the right. Ignored during print output.

8519 `\newlength{\VerbatimHTMLWidth}`
8520 `\setlength{\VerbatimHTMLWidth}{4in}`
8521 `\end{warpall}`

for HTML output: 8522 `\begin{warpHTML}`

\@setupverbvisiblespace   For X∃TEX or LuaTEX, the default visible space was drawn in PDF, but not a text character which could be copied to HTML.

8523 `\ifxetexorluatex`
8524
8525 `\newcommand*{\LWR@HTML@@setupverbvisiblespace}{\let\@xobeysp\textvisiblespace}`
8526
8527 `\LWR@formatted{@setupverbvisiblespace}`
8528
8529 `\fi`

LWR@verbtags (*bool*)   Used to temporarily turn off verbatim tags while doing `\verbatiminput` in the HTML head, during `\LWR@filestart`, or during MATHJAX. Verbatim tags are also disabled separately inside an HTML span.

8530 `\newbool{LWR@verbtags}`
8531 `\booltrue{LWR@verbtags}`

\verb   Patched to encapsulate the verbatim text inside span with a class of verb.

8532 `\LetLtxMacro\LWR@orig@verb@egroup\verb@egroup`
8533
8534 `\def\LWR@verb@egroup@endspan{%`
8535 `    \LWR@orig@verb@egroup%`
8536 `    \ifbool{LWR@verbtags}%`
8537 `        {\LWR@htmltag{/span}}%`
8538 `        {}%`
8539 `    \endgroup%`
8540 `}`

8541 `\xpretocmd{\verb}`
8542 `    {%`
8543 `        \begingroup%`
8544 `        \ifbool{LWR@verbtags}%`
8545 `            {\LWR@htmltag{span class=\textquotedbl{}verb\textquotedbl}}%`
8546 `            {}%`
8547 `        \let\verb@egroup\LWR@verb@egroup@endspan%`
8548 `    }`
8549 `    {}`
8550 `    {\LWR@patcherror{LaTeX}{verb}}`

\LWR@atbeginverbatim [⟨*1: style*⟩] {⟨*2: class*⟩}

> Encloses a verbatim environment with the given css class.
>
> The use of \textquotedbl instead of " improves compatibility with xeCJK.

```
8551 \newcommand*{\LWR@atbeginverbatim}[2][]
8552 {%
```

> Stop generating HTML paragraph tags:

```
8553 \LWR@stoppars%
```

> Avoid excessive space between lines:

```
8554 \setlength{\parskip}{0ex}%
8555 \setlength{\topsep}{0pt}%
8556 \setlength{\partopsep}{0pt}%
```

> Inside the verbatim, temporarily prevent underfull \hbox warnings.

```
8557          \hbadness=10000\relax%
```

> Create a new pre of the given class. The tags may temporarily be turned off for internal use, such as loading the MATHJAX script, or inside a <span>.

```
8558 \ifbool{LWR@verbtags}%
8559 {%
8560     \ifnumcomp{\value{LWR@spandepth}}{=}{0}{%
8561         \LWR@htmltag{pre class=\textquotedbl#2\textquotedbl%
8562             \ifthenelse{\equal{#1}{}}{}{ style=\textquotedbl#1\textquotedbl}%
8563         }%
8564         \par%
8565     }%
8566     {% in a span
8567         \LWR@spanwarnformat{verbatim}%
8568     }%
8569 }{}%
```

> Use a mono-spaced font to preserve horizontal positioning. If horizontal alignment is important for the user, use a mono-spaced font in the css for the verse class.

```
8570 \begingroup%

8571 \LWR@print@normalfont%
8572 \LWR@origttfamily%
```

> If not inside a lateximage, use a small font to avoid line overflow.

```
8573 \ifnumcomp{\value{LWR@lateximagedepth}}{=}{0}%
8574     {\LWR@print@scriptsize}%
8575     {}%
```

> Since inside a <pre>, restore the original list processing:

```
8576 \LWR@restoreoriglists%
```

Turn off babel-french extra space before punctuation:

8577 \LWR@hook@processingtags%

Do not produce HTML tags for \hspace inside a verse par. Restore plain LaTeX
\hspace functionality:

8578 \let\hspace\LWR@print@hspace%

Do not produce HTML tags for \nbsp.

8579 \boolfalse{LWR@HTMLsanitize@nobreakspace}%
8580 }

\LWR@afterendverbatim Finishes enclosing a verbatim environment.

8581 \newcommand*{\LWR@afterendverbatim}{%
8582 \endgroup%
8583 \par%

At the end of the environment, close the pre:

8584 \ifboolexpr{
8585   bool{LWR@verbtags} and
8586   test {\ifnumcomp{\value{LWR@spandepth}}{=}{0}}
8587 }%
8588 {%
8589   \noindent\LWR@htmltag{/pre}\par% pre
8590 }{}%

Resume regular paragraph handling:

8591 \LWR@startpars%
8592 }

\verbatiminput {⟨filename⟩}

Patch \verbatiminput to add HTML tags:

8593 \newcommand{\LWR@HTML@verbatim@input}[2]{%
8594   \ifbool{LWR@verbtags}{\LWR@forcenewpage}{}%
8595   \LWR@atbeginverbatim{Verbatim}%
8596   \LWR@print@verbatim@input{#1}{#2}%
8597   \LWR@afterendverbatim%
8598 }
8599
8600 \LWR@formatted{verbatim@input}

verbatim (*env.*)

8601 \AfterEndPreamble{
8602 \LWR@traceinfo{Patching verbatim.}
8603 \AtBeginEnvironment{verbatim}{%
8604   \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
8605   {}%
8606   {%

```
8607                \LWR@forcenewpage%
8608                \LWR@atbeginverbatim{verbatim}%
8609           }%
8610 }
8611 \AfterEndEnvironment{verbatim}{%
8612     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
8613        {}%
8614        {%
8615            \LWR@afterendverbatim%
8616        }%
8617 }
8618 %
8619 \AtBeginEnvironment{verbatim*}{%
8620     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
8621        {}%
8622        {%
8623            \LWR@forcenewpage%
8624            \LWR@atbeginverbatim{verbatim}%
8625        }%
8626 }
8627 \AfterEndEnvironment{verbatim*}{%
8628     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
8629        {}%
8630        {%
8631            \LWR@afterendverbatim%
8632        }%
8633 }
8634 }
```

tabbing (*env.*) The tabbing environment works, except that svg math and lateximages do not yet work inside the environment.

⚠ math in tabbing If math is used inside tabbing, place tabbing inside a lateximage environment, which will render the entire environment as a single svg image.

```
8635 \newenvironment*{LWR@HTML@tabbing}
8636 {%
8637     \LWR@forcenewpage%
8638     \LWR@atbeginverbatim{tabbing}%
8639     \let\enskip\LWR@print@enskip%
8640     \let\quad\LWR@print@quad%
8641     \let\qquad\LWR@print@qquad%
8642     \LetLtxMacro~\LWR@origtilde%
8643     \LetLtxMacro\nobreakspace\LWR@orignobreakspace%
8644     \let\,\LWR@origcomma%
8645     \let\thinspace\LWR@print@thinspace%
8646     \let\negthinspace\LWR@print@negthinspace%
8647     \LWR@print@tabbing%
8648 }
8649 {%
8650     \endLWR@print@tabbing%
8651     \LWR@afterendverbatim%
8652 }
8653
8654 \LWR@formattedenv{tabbing}

8655 \end{warpHTML}
```

75 Theorems

\newtheorem {⟨*text*⟩} [⟨*counter*⟩] — *or* — [⟨*oldname*⟩] {⟨*text*⟩}

A few minor changes are made to supply HTML tags.

- The entire theorem is placed into a <div> of class theoremcontents.

- The label for each theorem is placed inside a <span> of class theoremlabel.

- The contents are placed inside a <div> of class theoremcontents.

for HTML output: 8656 \begin{warpHTML}

\@thm {⟨*counter name*⟩} {⟨*text*⟩}

Recent LaTeX kernels use \@kernel@refstepcounter which does not include cleveref changes. Revert to \stepcounter instead. (Print-mode hyperref changes do not matter for HTML.)

```
8657 \VerifyCommand[lwarp][latex]{\@thm}{C35B697D8E37A052BF513D51C8660A87}
8658
8659 \def\@thm#1#2{%
8660 %    \@kernel@refstepcounter{#1}%
8661     \refstepcounter{#1}%              lwarp
8662     \@ifnextchar[{\@ythm{#1}{#2}}{\@xthm{#1}{#2}}%
8663 }
```

\@begintheorem {⟨*name*⟩} {⟨*number*⟩}

```
8664 \renewcommand{\@begintheorem}[2]{%
8665     \LWR@forcenewpage
8666     \LWR@printpendingfootnotes%                          lwarp
8667     \BlockClass{theoremcontents}
8668     \trivlist
8669     \item[\InlineClass{theoremlabel}{#1\ #2\ }]\itshape%
8670 }
```

\@opargbegintheorem {⟨*name*⟩} {⟨*number*⟩} {⟨*oparg*⟩}

LaTeX defines this, but amsthm \relaxes it, so it will not be defined if amsthm is loaded before lwarp.

```
8671 \ifundef{\@opargbegintheorem}{}{
8672     \renewcommand{\@opargbegintheorem}[3]{%
8673         \LWR@forcenewpage
8674         \BlockClass{theoremcontents}
8675         \trivlist
8676         \item[\InlineClass{theoremlabel}{#1\ #2\ (#3)\ }]\itshape%
8677     }
8678 }
```

\@endtheorem

```
8679 \renewcommand*{\@endtheorem}{%
8680 \endtrivlist
```

```
8681    \LWR@printpendingfootnotes%                              lwarp
```

```
8682 \endBlockClass% theoremcontents
8683 }
```

```
8684 \end{warpHTML}
```

76   Lists

The environments itemize, enumerate, and description are patched when lwarp is started. These patches support the standard LaTeX environments, as well as those of enumerate, enumitem, and paralist, and at least the French version of babel. Additional patches are done on a package-specific basis.

The LaTeX source for itemize and enumerate are found in source2e, but the source for description is found in article.cls, etc.

empty item   To have an empty item,use \mbox{} or a trailing backslash. This forces a new line in print output, matching the new line which will appear in HTML output. Ex:

```
begin{itemize}
item \mbox{}
    \begin{itemize}
...
    \end{itemize}
item \
    \begin{itemize}
  ...
    \end{itemize}
```

76.1   List environment

8685 \begin{warpHTML}

\LWR@printcloselist   May be locally redefined by enumerate or description.

```
8686 \newcommand*{\LWR@printcloselist}{\LWR@printcloseitemize}
```

\LWR@printopenlist   May be locally redefined by itemize, enumerate, description, or hanginglist from package hang.

```
8687 \newcommand*{\LWR@printopenlist}{%
8688    ul % space
8689    class=\textquotedbl{}list\textquotedbl{} % space
8690    style=\textquotedbl\LWR@print@mbox{list-style-type:none}\textquotedbl{}%
8691 }
```

\makelabel While inside a list environment, lwarp nullifies a number of TEX horizontal skip and fill commands, allowing the user to define \makelabel for print mode while HTML mode ignores those commands.

⚠ label font When defining \makelabel in a list environment, use \textbf etc. instead of \bfseries.

\@mklab Removes PDF spacing.

```
8692 \AtBeginDocument{
8693 \def\@mklab#1{%
8694 %    \hfil %
8695    #1}
8696 \let\makelabel\@mklab
8697 }
```

\@donoparitem Modified for HTML output by replacing TEX boxes with plain text. Also removes PDF spacing.

```
8698 \def\@donoparitem{%
8699   \@noparitemfalse
8700 %    \global\setbox\@labels\hbox{\hskip -\leftmargin
8701 %                               \unhbox\@labels
8702 %                               \hskip \leftmargin}%
8703 %    \if@minipage\else
8704 %      \@tempskipa\lastskip
8705 %      \vskip -\lastskip
8706 %      \advance\@tempskipa\@outerparskip
8707 %      \advance\@tempskipa -\parskip
8708 %      \vskip\@tempskipa
8709 %    \fi
8710 }
```

\LWR@makelabeltag Used to add <dt> for descriptions. Empty for other list types.

```
8711 \newcommand*{\LWR@makelabeltag}{}
```

\@item Modified for HTML output by replacing TEX boxes with plain text. Also removes PDF spacing.

```
8712 \def\LWR@HTML@item[#1]{%
8713 \LWR@traceinfo{@item}%
8714   \if@noparitem
8715     \@donoparitem
8716   \else
8717 %      \if@inlabel
8718 %        \indent
8719 %      \fi
8720   \ifhmode
8721 %        \unskip\unskip
8722   \fi
8723   \if@newlist
8724     \if@nobreak
8725       \@nbitem
8726     \else
8727 %        \addpenalty\@beginparpenalty
8728 %        \addvspace\@topsep
```

```
8729 %        \addvspace{-\parskip}%
8730 %      \fi
8731    \else
8732 %        \addpenalty\@itempenalty
8733 %        \addvspace\itemsep
8734    \fi
8735    \global\@inlabeltrue
8736  \fi
8737 % \everypar{%
8738   \@minipagefalse
8739   \global\@newlistfalse

8740 %    \if@inlabel
8741 %      \global\@inlabelfalse

8742 %      {\setbox\z@\lastbox
8743 %       \ifvoid\z@
8744 %         \kern-\itemindent
8745 %       \fi}%

8746 %      \box\@labels
8747 %      \penalty\z@
8748 %    \fi

8749 %    \if@nobreak
8750 %      \@nobreakfalse
8751 %      \clubpenalty \@M
8752 %    \else
8753 %      \clubpenalty \@clubpenalty
8754 %      \everypar{}%
8755 %    \fi}%

8756   \if@noitemarg
8757     \@noitemargfalse
8758     \if@nmbrlist

8759        \refstepcounter\@listctr
8760     \fi
8761   \fi
```

If not empty, print the label with the class `listmarker`:

```
8762     \ifboolexpr{
8763       test {\ifblank{#1}} or
8764       (
8765          test {\ifstrequal{#1}{\@itemlabel}} and
8766          test {\ifdefempty{\@itemlabel}}
8767       )
8768    }%
8769       {}%
8770       {%
8771        \ifdefempty{\LWR@makelabeltag}{}{\LWR@htmltag{\LWR@makelabeltag}}%
8772          \InlineClass{listmarker}{\makelabel{#1}}%
8773        \ifdefempty{\LWR@makelabeltag}{}{\LWR@htmltag{/\LWR@makelabeltag}} % extra space
8774       }%
8775 % \sbox\@tempboxa{\makelabel{#1}%
8776 % \global\setbox\@labels\hbox{%
8777 %    \unhbox\@labels
```

```
8778 %      \hskip \itemindent
8779 %      \hskip -\labelwidth
8780 %      \hskip -\labelsep
8781 %      \ifdim \wd\@tempboxa >\labelwidth
8782 %        \box\@tempboxa

8783 %      \else
8784 %        \hbox to\labelwidth {\unhbox\@tempboxa}%
8785 %      \fi
8786 %      \hskip \labelsep}%
8787   \ignorespaces%
8788 }
```

\@nbitem

```
8789 \def\@nbitem{%
8790 %    \@tempskipa\@outerparskip
8791 %    \advance\@tempskipa -\parskip
8792 %    \addvspace\@tempskipa
8793 }
```

\LWR@listitem [⟨*label*⟩]

Handles \item inside a list, itemize, or enumerate.

See \LWR@openparagraph where extra \hspace is used to leave room for the label while inside a list during paragraph construction.

```
8794 \newcommand*{\LWR@listitem}{%
8795     \LWR@stoppars%
8796     \LWR@startnewdepth{listitem}%
8797     \LWR@htmltag{li}%
8798     \LWR@orignewline%
8799     \LWR@startpars%
8800     \LWR@ensuredoingapar%
8801     \LWR@origitem%
8802 }
```

\LWR@nulllistfills Nullifies various TEX fill commands, in case they are used inside \makelabel. Problems are caused when these are nullified all the time.

```
8803 \newcommand*{\LWR@nulllistfills}{%
8804     \renewcommand*{\hss}{}%
8805     \renewcommand*{\llap}[1]{##1}%
8806     \renewcommand*{\rlap}[1]{##1}%
8807     \renewcommand*{\hfil}{}%
8808     \renewcommand*{\hfilneg}{}%
8809     \renewcommand*{\hfill}{}%
8810 }
```

list (*env.*) {⟨*label*⟩} {⟨*commands*⟩}

```
8811 \newcommand*{\LWR@liststart}{%
8812     \LWR@traceinfo{LWR@liststart}%
8813     \LWR@stoppars%
8814     \LWR@pushoneclose{list}%
8815     \LWR@htmltag{\LWR@printopenlist}\LWR@orignewline%
```

```
8816    \LWR@startpars%
8817    \setlength{\topsep}{0pt}%
8818    \setlength{\partopsep}{0pt}%
8819    \setlength{\itemsep}{0pt}%
8820    \setlength{\parsep}{0pt}%
8821    \setlength{\leftmargin}{0pt}%
8822    \setlength{\rightmargin}{0pt}%
8823    \setlength{\listparindent}{0pt}%
8824    \setlength{\itemindent}{0pt}%
8825    \setlength{\labelsep}{1em}%
8826    \LWR@nulllistfills%
8827 }


8828 \newcommand*{\LWR@listend}{%
8829    \LWR@traceinfo{LWR@listend}%
8830    \LWR@stoppars%
8831    \LWR@closeprevious{list}%
8832    \LWR@startpars%
8833 }
```

76.2 Itemize

\LWR@itemizeitem [⟨*label*⟩]

Handles \item inside an itemize or enumerate.

The optional argument is passed to \LWR@origitem.

See \LWR@openparagraph where extra \hspace is used to leave room for the label while inside a list during paragraph construction.

```
8834 \newcommand*{\LWR@itemizeitem}{%
8835    \LWR@stoppars%
8836    \LWR@startnewdepth{listitem}%
8837    \LWR@htmltag{li}%
8838    \LWR@orignewline%
8839    \LWR@startpars%
8840    \LWR@ensuredoingapar%
8841    \LWR@origitem%
8842 }
```

itemize (*env.*) [⟨*options*⟩]

```
8843 \newcommand*{\LWR@itemizestart}{%
8844    \renewcommand*{\LWR@printcloselist}{\LWR@printcloseitemize}%
8845    \renewcommand*{\LWR@printopenlist}{%
8846        ul % space
8847        class=\textquotedbl{}itemize\textquotedbl{} % space
8848       style=\textquotedbl\LWR@print@mbox{list-style-type:none}\textquotedbl{}%
8849    }%
8850    \LetLtxMacro\item\LWR@itemizeitem%
8851    \LWR@nulllistfills%
8852 }
```

76.3 Enumerate

An HTML unordered list is used with customized LATEX-generated labels.

enumerate (*env.*) [⟨*options*⟩]

```
8853 \newcommand*{\LWR@enumeratestart}{%
8854     \renewcommand*{\LWR@printcloselist}{\LWR@printcloseitemize}%
8855     \renewcommand*{\LWR@printopenlist}{%
8856         ul % space
8857         class=\textquotedbl{}enumerate\textquotedbl{} % space
8858         style=\textquotedbl\LWR@print@mbox{list-style-type:none}\textquotedbl{}%
8859     }%
8860     \LetLtxMacro\item\LWR@itemizeitem%
8861     \LWR@nulllistfills%
8862 }
```

76.4 Description

\LWR@descitem [⟨*label*⟩] Handles an \item inside a description.

```
8863 \newcommand*{\LWR@descitem}[1][]{%
8864     \LWR@stoppars%
8865     \LWR@setlatestname{#1}%
8866     \LWR@startnewdepth{descitem}%
```

While creating the label, encase it inside tags and disable \hspace, which is used by the standard classes to add space to the labels.

```
8867     \begingroup%
8868     \renewcommand*{\LWR@makelabeltag}{dt}%
8869     \RenewDocumentCommand{\hspace}{s m}{}%
8870     \LWR@origitem[#1]%
8871     \endgroup%
```

Allow \item without an argument:

```
8872     \leavevmode%

8873     \LWR@orignewline%
8874     \LWR@htmltag{dd}%
8875     \LWR@startpars%
8876 }
```

description (*env.*) [⟨*options*⟩]

Footnotes are modified to correctly parse optional arguments.

```
8877 \newcommand*{\LWR@descriptionstart}{%
8878     \renewcommand*{\LWR@printcloselist}{\LWR@printclosedescription}
8879     \renewcommand*{\LWR@printopenlist}{%
8880         dl % space
8881         class=\textquotedbl{}description\textquotedbl{} % space
8882     }
8883     \LetLtxMacro\item\LWR@descitem%
```

```
8884        \LWR@nulllistfills%
8885 }
```

76.5   Patching the lists

\LWR@patchlists   Patches list environments.

\LWR@patchlists remembers \item as defined by whatever packages have been loaded, then patches the itemize, enumerate, and description environments and \item. This works with the native LaTeX environments, as well as those provided by enumitem, enumerate, and paralist.

```
8886 \newcommand*{\LWR@patchlists}{%
8887     \LetLtxMacro\item\LWR@listitem%
8888     \LetLtxMacro\@item\LWR@HTML@item%
8889     \renewcommand*{\@trivlist}{%
8890         \LWR@traceinfo{@trivlist start}%
8891         \LWR@liststart%
8892         \LWR@orig@trivlist%
8893         \LWR@traceinfo{@trivlist done}%
8894     }%
8895     \renewcommand*{\trivlist}{%
8896         \LWR@traceinfo{trivlist}%
8897         \LWR@origtrivlist%
8898     }%
8899     \renewcommand*{\endtrivlist}{%
8900         \LWR@traceinfo{endtrivlist start}%
8901         \LWR@origendtrivlist\LWR@listend%
8902         \LWR@traceinfo{endtrivlist done}%
8903     }%
8904     \renewcommand*{\itemize}{%
8905         \LWR@itemizestart\LWR@origitemize%
8906     }%
8907     \renewcommand*{\enumerate}{%
8908         \LWR@enumeratestart\LWR@origenumerate%
8909     }%
8910     \renewcommand*{\description}{%
8911         \LWR@descriptionstart\LWR@origdescription%
8912     }%
8913 }
```

\LWR@restoreoriglists   Restores the original trivlist environment.

```
8914 \newcommand*{\LWR@restoreoriglists}{%
8915     \LWR@traceinfo{LWR@restoreoriglists}%
8916     \LetLtxMacro\item\LWR@origitem%
8917     \LetLtxMacro\@item\LWR@orig@item%
8918     \let\@trivlist\LWR@orig@trivlist%
8919     \let\trivlist\LWR@origtrivlist%
8920     \let\endtrivlist\LWR@origendtrivlist%
8921     \LetLtxMacro\itemize\LWR@origitemize%
8922     \LetLtxMacro\enditemize\LWR@endorigitemize%
8923     \LetLtxMacro\enumerate\LWR@origenumerate%
8924     \LetLtxMacro\endenumerate\LWR@endorigenumerate%
8925     \LetLtxMacro\description\LWR@origdescription%
8926     \LetLtxMacro\enddescription\LWR@endorigdescription%
8927     \let\@mklab\LWR@orig@mklab%
```

```
8928     \let\makelabel\LWR@origmakelabel%
8929     \let\@donoparitem\LWR@orig@donoparitem%
8930     \let\@nbitem\LWR@orig@nbitem%
8931 }

8932 \end{warpHTML}
```

77   Tabular

This is arguably the most complicated part of the entire package. Numerous tricks are employed to handle the syntax of the LaTeX core and the various tabular-related packages.

77.1   Limitations

Tabular mostly works as expected, but pay special attention to the following, especially if working with environments, macros inside tabulars, multirows, siunitx S columns, or the packages multirow, longtable, supertabular, or xtab.

Defining macros and environments:

⚠ Misplaced alignment tab character &

- When defining environments or macros which include tabular and instances of the & character, it may be necessary to make & active before the environment or macro is defined, then restore & to its default catcode after, using the following commands. These are are ignored in print mode.

    \StartDefiningTabulars
    <define macros or environments using tabular and & here>
    \StopDefiningTabulars

⚠ floatrow

This includes before and after defining any macro which used \ttabbox from floatrow.

⚠ tabular inside another environment

- When creating a new environment which contains a tabular environment, lwarp's emulation of the tabular does not automatically resume when the containing enviroment ends, resulting in corrupted HTML rows. To fix this, use \ResumeTabular as follows. This is ignored in print mode.

    \StartDefiningTabulars   %   (&   is   used   in   a
    definition)
    \newenvironment{outerenvironment}
    {
      \tabular{cc}
      left & right \\
    }
    {
      \TabularMacro\ResumeTabular
      left & right \\
      \endtabular
    }
    \StopDefiningTabulars

For developers:
- To automate the use of \StartDefiningTabulars and \EndDefiningTabulars, these macros may be embedded inside an HTML environment definition to automatically change the catcode of & before absorbing the arguments. Another environment may be embedded as well.

    ```
    % Does the work after the catcode has been changed:
    \newcommand*{\LWR@HTML@subsomename}[2]{%

      . . .

      \otherenvironmentname  [<args>]  {<args>}  %  for
    example
    }
    % Change catcode before absorbing arguments:
    \newcommand*{\LWR@HTML@somename{%
      \StartDefiningTabulars
      \LWR@HTML@subsomename
    }
    % Change catcode again at the end:
    \newcommand*{\LWR@HTML@endsomename}{%

      . . .

      \endotherenvironmentname   % for example
      \StopDefiningTabulars
    }
    % Combine with the existing print definition:
    \LWR@formattedenv{somename}
    ```

Cell contents:

⚠ macro in a table
- Using a custom macro inside a tabular data cell may result in an extra HTML data cell tag, corrupting the HTML table. To avoid this, use \TabularMacro just before the macro. This is ignored in print mode.

    ```
    \TabularMacro\somemacro & more row contents \\
    ```

Column specifiers:

⚠ math
- Due to the way math is gathered for processing, column specifiers such as >{$}c<{$} do not work with lwarp. Instead, each cell must specify math mode individually.

@ and !
- Only one each of @ and ! is used at each column, and they are used in that order.

\multirow
- In \multirow cells, the print version may have extra instances of <, >, @, and ! cells on the second and later rows in the \multirow which do not appear in the HTML version.

⚠ \newcolumntype
- If \newcolumntype does not work for HTML, add a simplified column type using \HTMLnewcolumntype.

font and alignment
- lwarp detects each of the following, and sets HTML CSS appropriately:

    ```
    >{\centering\arraybackslash}
    >{\raggedright\arraybackslash}
    >{\raggedleft\arraybackslash}
    >{\itshape}
    >{\bfseries}
    >{\bfseries\itshape}
    ```
  These may be used with \newcolumntype, such as:

    ```
    \newcolumntype{P}[1]{>{\centering\arraybackslash}p{#1}}
    ```

Rules:

- Doubled \hlines, \midrules, and vertical rules are supported.

vertical rules
- Vertical rules next to either side of an @ or ! column are displayed on both sides of the column.

width and trim
- Width options are honored. Trim options are converted to rounded top corners. Trim corners are not rounded with @ or ! columns, and full-width rules ignore trim. When given an optional width, each cell is styled to create the custom border. Without an optional width, the entire row is given a class to assign the standard border.

combined rules
- If you wish to use \cmidrule followed by \bottomrule, it may be necessary to use:

      \cmidrule{2-3} \\[-2ex]
      \bottomrule

  The optional -2ex is ignored in HTML, but improves the visual formatting in the print output.

⚠ \warpprintonly
⚠ Misplaced \noalign
- For \toprule and \bottomrule, when combined with a warpprint or warpHTML environment, if a "Misplaced \noalign" error occurs, change

      This & That \endhead

  to

      \warpprintonly{This & That \endhead}

  and likewise with the other \end headings. Keep the \endfirsthead row unchanged, as it is still relevent to HTML output.

Other:

- tabularx ignores the width, but X columns do produce paragraph columns or multicolumns.

longtable headings
- For longtable, place headings and footings which do not apply to HTML inside \warpprintonly{}.

⚠ S columns
- For S columns (from the siunitx package), while producing print output, anything non-numeric must be placed inside { } braces, including commands such as \multirow. While producing HTML output, though, anything placed inside braces is not seen by lwarp's tabular handling algorithm. To resolve this problem, make a copy of the row, with one version for print output, containing the extra braces, and another version for HTML output, without the extra braces, such as:

      \warpprintonly{1 & 2 & {\multirow{2}{2cm}{Text}} & 3 \\}

      \warpHTMLonly{1 & 2 & \multirow{2}{2cm}{Text} & 3 \\}

⚠ tabular inside a <span>
- In LATEX, a tabular may be placed inside a minipage, but in HTML a <table> may not be inside a <span>. If this situation is detected, a warning is printed instructing the user to isolate the <span> using \warpprintonly or the warpprint environment.

for HTML output: 8933 \begin{warpHTML}

77.2 Temporary package-related macros

These macros are temporary placeholders for macros defined by various packages. If the relevent package is not loaded, these placeholders are used instead.

77.2.1 arydshln

Emualated by the original LATEX non-dashed versions.

```
8934 \LetLtxMacro\hdashline\hline
8935 \LetLtxMacro\cdashline\cline
8936 \LetLtxMacro\firsthdashline\hline
8937 \LetLtxMacro\lasthdashline\hline
```

77.3   Token lookahead

Used by \LWR@futurenonspacelet to look at the next token.

\LWR@mynexttoken

```
8938 \newcommand\LWR@mynexttoken\relax
```

\LWR@futurenonspacelet   \futurelet copies the next token then executes a function to analyze it.

\LWR@futurenonspacelet does the same, but ignores intervening spaces and para-graphs.

Based on the booktabs style:

```
8939 \def\LWR@futurenonspacelet#1{\def\LWR@cs{#1}%
8940 \afterassignment\LWR@fnslone\let\nexttoken= }
8941
8942 \def\LWR@fnslone{\expandafter\futurelet\LWR@cs\LWR@fnsltwo}
8943
8944 \def\LWR@fnsltwo{%
8945     \expandafter\ifx\LWR@cs\@sptoken%
8946         \let\next=\LWR@fnslthree%
8947     \else%
8948         \expandafter\ifx\LWR@cs\par%
8949             \let\next=\LWR@fnslthree%
8950         \else%
8951             \let\next=\nexttoken%
8952         \fi%
8953     \fi\next}
8954
8955 \def\LWR@fnslthree{\afterassignment\LWR@fnslone\let\next= }
```

\LWR@getmynexttoken   Looks ahead and copies the next token into \LWR@mynexttoken.

```
8956 \newcommand*{\LWR@getmynexttoken}{%
8957     \LWR@traceinfo{LWR@getmynexttoken}%
```

⚠   Nothing must follow this next line:

```
8958     \LWR@futurenonspacelet\LWR@mynexttoken\LWR@tabledatacolumntag
8959 }
```

77.4   Tabular variables

In order to support nested tabulars, each of these is used locally. For local counters, **etoolbox**'s \defcounter and **lwarp**'s new \defaddtocounter are used.

LWR@startedrow (*bool*)   True if should print a row tag before this column.

```
8960 \newbool{LWR@startedrow}
8961 \boolfalse{LWR@startedrow}
```

LWR@tabularcelladded (*bool*)   True if have added a data cell for this position.

```
8962 \newbool{LWR@tabularcelladded}
8963 \boolfalse{LWR@tabularcelladded}
```

LWR@hlines (*Ctr*)   Number of \hlines or \midrules above the next row.

```
8964 \newcounter{LWR@hlines}
```

LWR@hdashedlines (*Ctr*)   Number of **arydshln** dashed lines above the next row.

```
8965 \newcounter{LWR@hdashedlines}
```

LWR@doingtbrule (*bool*)   True if the next row will have a top/bottom rule above it.

```
8966 \newbool{LWR@doingtbrule}
8967 \boolfalse{LWR@doingtbrule}
```

LWR@doingcmidrule (*bool*)   True if the next row will have a cmidrule above it.

This is used by \LWR@tabularfinishrow to force a final empty row to create the border for the \cmidrule.

```
8968 \newbool{LWR@doingcmidrule}
8969 \boolfalse{LWR@doingcmidrule}
```

LWR@tableparcell (*bool*)   True if are handling a paragraph inside a table cell, so must close the paragraph tag before moving on.

```
8970 \newbool{LWR@tableparcell}
```

LWR@skippingmrowcell (*bool*)   True if are doing an empty \multirow cell, and thus there is no data tag to close.

```
8971 \newbool{LWR@skippingmrowcell}
```

LWR@skippingmcolrowcell (*bool*)   True if are doing an empty \multicolumnrow cell, and thus there is no data tag to close, and do not print @ and ! columns.

```
8972 \newbool{LWR@skippingmcolrowcell}
```

LWR@usedmultirow (*bool*)   Used to error if used \multirow or \multicolumnrow without using \mrowcell or \mcolrowcell.

```
8973 \newbool{LWR@usedmultirow}
```

LWR@foundmrowcell (*bool*)   Used to error if used \multirow or \multicolumnrow without using \mrowcell or

\mcolrowcell.

8974 `\newbool{LWR@foundmrowcell}`

LWR@skipatbang (*bool*)   True if just finished a \multicolumn so should not create the trailing @ or ! columns table data cells.

8975 `\newbool{LWR@skipatbang}`

LWR@emptyatbang (*bool*)   True if finishing a row and should print empty @ or ! column table data cells.

8976 `\newbool{LWR@emptyatbang}`

LWR@intabularmetadata (*bool*)   True if are in a tabular but not in a data cell. Used to prevent extra HTML breaks if not inside table data.

8977 `\newbool{LWR@intabularmetadata}`
8978 `\boolfalse{LWR@intabularmetadata}`

LWR@exitingtabular (*bool*)   When \end is found, turns off the next opening data tag.

8979 `\newbool{LWR@exitingtabular}`

LWR@tabularmutemods (*bool*)   Mutes HTML output for @, !, < and >.

This is used while printing the final row to generate \bottomrules.

8980 `\newbool{LWR@tabularmutemods}`

LWR@tabularfinalrow (*bool*)   Used to set `aria-hidden` if adding a final row for the purpose of adding the bottom border.

8981 `\newbool{LWR@tabularfinalrow}`

LWR@validtablecol (*bool*)   True if found a valid table column type.

8982 `\newbool{LWR@validtablecol}`

LWR@opttablecol (*bool*)   True if found a table column optional argument.

8983 `\newbool{LWR@opttablecol}`

Used to add a style to a table data cell:

8984 `\newbool{LWR@tdhavecellstyle}`

LWR@tabularDepth (*Ctr*)   Tracks whether & is being used inside a tabular.

8985 `\newcounter{LWR@tabulardepth}`
8986 `\setcounter{LWR@tabulardepth}{0}`

LWR@tabularpardepth (*Ctr*)   Tracks whether should look ahead at the next token when encountering a \par while processing tabular contents.

When LWR@tabularpardepth is deeper than LWR@tabulardepth then lwarp has started looking at the contents of the tabular, and thus any \pars encountered must be followed by another token lookahead.

```
8987 \newcounter{LWR@tabularpardepth}
8988 \setcounter{LWR@tabularpardepth}{0}
```

```
8989 \newcommand*{\LWR@colsresult}{}%temp storage for column format results
8990 \newcommand*{\LWR@pposition}{}
8991 \newcommand*{\LWR@pleft}{}
8992 \newcommand*{\LWR@pright}{}
```

LWR@tablecolspec   Holds the parsed column specification, of total width LWR@tabletotalLaTeXcols, not counting @ and ! columns.

Will contain a string such as llrrccpc, exactly one letter per LaTeX table column, without @, !, >, <, or the vertical bar.

\LWR@strresult   Holds the result of Str functions.

```
8993 \providecommand*{\LWR@strresult}{}
8994 \providecommand*{\LWR@strresulttwo}{}
```

\LWR@origcolspec   Holds the original column specs given to tabular.

```
8995 \newcommand*{\LWR@origcolspec}{}
```

LWR@tablecolspecwidth (Ctr)   Holds the number of tokens in the table columns specification.

This is includes one for each @, !, <, > column, and also one for each of the parameters of p, @, !, <, > columns, and three for each D column.

(This is not the total # of LaTeX columns in the table.)

```
8996 \newcounter{LWR@tablecolspecwidth}
```

LWR@tablecolspecindex (Ctr)   While parsing the LaTeX table column specification, starts at 1 and is incremented per token of the specification.

```
8997 \newcounter{LWR@tablecolspecindex}
```

LWR@tableLaTeXcolindex (Ctr)   While producing the table, resets to 1 at the start of the table and also at each end of line, and is incremented by 1 by each ampersand.

```
8998 \newcounter{LWR@tableLaTeXcolindex}
```

LWR@tabletotalLaTeXcols (Ctr)   While parsing a table column specification, begins at 0 and increments by 1 per LaTeX table column. Eventually holds the final number of LaTeX table columns in each row, not counting @ and ! columns. (In HTML, @ and ! cells become their own columns, but are not included in LWR@tabletotalLaTeXcols.)

```
8999 \newcounter{LWR@tabletotalLaTeXcols}
```

LWR@tabletotalLaTeXcolsnext (Ctr)   Holds the next LaTeX table column index while parsing, equal to one more than LWR@tabletotalLaTeXcols.

```
9000 \newcounter{LWR@tabletotalLaTeXcolsnext}
```

LWR@colatspec   A data array of specifications for @ columns. The leftmost's index is leftedge, the

others are counter values. See section 44.

LWR@colbangspec A data array of specifications for ! columns. The leftmost's index is leftedge, the others are counter values. See section 44.

LWR@colbeforespec A data array of specifications for > columns.

LWR@colafterspec A data array of specifications for < columns.

LWR@colbarspec A data array of specifications for vertical rules.

LWR@coladdclass A data array of extra css class, as set by >.

LWR@cellcolordepth (*Ctr*) Counts how many cell color <div>s were added to the current tabular data cell.

```
9001 \newcounter{LWR@cellcolordepth}
```

77.4.1 Multicolumn variables

```
9002 \newcounter{LWR@tablemulticolswidth}
```

Indexes into the multicolumn specification:

```
9003 \newcounter{LWR@tablemulticolspos}
```

Remembers multicolumn vertical rules if found in the column spec.

```
9004 \newcounter{LWR@mcolvertbarsl}
9005 \newcounter{LWR@mcolvertbarsr}
9006 \newcounter{LWR@mcolvertbarsldash}
9007 \newcounter{LWR@mcolvertbarsrdash}
9008 \newbool{LWR@mcolvertbaronleft}
```

77.4.2 Longtable variables

LWR@starredlongtable (*bool*) Per the caption package, step the counter if longtable*.

```
9009 \newbool{LWR@starredlongtable}
9010 \boolfalse{LWR@starredlongtable}
```

77.4.3 Midrule variables

LWR@midrulecounter (*Ctr*) Indexes across the LWR@midrules and LWR@trim<l/r>rules data arrays.

```
9011 \newcounter{LWR@midrulecounter}
```

77.5 Handling &, @, !, and bar

For technical discussion regarding problems redefining \&, See:
[http://tex.stackexchange.com/questions/11638/
where-do-i-find-futurelets-nasty-behaviour-documented/11860#11860](http://tex.stackexchange.com/questions/11638/where-do-i-find-futurelets-nasty-behaviour-documented/11860#11860)

\LWR@insertatbangcols

```
9012 \newcommand*{\LWR@insertatbangcols}{%
9013     \ifbool{LWR@skipatbang}%
9014     {}%
9015     {%
9016         \LWR@printatbang{at}{\arabic{LWR@tableLaTeXcolindex}}%
9017         \LWR@printatbang{bang}{\arabic{LWR@tableLaTeXcolindex}}%
9018     }%
9019 }
```

\LWR@closetabledatacell If LWR@skippingmrowcell or LWR@skippingmcolrowcell then there is no data tag
to close. Otherwise, close any paragraphs, then close the data tag.

```
9020 \newcommand*{\LWR@closetabledatacell}{%
9021     \booltrue{LWR@intabularmetadata}%
9022     \ifbool{LWR@exitingtabular}%
9023     {%

9024         \LWR@stoppars%
9025     }%
9026     {% not exiting tabular
9027      \ifboolexpr{bool{LWR@skippingmrowcell} or bool{LWR@skippingmcolrowcell}}%
9028         {%

9029             \LWR@stoppars%
```

If not skipping a \multicolumnrow cell, insert the @ and ! columns after this non-
existant column.

```
9030             \ifbool{LWR@skippingmcolrowcell}%
9031                 {}%
9032                 {\LWR@insertatbangcols}%
9033         }%
9034         {% not skippingmrowcell
```

Insert any < then any @ and ! column contents, unless muted for the \bottomrule
or a \multicolumn:

```
9035             \unskip%
9036             \ifboolexpr{%
9037                 bool{LWR@tabularmutemods} or
9038                 bool{LWR@skipatbang} or
9039                 bool{LWR@emptyatbang}
9040             }%
9041                 {}%
9042                 {%
9043                     \LWR@getexparray{LWR@colafterspec}%
9044                         {\arabic{LWR@tableLaTeXcolindex}}%
9045                 }%
```

Close paragraphs:

```
9046             \LWR@stoppars%
9047             \boolfalse{LWR@tableparcell}%
```

Close the table data cell.

Close any color `<div>`s.

```
9048          \whileboolexpr{test {\ifnumcomp{\value{LWR@cellcolordepth}}{>}{0}}}{%
9049              \LWR@htmltag{/div}\LWR@orignewline%
9050              \defaddtocounter{LWR@cellcolordepth}{-1}%
9051          }%
```

Skip the @ and ! cells if are closing a multicolumn cell.

```
9052              \leavevmode\unskip\LWR@htmltag{/td}\LWR@orignewline%
9053              \global\booltrue{LWR@tabularcelladded}%
9054              \LWR@insertatbangcols%
9055          }% not skipping mrowcell
9056      }% not exiting tabular
9057      \boolfalse{LWR@skippingmrowcell}%
9058      \boolfalse{LWR@skippingmcolrowcell}%
9059      \boolfalse{LWR@skipatbang}%
```

Color control. Column is set by >{} for each cell, so it must be cleared here.

```
9060      \def\LWR@cellHTMLcolor{}%
9061      \def\LWR@columnHTMLcolor{}%
9062      \defcounter{LWR@cellcolordepth}{0}%
9063 }
```

When not used inside a tabular, & performs its original function as recorded here (with catcode 4).

```
9064 \let\LWR@origampmacro&
```

```
9065 \end{warpHTML}
```

77.5.1   Handling &

`9066 \begin{warpHTML}`

& Will behave depending on whether it is being used inside tabular.

& is redefined to test whether it is inside a tabular environment, in which case it performs special processing for HTML conversion. If not, it behaves normally.

```
9067 \newcommand*{\LWR@tabularampersand}{%
9068      \LWR@traceinfo{LWR@tabularampersand}%
9069      \ifnumcomp{\value{LWR@tabulardepth}}{>}{0}%
9070      {%
```

If not skipping a multirow cell, close the current data cell.

```
9071          \unskip%
9072          \LWR@closetabledatacell%
```

Move to the next column.

```
9073          \defaddtocounter{LWR@tableLaTeXcolindex}{1}%
```

Have not yet added data in this column:

```
9074        \global\boolfalse{LWR@tabularcelladded}%
```

Look at the next token to decide multi or single column data tag.

```
9075        \LWR@getmynexttoken%
9076    }%
```

If not inside a tabular, performs the original action:

```
9077    {%
9078        \LWR@origampmacro%
9079    }%
9080 }
```

& is left with its original catcode for now.

tikz package seems to require & be left alone until after tikz has been loaded. Also, cleveref uses the ampersand in one of its options.

& is made active inside a tabular.

& is left alone when in math alignments.

77.6   Filling an unfinished row

\LWR@tabularfinishrow   Adds empty table cells if necessary to finish the row.

At the end of the table, if any bottom rules are requested then an empty row must be generated to form the borders which show the rules.

```
9081 \newcommand*{\LWR@tabularfinishrow}{%
```

If not exiting the tabular, or doing a rule, or have already started a row, finish this row:

```
9082    \ifboolexpr{%
9083        not bool {LWR@exitingtabular} or%
9084        bool{LWR@doingtbrule} or%
9085        bool{LWR@doingcmidrule} or%
9086        test{\ifnumcomp{\value{LWR@hlines}}{>}{0}} or%
9087        test{\ifnumcomp{\value{LWR@hdashedlines}}{>}{0}} or%
9088        bool{LWR@startedrow}%
9089    }{%
```

Temporarily turn off LWR@exitingtabular so that table data tags will still be generated.

If generating a final row for the \bottomrule borders, turn off the @, !, <, and > column output:

```
9090    \ifbool{LWR@exitingtabular}{%
9091        \booltrue{LWR@tabularmutemods}%
9092    }{%
9093        \boolfalse{LWR@tabularmutemods}%
9094    }%
```

Locally reenable the table data tags until finished with the final row:

```
9095    \boolfalse{LWR@exitingtabular}%
```

Generate table data tags and ampersands until the right edge:

```
9096    \whileboolexpr{%
9097        test {
9098            \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{<}
9099                {\value{LWR@tabletotalLaTeXcols}}
9100        } or %
9101        (%
9102            bool{LWR@intabularmetadata} and%
9103            not bool{LWR@tabularcelladded} and%
9104            test {
9105                \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{=}
9106                    {\value{LWR@tabletotalLaTeXcols}}
9107            }%
9108        )%
9109    }%
9110    {%
9111        \LWR@tabledatasinglecolumntag%
```

The following is essentially \LWR@tabularampersand with LWR@emptyatbang added to empty the following cells:

```
9112        \LWR@closetabledatacell%
9113        \defaddtocounter{LWR@tableLaTeXcolindex}{1}%
9114        \global\boolfalse{LWR@tabularcelladded}%
9115        \booltrue{LWR@emptyatbang}%
```

Starts the next cell:

```
9116        \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{<}
9117                {\value{LWR@tabletotalLaTeXcols}}%
9118            {\LWR@getmynexttoken}%
9119            {}%
9120    }%
```

Reenable the original LWR@exitingtabular to close the entire table:

```
9121    \ifbool{LWR@tabularmutemods}{%
9122        \booltrue{LWR@exitingtabular}%
9123    }{%
9124        \boolfalse{LWR@exitingtabular}%
9125    }%
9126    \boolfalse{LWR@tabularmutemods}%

9127    \boolfalse{LWR@emptyatbang}%
9128    }{}% ifboolexpr
9129 }
```

77.7   Handling \\

Inside tabular, \\ is redefined to \LWR@tabularendofline

Throws away options \\[dim] or *

\LWR@tabularendofline

9130 \NewDocumentCommand{\LWR@tabularendofline}{s o}{%

Finish the row:

9131   \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{<}
9132   {\value{LWR@tabletotalLaTeXcols}}%
9133   {\LWR@tabularfinishrow}%
9134   {\LWR@closetabledatacell}%
9135   \LWR@htmltag{/tr}\LWR@orignewline%

xcolor row color support:

9136   \@rowc@lors%

No longer inside a data cell:

9137   \booltrue{LWR@intabularmetadata}%

Not yet started a table row:

9138   \boolfalse{LWR@startedrow}%

Additional setup:

9139   \defcounter{LWR@hlines}{0}%
9140   \defcounter{LWR@hdashedlines}{0}%
9141   \boolfalse{LWR@doingtbrule}%
9142   \boolfalse{LWR@doingcmidrule}%
9143   \LWR@clearmidrules%

9144   \def\LWR@rowHTMLcolor{}%

Start at first column:

9145   \defcounter{LWR@tableLaTeXcolindex}{1}%

Have not yet added data in this column:

9146   \global\boolfalse{LWR@tabularcelladded}%

Allow TeX to flush the pending paragraph. Not doing so causes a slowdown for very large tables.

9147   \LWR@stoppars%
9148   \LWR@forceemptyline%

Look at the next token to decide between single column data tag or a special case:

9149   \LWR@getmynexttoken%
9150 }

77.8   Looking ahead in the column specifications

\LWR@columnspeclookahead {⟨*offset*⟩}

Looks `offset` tokens ahead in the column specification, setting `\LWR@strresulttwo`.

The `w` column alignment will be seen as a single unit such as `{c}`.

```
9151 \newcommand*{\LWR@columnspeclookahead}[1]{%
9152     \setcounter{LWR@tempcountone}{\value{LWR@tablecolspecindex}}%
9153     \addtocounter{LWR@tempcountone}{#1}%
9154     \fullexpandarg%
9155     \StrChar{\LWR@origcolspec}{\arabic{LWR@tempcountone}}[\LWR@strresulttwo]%
```

Get the contents of the first group in `\LWR@strresulttwo`:

```
9156         \exploregroups%
9157         \StrChar{\LWR@strresulttwo}{1}[\LWR@strresulttwo]%
9158         \noexploregroups%
9159 }
```

77.9   Parsing @, >, <, !, bar columns

Holds the parsed argument for @, >, <, or ! columns:

```
9160 \newcommand*{\LWR@colparameter}{}
```

`\LWR@parseatcolumn` {⟨*this column type*⟩}

Handles @{`text`} columns.

The argument is ignored, but provided for compatibility with `\LWR@parsenormalcolumn`.

```
9161 \newcommand*{\LWR@parseatcolumn}[1]{%
```

Move to the next token after the '@':

```
9162     \LWR@traceinfo{at column}%
9163     \defaddtocounter{LWR@tablecolspecindex}{1}%
```

Read the next token into `\LWR@colparameter`, expanding once:

```
9164     \LWR@traceinfo{about to read the next token:}%
9165     \expandarg%
9166     \StrChar{\LWR@origcolspec}%
9167         {\arabic{LWR@tablecolspecindex}}[\LWR@colparameter]%
9168     \fullexpandarg%
```

Store the result into a data array, expanding once out of `\LWR@colparameter`:

```
9169     \LWR@traceinfo{have now read the next token}%
9170     \ifnumcomp{\value{LWR@tabletotalLaTeXcols}}{=}{0}%
9171     {% left edge of the table:
9172         \LWR@traceinfo{at the left edge}%
9173         \LWR@setexparray{LWR@colatspec}%
9174             {leftedge}%
9175             {\expandafter\@firstofone\LWR@colparameter}%
9176         \LWR@traceinfo{at the left edge: %
9177             \LWR@getexparray{LWR@colatspec}{leftedge}}%
9178     }%
```

```
9179        {% not at the left edge:
9180            \LWR@traceinfo{not at the left edge}%
9181            \LWR@setexparray{LWR@colatspec}%
9182                {\arabic{LWR@tabletotalLaTeXcols}}%
9183                {\expandafter\@firstofone\LWR@colparameter}%
9184            \LWR@traceinfo{at \arabic{LWR@tabletotalLaTeXcols}%
9185            : % space
9186            \LWR@getexparray{LWR@colatspec}{\arabic{LWR@tabletotalLaTeXcols}}}%
9187        }%
9188    \let\LWR@colparameter\relax%
9189    \booltrue{LWR@validtablecol}%
9190 }
```

\LWR@parsebangcolumn {⟨*this column type*⟩} Handles !{text} columns.

The argument is ignored, but provided for compatibility with \LWR@parsenormalcolumn.

```
9191 \newcommand*{\LWR@parsebangcolumn}[1]{%
```

Move to the next token after the '!':

```
9192    \LWR@traceinfo{bang column}%
9193    \defaddtocounter{LWR@tablecolspecindex}{1}%
```

Read the next token into \LWR@colparameter, expanding once:

```
9194    \LWR@traceinfo{about to read the next token:}%
9195    \expandarg%
9196    \StrChar{\LWR@origcolspec}%
9197        {\arabic{LWR@tablecolspecindex}}[\LWR@colparameter]%
9198    \fullexpandarg%
```

Store the result into a data array, expanding once out of \LWR@colparameter:

```
9199    \LWR@traceinfo{have now read the next token}%
9200    \ifnumcomp{\value{LWR@tabletotalLaTeXcols}}{=}{0}%
9201    {% left edge of the table:
9202        \LWR@traceinfo{at the left edge}%
9203        \LWR@setexparray{LWR@colbangspec}%
9204            {leftedge}%
9205            {\expandafter\@firstofone\LWR@colparameter}%
9206    }%
9207    {% not at the left edge:
9208        \LWR@traceinfo{not at the left edge}%
9209        \LWR@setexparray{LWR@colbangspec}%
9210            {\arabic{LWR@tabletotalLaTeXcols}}%
9211            {\expandafter\@firstofone\LWR@colparameter}%
9212     \LWR@traceinfo{bang \arabic{LWR@tabletotalLaTeXcols}: \LWR@colparameter!}%
9213    }%
9214    \let\LWR@colparameter\relax%
9215    \booltrue{LWR@validtablecol}%
9216 }
```

\LWR@checkbeforeaddclass {⟨*compared csname*⟩} {⟨*css class to add*⟩}

```
9217 \newcommand*{\LWR@checkbeforeaddclass}[2]{%
9218    \ifcsstrequal{LWR@tempone}{#1}%
9219        {%
```

```
9220              \LWR@setexparray{LWR@coladdclass}%
9221                    {\arabic{LWR@tabletotalLaTeXcolsnext}}%
9222                    { #2}% space is intentional
9223          }{}%
9224 }
```

\LWR@checkmathcolpar Error if using math in column parameters.

```
9225 \newcommand*{\LWR@checkmathcolpar}{%
9226     \IfSubStr{\detokenize\expandafter{\LWR@colparameter}}{\LWRdollar}%
9227        {%
9228            \PackageError{lwarp}%
9229                {%
9230                Lwarp does not support `$' in column specifiers.\MessageBreak
9231                 Specify `$' math for each cell in the column.\MessageBreak
9232                  Enter 'h' for more info%
9233                }%
9234                {%
9235              For example, replace `>{$}c<{$}' with `c', and then\MessageBreak
9236                    use `$cell contents$' for each cell in the column.%
9237                }%
9238        }{}%
9239 }
```

\LWR@parsebeforecolumn {⟨*this column type*⟩}

Handles >{text} columns.

The argument is ignored, but provided for compatibility with \LWR@parsenormalcolumn.

```
9240 \newcommand*{\LWR@parsebeforecolumn}[1]{%
```

Move to the next token after the '>':

```
9241    \defaddtocounter{LWR@tablecolspecindex}{1}%
```

Read the next token, expanding once into \LWR@colparameter:

```
9242    \expandarg%
9243    \StrChar{\LWR@origcolspec}%
9244         {\arabic{LWR@tablecolspecindex}}[\LWR@colparameter]%
9245    \fullexpandarg%
```

Error if using >{$}, which is not supported by lwarp.

```
9246    \LWR@checkmathcolpar%
```

Store the result into a data array, expanding once out of \LWR@colparameter:

```
9247    \LWR@setexparray{LWR@colbeforespec}%
9248         {\arabic{LWR@tabletotalLaTeXcolsnext}}%
9249         {\expandafter\@firstofone\LWR@colparameter}%
9250 %
9251    \edef\LWR@tempone{\expandafter\@firstofone\LWR@colparameter}%
```

If detect >{\centering\arraybackslash} or related, add a css class.

```
9252    \LWR@checkbeforeaddclass{LWR@detect@centeringarraybackslash}{tdcenter}
9253    \LWR@checkbeforeaddclass{LWR@detect@raggedrightarraybackslash}{tdleft}
9254    \LWR@checkbeforeaddclass{LWR@detect@raggedleftarraybackslash}{tdright}
9255    \LWR@checkbeforeaddclass{LWR@detect@itshape}{tditshape}
9256    \LWR@checkbeforeaddclass{LWR@detect@bfseries}{tdbfseries}
9257    \LWR@checkbeforeaddclass{LWR@detect@bfit}{tdbfit}

9258    \let\LWR@colparameter\relax%
9259    \booltrue{LWR@validtablecol}%
9260 }
```

\LWR@parseaftercolumn {⟨*this column type*⟩}

Handles <{text} columns.

The argument is ignored, but provided for compatibility with \LWR@parsenormalcolumn.

```
9261 \newcommand*{\LWR@parseaftercolumn}[1]{%
```

Move to the next token after the '<':

```
9262    \defaddtocounter{LWR@tablecolspecindex}{1}%
```

Read the next token, expanding once into \LWR@colparameter:

```
9263    \expandarg%
9264    \StrChar{\LWR@origcolspec}%
9265        {\arabic{LWR@tablecolspecindex}}[\LWR@colparameter]%
9266    \fullexpandarg%
```

Error if using >{$}, which is not supported by lwarp.

```
9267    \LWR@checkmathcolpar%
```

Store the result into a data array, expanding once out of \LWR@colparameter:

```
9268    \LWR@setexparray{LWR@colafterspec}%
9269        {\arabic{LWR@tabletotalLaTeXcols}}%
9270        {\expandafter\@firstofone\LWR@colparameter}%
9271    \let\LWR@colparameter\relax%
9272    \booltrue{LWR@validtablecol}%
9273 }
```

\LWR@parsebarcolumn {⟨*this column type*⟩}

Handles vertical rules.

The argument is ignored, but provided for compatibility with \LWR@parsenormalcolumn.

```
9274 \newcommand*{\LWR@parsebarcolumn}[1]{%
9275    \LWR@traceinfo{LWR@parsebarcolumn}%
```

Remember the bar at this position:

```
9276    \ifnumcomp{\value{LWR@tabletotalLaTeXcols}}{=}{0}%
9277    {% left edge of the table:
9278        \edef\LWR@tempone{\LWR@getexparray{LWR@colbarspec}{leftedge}}%
```

```
9279        \ifdefstring{\LWR@tempone}{tvertbarl}%
9280        {\LWR@setexparray{LWR@colbarspec}{leftedge}{tvertbarldouble}}%
9281        {\LWR@setexparray{LWR@colbarspec}{leftedge}{tvertbarl}}%
9282    }%
9283    {% not at the left edge:
9284        \edef\LWR@tempone{%
9285        \LWR@getexparray{LWR@colbarspec}{\arabic{LWR@tabletotalLaTeXcols}}%
9286        }%
9287        \ifdefstring{\LWR@tempone}{tvertbarr}%
9288        {%
9289            \LWR@setexparray{LWR@colbarspec}%
9290                {\arabic{LWR@tabletotalLaTeXcols}}{tvertbarrdouble}%
9291        }%
9292        {%
9293            \LWR@setexparray{LWR@colbarspec}%
9294                {\arabic{LWR@tabletotalLaTeXcols}}{tvertbarr}%
9295        }%
9296    }%
9297    \booltrue{LWR@validtablecol}%
9298 }
```

\LWR@parsecoloncolumn {⟨*this column type*⟩}

> Handles vertical rules.

> The argument is ignored, but provided for compatibility with \LWR@parsenormalcolumn.

```
9299 \newcommand*{\LWR@parsecoloncolumn}[1]{%
9300     \LWR@traceinfo{\LWR@parsecoloncolumn}%
```

> Remember the bar at this position:

```
9301    \ifnumcomp{\value{LWR@tabletotalLaTeXcols}}{=}{0}%
9302    {% left edge of the table:
9303        \edef\LWR@tempone{\LWR@getexparray{LWR@colbarspec}{leftedge}}%
9304        \ifdefstring{\LWR@tempone}{tvertbarldash}%
9305        {\LWR@setexparray{LWR@colbarspec}{leftedge}{tvertbarldoubledash}}%
9306        {\LWR@setexparray{LWR@colbarspec}{leftedge}{tvertbarldash}}%
9307    }%
9308    {% not at the left edge:
9309        \edef\LWR@tempone{%
9310        \LWR@getexparray{LWR@colbarspec}{\arabic{LWR@tabletotalLaTeXcols}}%
9311        }%
9312        \ifdefstring{\LWR@tempone}{tvertbarrdash}%
9313        {\LWR@setexparray{LWR@colbarspec}%
9314            {\arabic{LWR@tabletotalLaTeXcols}}{tvertbarrdoubledash}}%
9315        {\LWR@setexparray{LWR@colbarspec}%
9316            {\arabic{LWR@tabletotalLaTeXcols}}{tvertbarrdash}}%
9317    }%
9318    \booltrue{LWR@validtablecol}%
9319 }
```

\LWR@parsesemicoloncolumn {⟨*this column type*⟩}

> Handles vertical rules.

> The argument is ignored, but provided for compatibility with \LWR@parsenormalcolumn.

The arguments to the column type are absorbed by \LWR@columntype@<char>,
defined by \LWR@modifycolumntype.

```
9320 \newcommand*{\LWR@parsesemicoloncolumn}[1]{%
```

Treat ; as a : column:

```
9321     \LWR@parsecoloncolumn{}%
9322 }
```

77.10   Parsing common column types

\LWR@parsenormalcolumn {⟨*this column type*⟩}

Add to the accumulated column specs, advance counters, and pre-clear another
column of at, before, and after specs.

\newcolumntype definitons use \LWR@parsenormalcolumn, so an HTML and print
version are given so that they may work inside a lateximage.

The arguments to the column type are absorbed by \LWR@columntype@<char>,
defined by \LWR@modifycolumntype.

```
9323 \newcommand*{\LWR@HTML@LWR@parsenormalcolumn}[1]{%
9324     \defaddtocounter{LWR@tabletotalLaTeXcols}{1}%
9325     \defaddtocounter{LWR@tabletotalLaTeXcolsnext}{1}%

9326     \LWR@setexparray{LWR@tablecolspec}{\arabic{LWR@tabletotalLaTeXcols}}{#1}%

9327     \LWR@traceinfo{normal column \arabic{LWR@tabletotalLaTeXcols}: #1}%
9328     \LWR@setexparray{LWR@colatspec}{\arabic{LWR@tabletotalLaTeXcolsnext}}{}%
9329     \LWR@setexparray{LWR@colbangspec}{\arabic{LWR@tabletotalLaTeXcolsnext}}{}%
9330     \LWR@setexparray{LWR@colbeforespec}{\arabic{LWR@tabletotalLaTeXcolsnext}}{}%
9331     \LWR@setexparray{LWR@colafterspec}{\arabic{LWR@tabletotalLaTeXcolsnext}}{}%
9332     \LWR@setexparray{LWR@colbarspec}{\arabic{LWR@tabletotalLaTeXcolsnext}}{}%
9333     \LWR@setexparray{LWR@coladdclass}{\arabic{LWR@tabletotalLaTeXcolsnext}}{}%
9334     \booltrue{LWR@validtablecol}%
9335 }
9336
9337 \newcommand*{\LWR@print@LWR@parsenormalcolumn}[1]{}
9338
9339 \LWR@formatted{LWR@parsenormalcolumn}
```

77.11   Parsing 'w' columns

W and w columns are handled via array with \HTMLnewcolumntype.

77.12   Parsing '*' columns

\LWR@parsestarcolumn {⟨*this column type*⟩} Star columns should already have been expanded, so this
should never be used.

Table 13: Tabular baseline

l	p	m	b	r
			bot	
		mid	bot	
l	par	mid	bot	r
	par	mid		
	par			

The arguments to the column type are absorbed by \LWR@columntype@<char>, defined by \LWR@modifycolumntype.

The argument is ignored, but provided for compatibility with \LWR@parsenormalcolumn.

```
9340 \newcommand*{\LWR@parsestarcolumn}[1]{}
```

77.13   Expanding the star column specifications

\LWR@expandpreamble {⟨*tabular preamble*⟩}

From array \@mkpream.

The resulting expanded preamble is stored in \the\@temptokena. Assign as:

> \edef\destination{\the\@temptokena}

```
9341 \newcommand*{\LWR@expandpreamble}[1]{%
9342     \edef\@tempa{\@temptokena={#1}}%
9343     \@tempa%
9344     \@tempswatrue%
9345     \@whilesw\if@tempswa\fi{%
9346         \@tempswafalse\the\NC@list%
9347     }%
9348 }
```

77.14   Parsing the column specifications

⚠ tabular baselines   HTML CSS cannot exactly match the LaTeX concept of a baseline for a table row. Table 13 shows the LaTeX results for various vertical-alignment choices, with the baseline of the first column drawn across all the columns for comparison. See the p column specification in table 14 for details.

Table 14 describes how each kind of column is converted to HTML.

Table 15 shows the various internal macros generated for each column type.

\LWR@modifycolumntype {⟨*1: column type letter*⟩} {⟨*2: number args to ignore*⟩} {⟨*3: csname of the cell action*⟩} {⟨*4: csname of the multicolumn print type action*⟩} {⟨*5: csname of the multicolumn print data action*⟩}

Table 14: Tabular HTML column conversions

Each cell is given a CSS class of td<columntype>.

---

l, r, c: Converted to table cells without paragraph tags. Uses CSS `vertical-align:middle` so that top or bottom-aligned cells may go above or below this cell.

p: Converted to table cells with paragraph tags. Ref: Table 13, LaTeX places the top line of a parbox aligned with the rest of the text line, so CSS `vertical-align:bottom` is used to have the HTML result appear with the paragraph extending below the L, R, C cells at the middle, if possible. This may be confusing as a P cell may not top-align with an L,R,C cell in the HTML conversion, especially in the presence of a B cell, and two P cells side-by-side will be aligned at the bottom instead of the top. Some adjustment of the CSS may be desired, changing `td.tdp`, `td.tdP`, `td.tdprule`, and `td.tdPrule` to `vertical-align: middle`. Another possibility is to change L,R,C, and P to `vertical-align: top` and not worry about the alignment of B and M cells or trying to approximate LaTeX baselines.

m: With paragraph tags, CSS `vertical-align:middle`.

b: With paragraph tags, CSS `vertical-align:top` so that the bottom of the text is closest to the middle of the text line.

w and W: Converted to `l`, `c`, or `r`. No paragraph tags.

P, M, B: Horizontally-centered versions.

S: Treated as 'c'. Ignores optional argument. From the siunitx package.

D: Treated as 'c'. From the dcolumn package.

@, !, >, <: One each, in that order.

|: Vertical rule.

Unknown: Converted to 'l'.

\newcolumntype: Expands to its replacement text.

\HTMLnewcolumntype: Provides simplified replacement text for HTML.

Table 15: HTML column type internal macros

<coltype>: The single-letter column type, such as c or X.

Created by \LWR@modifycolumntype: Used by lwarp to add HTML functionality to each built-in column type.

    \LWR@columntype@<coltype>: Handles tabular columns depending on the type. Calls \LWR@parsenormalcolumn or related, then advances \LWR@tablecolspecindex.

    \LWR@columntype@mctype@<coltype>: Generates the \multicolumn HTML cell css class. Calls \LWR@printmccoltype@normal or related.

    \LWR@columntype@mcdata@<coltype>: Generates the \multicolumn HTML cell data. Calls \LWR@printmccoldata@normal or related.

Created by \newcolumntype: From array.

    \NC@find@<coltype>: Internally used to parse the column specifier.

    \NC@rewrite@<coltype>: Stores the print-mode replacement text.

Created by \HTMLnewcolumntype: From lwarp.

    \LWR@print@NC@rewrite@<coltype>: Copied from \NC@rewrite@<type>.

    \LWR@HTML@NC@rewrite@<coltype>: Stores the HTML-mode replacement text.

    \NC@rewrite@<coltype>: Redefined to use the print or HTML version.

Add HTML functionality to an existing print version column type.

```
9349 \newcommand*{\LWR@modifycolumntype}[5]{%
9350     \LWR@traceinfo{LWR@modifycolumntype !#1!#2!#3!#4!#5!}%
9351         \LWR@traceinfo{LWR@modifycolumntype #1}%
9352         \edef\@tempa{%
9353             \noexpand\csdef{LWR@columntype@#1}{%
9354                 \noexpand\@nameuse{#3}{#1}%
9355                 \noexpand\defaddtocounter{LWR@tablecolspecindex}{#2}%
9356             }%
9357             \noexpand\csdef{LWR@columntype@mctype@#1}{%
9358                 \noexpand\@nameuse{#4}{#1}%
9359             }%
9360             \noexpand\csdef{LWR@columntype@mcdata@#1}{%
9361                 \noexpand\@nameuse{#5}{#2}%
9362             }%
9363         }%
9364         \@tempa%
9365     \LWR@traceinfo{LWR@modifycolumntype done}%
9366 }


9367 \LWR@modifycolumntype{l}{0}{LWR@parsenormalcolumn}
9368     {LWR@printmccoltype@normal}{LWR@printmccoldata@normal}
9369
9370 \LWR@modifycolumntype{c}{0}{LWR@parsenormalcolumn}
9371     {LWR@printmccoltype@normal}{LWR@printmccoldata@normal}
9372
9373 \LWR@modifycolumntype{r}{0}{LWR@parsenormalcolumn}
```

```
9374        {LWR@printmccoltype@normal}{LWR@printmccoldata@normal}
```

```
9375 \LWR@modifycolumntype{@}{0}{LWR@parseatcolumn}
9376        {LWR@printmccoltype@ignore}{LWR@printmccoldata@other}
9377
9378 \LWR@modifycolumntype{!}{0}{LWR@parsebangcolumn}
9379        {LWR@printmccoltype@ignore}{LWR@printmccoldata@other}
9380
9381 \LWR@modifycolumntype{>}{0}{LWR@parsebeforecolumn}
9382        {LWR@printmccoltype@ignore}{LWR@printmccoldata@other}
9383
9384 \LWR@modifycolumntype{<}{0}{LWR@parseaftercolumn}
9385        {LWR@printmccoltype@ignore}{LWR@printmccoldata@other}
9386
9387 \LWR@modifycolumntype{|}{0}{LWR@parsebarcolumn}
9388        {LWR@printmccoltype@vertbar}{LWR@printmccoldata@skip}
9389
9390 \LWR@modifycolumntype{:}{0}{LWR@parsecoloncolumn}
9391        {LWR@printmccoltype@colon}{LWR@printmccoldata@skip}
9392
9393 \LWR@modifycolumntype{;}{1}{LWR@parsesemicoloncolumn}
9394        {LWR@printmccoltype@semicolon}{LWR@printmccoldata@skip}
```

```
9395 \LWR@modifycolumntype{p}{1}{LWR@parsenormalcolumn}
9396        {LWR@printmccoltype@normal}{LWR@printmccoldata@paragraph}
9397
9398 \LWR@modifycolumntype{m}{1}{LWR@parsenormalcolumn}
9399        {LWR@printmccoltype@normal}{LWR@printmccoldata@paragraph}
9400
9401 \LWR@modifycolumntype{b}{1}{LWR@parsenormalcolumn}
9402        {LWR@printmccoltype@normal}{LWR@printmccoldata@paragraph}
```

A star column:

```
9403 \LWR@modifycolumntype{*}{2}{LWR@parsestarcolumn}
9404        {LWR@printmccoltype@ignore}{LWR@printmccoldata@skip}
```

\HTMLnewcolumntype {⟨*col type*⟩} [⟨*num args*⟩] [⟨*optional arg*⟩] {⟨*replacement text*⟩}

A user-level macro to creates an HTML version of the replacement text for the column type.

This is the equivalent to:

```
\newcommand*{\LWR@HTML@NC@rewrite@<columntype>}[<num args>]
    {\NC@find <replacement text>}
\LWR@formatted{NC@rewrite@<columntype>}
```

```
9405 \NewDocumentCommand{\HTMLnewcolumntype}{m O{0} o m}{%
9406      \IfValueTF{#3}
9407      {
9408          \expandafter\newcommand\expandafter*%
9409              \csname LWR@HTML@NC@rewrite@#1\endcsname[#2][#3]{\NC@find #4}%
9410          \LWR@formatted{NC@rewrite@#1}%
9411      }
9412      {
9413          \expandafter\newcommand\expandafter*%
```

```
9414              \csname LWR@HTML@NC@rewrite@#1\endcsname[#2]{\NC@find #4}%
9415          \LWR@formatted{NC@rewrite@#1}%
9416      }
9417 }
```

```
9418 \end{warpHTML}
```

`9419 \begin{warpprint}`

```
9420 \NewDocumentCommand{\HTMLnewcolumntype}{m O{0} o m}{}
```

```
9421 \end{warpprint}
```

`9422 \begin{warpHTML}`

\LWR@parsetablecols {⟨*colspecs*⟩}

Scans the column specification left to right.

Builds \LWR@tablecolspec with the final specification, one LaTeX column per entry. The final number of LaTeX columns in each row is stored in LWR@tabletotalLaTeXcols, which is the number of & and \\ in each line, but which does not include @, !, <, > specifications in the count.

```
9423 \newcommand*{\LWR@parsetablecols}[1]{%
9424     \LWR@traceinfo{LWR@parsetablecols}%
```

Remember the original supplied column spec:

```
9425     \renewcommand*{\LWR@origcolspec}{#1}%
```

Remove spaces:

```
9426     \expandarg%
9427     \StrSubstitute{\LWR@origcolspec}{ }{}[\LWR@origcolspec]%
```

Expand any star columns:

```
9428     \LWR@expandpreamble{\LWR@origcolspec}%
9429     \edef\LWR@origcolspec{\the\@temptokena}%
```

The parsed column spec data array, LWR@tablecolspec, will be overwritten with new values.

Total number of columns found so far. Also pre-initialize the first several columns of specs:

```
9430     \defcounter{LWR@tabletotalLaTeXcols}{0}%
9431     \defcounter{LWR@tabletotalLaTeXcolsnext}{1}%
9432     \LWR@setexparray{LWR@colatspec}{leftedge}{}%
9433     \LWR@setexparray{LWR@colatspec}{1}{}%
9434     \LWR@setexparray{LWR@colatspec}{2}{}%
9435     \LWR@setexparray{LWR@colatspec}{3}{}%
9436     \LWR@setexparray{LWR@colbangspec}{leftedge}{}%
9437     \LWR@setexparray{LWR@colbangspec}{1}{}%
9438     \LWR@setexparray{LWR@colbangspec}{2}{}%
```

```
9439    \LWR@setexparray{LWR@colbangspec}{3}{}%
9440    \LWR@setexparray{LWR@colbeforespec}{1}{}%
9441    \LWR@setexparray{LWR@colbeforespec}{2}{}%
9442    \LWR@setexparray{LWR@colbeforespec}{3}{}%
9443    \LWR@setexparray{LWR@colafterspec}{1}{}%
9444    \LWR@setexparray{LWR@colafterspec}{2}{}%
9445    \LWR@setexparray{LWR@colafterspec}{3}{}%
9446    \LWR@setexparray{LWR@colbarspec}{leftedge}{}%
9447    \LWR@setexparray{LWR@colbarspec}{1}{}%
9448    \LWR@setexparray{LWR@colbarspec}{2}{}%
9449    \LWR@setexparray{LWR@colbarspec}{3}{}%
9450    \LWR@setexparray{LWR@coladdclass}{1}{}%
9451    \LWR@setexparray{LWR@coladdclass}{2}{}%
9452    \LWR@setexparray{LWR@coladdclass}{3}{}%
```

Starting at the first column specification:

```
9453    \defcounter{LWR@tablecolspecindex}{1}%
```

Place the colspecs string length into `\LWR@strresult`, and remember the number of characters in the column specification:

```
9454    \expandarg%
9455    \StrLen{\LWR@origcolspec}[\LWR@strresult]%
9456    \fullexpandarg%
9457    \LWR@traceinfo{original column spec length: \LWR@strresult}%
9458    \defcounter{LWR@tablecolspecwidth}{\LWR@strresult}%
```

Haven't seen any optional arguments so far

```
9459    \boolfalse{LWR@opttablecol}%
```

Scan through the column specifications:

```
9460    \whileboolexpr{%
9461        not test{%
9462            \ifnumcomp{\value{LWR@tablecolspecindex}}{>}%
9463                {\value{LWR@tablecolspecwidth}}%
9464        }%
9465    }%
9466    {%
```

Place the next single-character column type into `\LWR@strresult`:

```
9467    \expandarg%
9468    \StrChar{\LWR@origcolspec}{\arabic{LWR@tablecolspecindex}}[\LWR@strresult]%
9469    \LWR@traceinfo{position \arabic{LWR@tablecolspecindex}: \LWR@strresult}%
9470    \fullexpandarg%
```

Not yet found a valid column type:

```
9471    \boolfalse{LWR@validtablecol}%
```

Skip over any optional arguments, such as siunitx S column:

```
9472    \IfStrEq{\LWR@strresult}{[}{\booltrue{LWR@opttablecol}}{}%
```

Throw away anything found inside the optional argument:

```
9473    \ifbool{LWR@opttablecol}%
9474    {}% inside an optional argument
9475    {% not an optional tabular argument
```

Not inside an optional argument, so consider the column type:

```
9476        \ifcsdef{LWR@columntype@\LWR@strresult}%
9477            {\csuse{LWR@columntype@\LWR@strresult}}%
9478            {}%
```

If an unknown column type, use l:

```
9479        \ifbool{LWR@validtablecol}{}{%
9480            \LWR@traceinfo{invalid column type: \LWR@strresult}%
9481            \LWR@parsenormalcolumn{l}%
9482        }%
9483    }% not an optional column argument
```

If read the closing bracket, no longer inside the optional argument:

```
9484    \IfStrEq{\LWR@strresult}{]}{\boolfalse{LWR@opttablecol}}{}%
```

Move to the next character:

```
9485    \defaddtocounter{LWR@tablecolspecindex}{1}%
9486    }% whiledo
9487 }%
```

77.15   colortbl and xcolor tabular color support

These macros provide a minimal emulation of some colortbl macros which might appear between table cells. If colortbl is loaded, these macros will be replaced with functional versions.

For each of the HTML colors below, the text for the HTML color is set if requested, but the macro is empty if none has been set.

\rownum   Reserve a counter register.

```
9488  \@ifundefined{rownum}{\newcount\rownum}{}
```

\@rowcolors   Emulated in case xcolor is not used.

```
9489 \newcommand*{\@rowcolors}{}
```

\@rowc@lors   Emulated in case xcolor is not used.

```
9490 \newcommand*{\@rowc@lors}{}
```

\LWR@xcolorrowHTMLcolor   Emulated xcolor row color.

```
9491 \newcommand*{\LWR@xcolorrowHTMLcolor}{}
```

\LWR@columnHTMLcolor HTMLstyle code for the column color.

> 9492 \def\LWR@columnHTMLcolor{}

\LWR@rowHTMLcolor HTMLstyle code for the row color.

> 9493 \def\LWR@rowHTMLcolor{}

\LWR@cellHTMLcolor HTMLstyle code for the cell color.

> 9494 \def\LWR@cellHTMLcolor{}

\LWR@ruleHTMLcolor HTMLstyle code for the rule color.

> 9495 \newcommand*{\LWR@ruleHTMLcolor}{}

\rowcolor [⟨*model*⟩] {⟨*color*⟩} [⟨*left overhang*⟩] [⟨*right overhang*⟩] Print version. The HTML version is in lwarp-colortbl. Used before starting a tabular data cell, thus \LWR@getmynexttoken.

> 9496 \newcommand*{\rowcolor}{\LWR@getmynexttoken}%

\arrayrulecolor [⟨*model*⟩] {⟨*color*⟩}

\arrayrulecolornexttoken [⟨*model*⟩] {⟨*color*⟩}

> Print versions for use outside and inside a tabular:

> 9497 \newcommand{\arrayrulecolor}[2][named]{}
> 9498 \newcommand{\arrayrulecolornexttoken}[2][named]{\LWR@getmynexttoken}

\doublerulesepcolor [⟨*model*⟩] {⟨*color*⟩}

\doublerulesepcolornexttoken [⟨*model*⟩] {⟨*color*⟩}

> Print versions for use inside and outside a tabular:

> 9499 \newcommand{\doublerulesepcolor}[2][named]{}
> 9500 \newcommand{\doublerulesepcolornexttoken}[2][named]{\LWR@getmynexttoken}

77.16 Starting a new row

\LWR@maybenewtablerow If have not yet started a new table row, begin one now. Creates a new row tag, adding a class for hline or tbrule if necessary.

> 9501 \newcommand*{\LWR@maybenewtablerow}
> 9502 {%
> 9503   \ifbool{LWR@startedrow}%
> 9504   {}% started the row
> 9505   {% not started the row

Pre-compute the aria-hidden attribute, used to hide from screen readers the final row if it is only used to create the bottom border:

```
9506        \ifbool{LWR@tabularfinalrow}%
9507          {%
9508             \renewcommand*{\LWR@tempone}%
9509               { aria-hidden=\textquotedbl{}true\textquotedbl}%
9510          }%
9511          {%
9512             \renewcommand*{\LWR@tempone}{}%
9513          }%
```

Start a new row if doing \hline:

```
9514        \ifboolexpr{%
9515          test{\ifnumcomp{\value{LWR@hlines}}{>}{0}} or%
9516          test{\ifnumcomp{\value{LWR@hdashedlines}}{>}{0}}%
9517        }%
9518        {%
9519          \LWR@htmltag{%
9520             tr %
9521             class=\textquotedbl{}hline\textquotedbl%
9522             \LWR@tempone% aria-hidden
9523          }%
9524          \LWR@orignewline%
```

Remember that now have started the row, and create the row tag, with a class if necessary.

```
9525             \booltrue{LWR@startedrow}%
9526             \booltrue{LWR@intabularmetadata}%
9527        }%
```

If not doing \hline, start a row if doing a top or bottom rule:

```
9528        {% not doing hline
9529          \ifbool{LWR@doingtbrule}%
9530          {%
9531             \ifdefvoid{\LWR@ruleHTMLcolor}{%
9532                \LWR@htmltag{%
9533                   tr %
9534                   class=\textquotedbl{}tbrule\textquotedbl%
9535                   \LWR@tempone% aria-hidden
9536                }%
9537          }{%
9538                \LWR@htmltag{%
9539                   tr class=\textquotedbl{}tbrule\textquotedbl\ % space
9540                   style=\textquotedbl{}border-top: 1px solid % space
9541                      \LWR@origpound\LWR@ruleHTMLcolor \textquotedbl{}%
9542                   \LWR@tempone% aria-hidden
9543                }%
9544          }%
9545          \LWR@orignewline%
```

Remember that now have started the row, and create the row tag, with a class if necessary.

```
9546             \booltrue{LWR@startedrow}%
9547             \booltrue{LWR@intabularmetadata}%
```

```
9548                    }%
9549                    {%
```

If not the final row, start a new row:

```
9550                           \ifbool{LWR@tabularfinalrow}%
9551                              {}%
9552                              {%
9553                                  \LWR@htmltag{tr}\LWR@orignewline%
```

Remember that now have started the row, and create the row tag, with a class if necessary.

```
9554                                  \booltrue{LWR@startedrow}%
9555                                  \booltrue{LWR@intabularmetadata}%
9556                              }%
9557                          }%
9558         }% end of not doing hline
9559     }% end of not started the row
9560 }
```

77.17   Printing vertical bar tags

\LWR@printbartag {⟨*index*⟩}

Adds to a tabular data cell an HTML class name for a left/right vertical bar.

```
9561 \newcommand*{\LWR@printbartag}[1]{%
9562     \LWR@traceinfo{LWR@printbartag !#1!}%
9563     \ifboolexpr{bool{LWR@tabularmutemods} or bool{LWR@emptyatbang}}%
9564     {}% muting or empty
9565     {% not muting
9566         \edef\LWR@tempone{\LWR@getexparray{LWR@colbarspec}{#1}}%
9567         \ifdefempty{\LWR@tempone}{}{ \LWR@tempone}%
9568     }% not muting
9569     \LWR@traceinfo{LWR@printbartag done}%
9570 }
```

77.18   Printing @ or ! tags

\LWR@printatbang {⟨*at — or — bang*⟩} {⟨*index*⟩}

```
9571 \newcommand*{\LWR@printatbang}[2]{%
```

Fetch the column at or bang spec:

```
9572     \xdef\LWR@atbangspec{\LWR@getexparray{LWR@col#1spec}{#2}}%
9573     \LWR@traceinfo{atbang: #2 !\LWR@atbangspec!}%
```

Only generate if is not empty;

```
9574     \ifdefempty{\LWR@atbangspec}%
9575     {}%
9576     {% not empty
```

```
9577          \LWR@htmltag{%
9578              td class=\textquotedbl{}td#1%
9579              \LWR@subaddcmidruletrim{}{}%
9580              \LWR@printbartag{#2}%
9581              \textquotedbl{}%
9582              \LWR@tdstartstyles%
9583              \LWR@addcmidrulewidth%
9584              \LWR@addcdashline%
9585              \LWR@addtabularrulecolors%
9586              \LWR@tdendstyles%
9587          }%
```

Create an empty cell if muting for the \bottomrule:

```
9588          \ifboolexpr{bool{LWR@tabularmutemods} or bool{LWR@emptyatbang}}%
9589              {}%
9590              {\LWR@atbangspec}%
9591 %
9592          \LWR@htmltag{/td}\LWR@orignewline%
9593          \global\booltrue{LWR@tabularcelladded}%
9594      }% not empty
9595 }%
```

\LWR@addleftmostbartag

```
9596 \newcommand*{\LWR@addleftmostbartag}{%
9597      \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{=}{1}{%
9598          \LWR@printbartag{leftedge}%
9599      }{}%
9600 }
```

\LWR@tabularleftedge

```
9601 \newcommand*{\LWR@tabularleftedge}{%
9602      \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{=}{1}%
9603      {%
9604          \LWR@printatbang{at}{leftedge}%
9605          \LWR@printatbang{bang}{leftedge}%
9606      }% left edge
9607      {}% not left edge
9608 }
```

77.19   Cell opening tag

\LWR@thiscolspec   Temporary storage.

```
9609 \newcommand*{\LWR@thiscolspec}{}
```

\LWR@tabledatasinglecolumntag   Print a table data opening tag with style for alignment and color.

```
9610 \newcommand*{\LWR@tabledatasinglecolumntag}%
9611 {%
9612      \LWR@traceinfo{LWR@tabledatasinglecolumntag}%
9613      \LWR@maybenewtablerow%
```

Don't start a new paragraph tag if have already started one, or have found the end
of the tabular, or if are inside a \multirow:

```
9614    \ifboolexpr{
9615        bool{LWR@intabularmetadata}
9616        and not bool{LWR@exitingtabular}
9617        and not bool {LWR@in@multirow@par}
9618    }%
9619    {% making a tabular data cell
```

Print the @ and ! contents before first column:

```
9620            \LWR@tabularleftedge%
```

Fetch the current column's alignment character into \LWR@strresult:

```
9621            \xdef\LWR@strresult{%
9622        \LWR@getexparray{LWR@tablecolspec}{\arabic{LWR@tableLaTeXcolindex}}}%
9623            }%
```

Print the start of a new table data cell:

```
9624            \LWR@traceinfo{LWR@tabledatasinglecolumntag: about to print td tag}%
9625          \LWR@htmltag{%
9626                td class=\textquotedbl{}td%
```

Append this column's spec:

```
9627                \LWR@strresult%
```

If this column has a cmidrule, add "rule" to the end of the HTML class tag. Also
add vertical bar tags.

```
9628                \LWR@addcmidruletrim%
9629                \LWR@addleftmostbartag%
9630                \LWR@printbartag{\arabic{LWR@tableLaTeXcolindex}}%
```

Add any tabular > column text alignment or font control css:

```
9631                \LWR@getexparray{LWR@coladdclass}%
9632                    {\arabic{LWR@tableLaTeXcolindex}}%
```

Close the class description:

```
9633                \textquotedbl{}%
```

Add styles for rules, alignment:

```
9634                \LWR@tdstartstyles%
9635                \LWR@addcmidrulewidth%
9636                \LWR@addcdashline%

9637                \xdef\LWR@thiscolspec{%
9638                    \LWR@getexparray{LWR@tablecolspec}%
9639                        {\arabic{LWR@tableLaTeXcolindex}}%
9640                }%
9641                \LWR@addformatwpalignment{\LWR@thiscolspec}%
```

Add styles for cell and rule colors:

```
9642                      \LWR@addtabularrowcolor%
9643                      \LWR@addtabularrulecolors%


9644                      \LWR@tdendstyles%
9645                 }% HTML td
9646            \LWR@traceinfo{LWR@tabledatasinglecolumntag: done printing td tag}%
```

If this is a p, m, b, or X column, allow paragraphs:

```
9647                  \ifboolexpr{%
9648                      test{ \ifdefstring{\LWR@strresult}{p} } or
9649                      test{ \ifdefstring{\LWR@strresult}{m} } or
9650                      test{ \ifdefstring{\LWR@strresult}{b} }
9651                  }%
9652                  {% allow pars
9653                 \LWR@traceinfo{LWR@tabledatasinglecolumntag: about to LWR@startpars}%
9654                      \booltrue{LWR@tableparcell}%
9655                      \LWR@startpars%
9656                 \LWR@traceinfo{LWR@tabledatasinglecolumntag: done with LWR@startpars}%
9657                  }% allow pars
9658                  {}% no pars
```

Print the > contents unless muted for the \bottomrule:

```
9659                  \ifboolexpr{bool{LWR@tabularmutemods} or bool{LWR@emptyatbang}}%
9660                  {}%
9661                  {%
9662                 \LWR@getexparray{LWR@colbeforespec}{\arabic{LWR@tableLaTeXcolindex}}%
9663                  }%
9664                  \boolfalse{LWR@intabularmetadata}%
9665           }% making a tabular data cell
9666           {}% not making a tabular data cell
9667           \LWR@traceinfo{LWR@tabledatasinglecolumntag: done}%
9668 }%
```

77.20   Midrules

LWR@midrules   LWR@midrules is a data array (section 44) of columns each containing a non-zero width if a midrule should be created for this column.

LWR@trimlrules   LWR@trimlrules is a data array (section 44) of columns containing l if a midrule should be left trimmed for each column.

LWR@trimrrules   LWR@trimrrules is a data array (section 44) of columns containing r if a midrule should be right trimmed for each column.

LWR@cdashlines   LWR@cdashlines is a data array (section 44) of columns each containing a Y if an arydshln package "cdashed line" should be created for this column.

\LWR@heavyrulewidth (*Len*)   The default width of the rule.

```
9669 \newlength{\LWR@heavyrulewidth}
9670 \setlength{\LWR@heavyrulewidth}{.08em}
```

\LWR@lightrulewidth (*Len*)   The default width of the rule.

```
9671 \newlength{\LWR@lightrulewidth}
9672 \setlength{\LWR@lightrulewidth}{.05em}
```

\LWR@cmidrulewidth (*Len*)   The default width of the rule.

```
9673 \newlength{\LWR@cmidrulewidth}
9674 \setlength{\LWR@cmidrulewidth}{.03em}
```

\LWR@thiscmidrulewidth (*Len*)   The width of the next rule, defaulting to \LWR@cmidrulewidth.

If not \LWR@cmidrulewidth, a style will be used to generate the custom width.

Assigned from the LWR@midrules array.

```
9675 \newlength{\LWR@thiscmidrulewidth}
9676 \setlength{\LWR@thiscmidrulewidth}{\LWR@cmidrulewidth}
```

\LWR@clearmidrules   Start new midrules. Called at beginning of tabular and also at \\.

Clears all LWR@midrules and LWR@trimrules markers for this line.

```
9677 \newcommand*{\LWR@clearmidrules}
9678 {%
9679     \defcounter{LWR@midrulecounter}{1}%
9680     \whileboolexpr{%
9681         not test{%
9682             \ifnumcomp{\value{LWR@midrulecounter}}{>}%
9683                 {\value{LWR@tabletotalLaTeXcols}}%
9684         }%
9685     }%
9686     {%
9687         \LWR@setexparray{LWR@midrules}{\arabic{LWR@midrulecounter}}{0pt}%
9688         \setlength{\LWR@thiscmidrulewidth}{\LWR@cmidrulewidth}%
9689         \LWR@setexparray{LWR@trimlrules}{\arabic{LWR@midrulecounter}}{}%
9690         \LWR@setexparray{LWR@trimrrules}{\arabic{LWR@midrulecounter}}{}%
9691         \LWR@setexparray{LWR@cdashlines}{\arabic{LWR@midrulecounter}}{N}%
9692         \defaddtocounter{LWR@midrulecounter}{1}%
9693     }%
9694 }
```

\LWR@subcmidrule   {⟨*width*⟩} {⟨*trim*⟩} {⟨*leftcolumn*⟩} {⟨*rightcolumn*⟩}

Marks LWR@midrules data array elements to be non-zero widths from left to right columns. Also marks trimming for the L and/or R columns.

LWR@doingcmidrule is set to force an empty row at the end of the tabular to create the rule.

```
9695 \newcommand*{\LWR@subcmidrule}[4]{%
9696     \defcounter{LWR@midrulecounter}{#3}%
9697     \whileboolexpr{%
9698         not test {%
9699             \ifnumcomp{\value{LWR@midrulecounter}}{>}{#4}%
9700         }%
9701     }%
9702     {%
9703         \LWR@setexparray{LWR@midrules}{\arabic{LWR@midrulecounter}}{#1}%
9704         \defaddtocounter{LWR@midrulecounter}{1}%
```

```
9705        }% whiledo
9706        \IfSubStr{#2}{l}{\LWR@setexparray{LWR@trimlrules}{#3}{l}}{}%
9707        \IfSubStr{#2}{r}{\LWR@setexparray{LWR@trimrrules}{#4}{r}}{}%
9708        \booltrue{LWR@doingcmidrule}%
9709 }
```

\LWR@docmidrule [⟨*width*⟩] (⟨*trim*⟩) {⟨*leftcolumn-rightcolumn*⟩}

> Marks LWR@midrules array elements to be a non-zero width from left to right columns. Also marks trimming for the L and/or R columns.

```
9710 \NewDocumentCommand{\LWR@docmidrule}
9711      {O{\LWR@cmidrulewidth} D(){} >{\SplitArgument{1}{-}}m}
9712      {\LWR@subcmidrule{#1}{#2}#3}
```

\LWR@subcdashline {⟨*leftcolumn*⟩} {⟨*rightcolumn*⟩}

> Marks LWR@cdashlines data array elements to be Y from left to right columns.

> LWR@doingcmidrule is set to force an empty row at the end of the tabular to create the rule.

```
9713 \newcommand*{\LWR@subcdashline}[2]{%
9714      \defcounter{LWR@midrulecounter}{#1}%
9715      \whileboolexpr{%
9716          not test {%
9717              \ifnumcomp{\value{LWR@midrulecounter}}{>}{#2}%
9718          }%
9719      }%
9720      {%
9721          \LWR@setexparray{LWR@cdashlines}{\arabic{LWR@midrulecounter}}{Y}%
9722          \defaddtocounter{LWR@midrulecounter}{1}%
9723      }% whiledo
9724      \booltrue{LWR@doingcmidrule}%
9725 }
```

\LWR@docdashline {⟨*leftcolumn-rightcolumn*⟩}

> Marks LWR@cdashlines data array elements to be Y from left to right columns.

```
9726 \NewDocumentCommand{\LWR@docdashline}{>{\SplitArgument{1}{-}}m}%
9727 {%
9728      \LWR@subcdashline#1%
9729 }
```

\LWR@tdstartstyles Begins possibly adding a table data cell style.

```
9730 \newcommand*{\LWR@tdstartstyles}{\boolfalse{LWR@tdhavecellstyle}}
```

\LWR@tdaddstyle Starts adding a table data cell style.

```
9731 \newcommand*{\LWR@tdaddstyle}{%
9732      \ifbool{LWR@tdhavecellstyle}%
9733          {; }%
9734          { style=\textquotedbl}%
9735      \booltrue{LWR@tdhavecellstyle}%
9736 }
```

\LWR@tdendstyles   Finishes possibly adding a table data cell style. Prints the closing quote.

```
9737 \newcommand*{\LWR@tdendstyles}{%
9738     \ifbool{LWR@tdhavecellstyle}%
9739         {%
9740             \textquotedbl%
9741             \boolfalse{LWR@tdhavecellstyle}%
9742         }{}%
9743 }
```

\LWR@subaddcmidruletrim {⟨*lefttrim*⟩} {⟨*righttrim*⟩} Adds a \cmidrule with optional trim.

```
9744 \newcommand*{\LWR@subaddcmidruletrim}[2]{%
9745     \setlength{\LWR@templengthone}{%
9746         \LWR@getexparray{LWR@midrules}{\arabic{LWR@tableLaTeXcolindex}}%
9747     }%
9748     \ifdimcomp{\LWR@templengthone}{>}{0pt}%
9749         {%
```

Print the class with left and right trim letters appended:

```
9750             \space tdrule#1#2%
```

Remember the width of the rule:

```
9751             \setlength{\LWR@thiscmidrulewidth}{\LWR@templengthone}%
9752         }%
9753         {%
9754             \setlength{\LWR@thiscmidrulewidth}{0pt}%
9755         }%
9756 }
```

\LWR@addcmidruletrim   Adds left or right trim to a \cmidrule.

```
9757 \newcommand*{\LWR@addcmidruletrim}{%
9758     \LWR@subaddcmidruletrim%
9759         {\LWR@getexparray{LWR@trimlrules}{\arabic{LWR@tableLaTeXcolindex}}}%
9760         {\LWR@getexparray{LWR@trimrrules}{\arabic{LWR@tableLaTeXcolindex}}}%
9761 }
```

\LWR@addrulewidth {⟨*thiswidth*⟩} {⟨*defaultwidth*⟩}

If not default width, add a custom style with width and color depending on thiswidth.

Must be placed between \LWR@tdstartstyles and \LWR@tdendstyles.

```
9762 \newcommand{\LWR@addrulewidth}[2]{%
```

Only add a custom width if thiswidth is different than the defaultwidth, or if a color is being used:

```
9763     \ifboolexpr{%
9764         test{\ifdimcomp{#1}{=}{0pt}} or
9765         (
9766             ( test{\ifdimcomp{#1}{=}{#2}} and not bool{FormatWP} )
```

```
9767            and ( test {\ifdefvoid{\LWR@ruleHTMLcolor}} )
9768        )
9769    }%
9770    {}% default width and color
9771    {% custom width and/or color
```

Ensure that the width is wide enough to display in the browser:

```
9772        \LWR@forceminwidth{#1}%
```

Begin adding another style:

```
9773        \LWR@tdaddstyle%
```

The style itself:

```
9774        border-top:\LWR@printlength{\LWR@atleastonept} solid % space
```

If default gray, the darkness of the color depends on the thickness of the rule:

```
9775        \ifdefvoid{\LWR@ruleHTMLcolor}{%
9776            \ifdimcomp{#1}{<}{\LWR@lightrulewidth}%
9777            {\LWR@origpound{}A0A0A0}%
9778            {% lightrule or heaver
9779                \ifdimcomp{#1}{<}{\LWR@heavyrulewidth}%
9780                {\LWR@origpound{}808080}%
9781                {black}%
9782            }% lightrule or heavier
9783        }{%
9784            \LWR@origpound\LWR@ruleHTMLcolor%
9785        }%
9786    }% custom width and/or color
9787 }
```

\LWR@addcmidrulewidth Adds a style for the rule width.

Must be placed between \LWR@tdstartstyles and \LWR@tdendstyles.

```
9788 \newcommand{\LWR@addcmidrulewidth}{%
9789    \LWR@addrulewidth{\LWR@thiscmidrulewidth}{\LWR@cmidrulewidth}%
9790 }
```

\LWR@addcdashline Must be placed between \LWR@tdstartstyles and \LWR@tdendstyles.

```
9791 \newcommand{\LWR@addcdashline}{%
9792    \edef\LWR@tempone{%
9793        \LWR@getexparray{LWR@cdashlines}{\arabic{LWR@tableLaTeXcolindex}}%
9794    }%
9795    \ifdefstring{\LWR@tempone}{Y}{%
9796        \LWR@tdaddstyle%
9797        border-top: 1pt dashed %
9798        \ifdefvoid{\LWR@ruleHTMLcolor}%
9799            {black}%
9800            {\LWR@origpound\LWR@ruleHTMLcolor}%
9801    }{}%
9802 }
```

\LWR@WPcell {⟨*text-align*⟩} {⟨*vertical-align*⟩}

```
9803 \newcommand*{\LWR@WPcell}[2]{%
9804     \LWR@tdaddstyle%
9805     \LWR@print@mbox{text-align:#1}; \LWR@print@mbox{vertical-align:#2}%
9806 }
```

\LWR@addformatwpalignment {⟨*colspec*⟩}

> If FormatWP, adds a style for the alignment.

> Must be placed between \LWR@tdstartstyles and \LWR@tdendstyles.

```
9807 \newcommand*{\LWR@addformatwpalignment}[1]{%
9808     \ifbool{FormatWP}{%
9809         \IfSubStr{#1}{l}{\LWR@WPcell{left}{middle}}{}%
9810         \IfSubStr{#1}{c}{\LWR@WPcell{center}{middle}}{}%
9811         \IfSubStr{#1}{r}{\LWR@WPcell{right}{middle}}{}%
9812         \IfSubStr{#1}{p}{\LWR@WPcell{left}{bottom}}{}%
9813         \IfSubStr{#1}{m}{\LWR@WPcell{left}{middle}}{}%
9814         \IfSubStr{#1}{b}{\LWR@WPcell{left}{top}}{}%
9815     }{}%
9816 }
```

77.21   Cell colors

\LWR@addtabularrowcolor Adds a cell's row color style, if needed.

> No color is added for the final row of empty cells which finishes each tabular.

```
9817 \newcommand*{\LWR@addtabularrowcolor}{%
9818     \ifbool{LWR@tabularmutemods}{}{%
9819         \ifdefvoid{\LWR@rowHTMLcolor}{%
9820             \ifdefvoid{\LWR@xcolorrowHTMLcolor}{}%
9821             {% xcolor row color
9822                 \LWR@tdaddstyle%
9823                 background:\LWR@origpound\LWR@xcolorrowHTMLcolor%
9824             }%
9825         }%
9826         {% explicit row color
9827             \LWR@tdaddstyle%
9828             background:\LWR@origpound\LWR@rowHTMLcolor%
9829         }%
9830     }%
9831 }
```

\LWR@addtabularhrulecolor Adds a cell's horizontal rule color style, if needed.

```
9832 \newcommand*{\LWR@addtabularhrulecolor}{%
```

> If either form of horizontal rule is requested:

```
9833     \ifboolexpr{%
9834         test{\ifnumcomp{\value{LWR@hlines}}{>}{0}} or%
9835         test{\ifnumcomp{\value{LWR@hdashedlines}}{>}{0}} or%
9836         bool{LWR@doingtbrule}%
9837     }{%
```

If there is a no custom color:

```
9838        \ifdefvoid{\LWR@ruleHTMLcolor}%
9839        {%
9840            \ifnumcomp{\value{LWR@hlines}}{>}{1}%
9841            {%
9842                \LWR@tdaddstyle%
9843                border-top: 4px double%
9844            }{% else
9845            \ifnumcomp{\value{LWR@hdashedlines}}{>}{1}%
9846            {%
9847                \LWR@tdaddstyle%
9848                border-top: 2px dashed%
9849            }{% else
9850            \ifnumcomp{\value{LWR@hdashedlines}}{=}{1}%
9851            {%
9852                \LWR@tdaddstyle%
9853                border-top: 1px dashed%
9854            }{}}}%
```

If no color and not doubled or dashed, then add nothing, since a simpler rule is the default.

```
9855        }%
```

If there is a custom color:

```
9856        {%
9857            \ifnumcomp{\value{LWR@hlines}}{>}{1}%
9858            {%
9859                \LWR@tdaddstyle%
9860                border-top: 4px double \LWR@origpound\LWR@ruleHTMLcolor%
9861            }{% else
9862            \ifnumcomp{\value{LWR@hdashedlines}}{>}{1}%
9863            {%
9864                \LWR@tdaddstyle%
9865                border-top: 2px dashed \LWR@origpound\LWR@ruleHTMLcolor%
9866            }{% else
9867            \ifnumcomp{\value{LWR@hdashedlines}}{=}{1}%
9868            {%
9869                \LWR@tdaddstyle%
9870                border-top: 1px dashed \LWR@origpound\LWR@ruleHTMLcolor%
9871            }{% else
9872                \LWR@tdaddstyle%
9873                border-top: 1px solid \LWR@origpound\LWR@ruleHTMLcolor%
9874            }}}%
9875        }%
9876    }{}%
9877 }
```

\LWR@addtabularrulecolors Adds a cell's rule color styles, if needed.

No color is added for the final row of empty cells which finishes each tabular.

```
9878 \newcommand*{\LWR@addtabularrulecolors}{%
```

Custom horizonal rule color:

```
9879    \LWR@addtabularhrulecolor%
```

No vertical rules if finishing the `tabular` with a row of empty cells:

```
9880        \ifbool{LWR@tabularmutemods}{}{%
```

If at the leftmost cell, possibly add a leftmost vertical rule:

```
9881            \ifnumequal{\value{LWR@tableLaTeXcolindex}}{1}{%
```

Fetch the left edge's vertical bar specification:

```
9882                \edef\LWR@tempone{\LWR@getexparray{LWR@colbarspec}{leftedge}}%
```

Add a custom style if a vertical bar was requested:

```
9883                \ifdefstring{\LWR@tempone}{tvertbarl}{%
9884                    \LWR@tdaddstyle%
9885                    border-left: 1px solid % space
9886                        \LWR@vertruleHTMLcolor%
9887                }{}%
9888                \ifdefstring{\LWR@tempone}{tvertbarldouble}{%
9889                    \LWR@tdaddstyle%
9890                    border-left: 4px double % space
9891                        \LWR@vertruleHTMLcolor%
9892                }{}%
9893                \ifdefstring{\LWR@tempone}{tvertbarldash}{%
9894                    \LWR@tdaddstyle%
9895                    border-left: 1px dashed % space
9896                        \LWR@vertruleHTMLcolor%
9897                }{}%
9898                \ifdefstring{\LWR@tempone}{tvertbarldoubledash}{%
9899                    \LWR@tdaddstyle%
9900                    border-left: 2px dashed % space
9901                        \LWR@vertruleHTMLcolor%
9902                }{}%
9903            }{}%
```

Possibly add a right vertical rule for this cell:

```
9904            \edef\LWR@tempone{%
9905             \LWR@getexparray{LWR@colbarspec}{\arabic{LWR@tableLaTeXcolindex}}%
9906            }%
9907            \ifdefstring{\LWR@tempone}{tvertbarr}{%
```

Add a custom style if a vertical bar was requested:

```
9908                \LWR@tdaddstyle%
9909                border-right: 1px solid \LWR@vertruleHTMLcolor%
9910            }{}%
9911            \ifdefstring{\LWR@tempone}{tvertbarrdouble}{%
9912                \LWR@tdaddstyle%
9913                border-right: 4px double \LWR@vertruleHTMLcolor%
9914            }{}%
9915            \ifdefstring{\LWR@tempone}{tvertbarrdash}{%
9916                \LWR@tdaddstyle%
9917                border-right: 1px dashed \LWR@vertruleHTMLcolor%
9918            }{}%
9919            \ifdefstring{\LWR@tempone}{tvertbarrdoubledash}{%
9920                \LWR@tdaddstyle%
9921                border-right: 2px dashed \LWR@vertruleHTMLcolor%
```

```
9922        }{}%
9923     }%
9924 }
```

\LWR@subaddtabularcellcolor {⟨*html color*⟩}

```
9925 \newcommand*{\LWR@subaddtabularcellcolor}[1]{%
9926     \LWR@htmltag{div class=\textquotedbl{}cellcolor\textquotedbl\ % space
9927         style=\textquotedbl{}%
9928         background:\LWR@origpound{}{}#1 %
9929         \textquotedbl\ %
9930     }% space
9931     \defaddtocounter{LWR@cellcolordepth}{1}%
9932 }
```

\LWR@addtabularcellcolor Adds a cell color style, if needed.

```
9933 \newcommand*{\LWR@addtabularcellcolor}{%
9934     \ifdefvoid{\LWR@cellHTMLcolor}%
9935     {%
9936         \ifdefvoid{\LWR@rowHTMLcolor}%
9937         {%
9938             \ifdefvoid{\LWR@xcolorrowHTMLcolor}%
9939             {%
9940                 \ifdefvoid{\LWR@columnHTMLcolor}%
9941                 {}%
9942                 {\LWR@subaddtabularcellcolor{\LWR@columnHTMLcolor}}%
9943             }%
9944             {\LWR@subaddtabularcellcolor{\LWR@xcolorrowHTMLcolor}}%
9945         }%
9946         {\LWR@subaddtabularcellcolor{\LWR@rowHTMLcolor}}%
9947     }%
9948     {\LWR@subaddtabularcellcolor{\LWR@cellHTMLcolor}}%
9949 }
```

77.22   Multicolumns

77.22.1   Parsing multicolumns

\LWR@printmccoltype@normal {⟨*col type*⟩}

Prints the column type, and remembers that any vertical bars are no longer on the left edge.

```
9950 \newcommand*{\LWR@printmccoltype@normal}[1]{%
9951     #1%
9952     \boolfalse{LWR@mcolvertbaronleft}%
9953 }
```

\LWR@printmccoltype@ignore {⟨*col type*⟩}

This type does not print a multi-column data cell.

```
9954 \newcommand*{\LWR@printmccoltype@ignore}[1]{}%
```

\LWR@printmccoltype@vertbar {⟨*col type*⟩}

> Adds a left or right vertical bar.

```
9955 \newcommand*{\LWR@printmccoltype@vertbar}[1]{%
9956     \ifbool{LWR@mcolvertbaronleft}%
9957         {\defaddtocounter{LWR@mcolvertbarsl}{1}}% left edge
9958         {\defaddtocounter{LWR@mcolvertbarsr}{1}}% not left edge
9959 }
```

\LWR@printmccoltype@colon {⟨*col type*⟩}

> Adds a left or right vertical bar.

```
9960 \newcommand*{\LWR@printmccoltype@colon}[1]{%
9961     \ifbool{LWR@mcolvertbaronleft}%
9962         {\defaddtocounter{LWR@mcolvertbarsldash}{1}}% left edge
9963         {\defaddtocounter{LWR@mcolvertbarsrdash}{1}}% not left edge
9964 }
```

\LWR@printmccoltype@semicolon {⟨*col type*⟩}

> Adds a left or right vertical bar.

```
9965 \let\LWR@printmccoltype@semicolon\LWR@printmccoltype@colon
```

\LWR@printmccoltype {⟨*colspec*⟩} Print any valid column type found. Does not print @, !, >, or < columns or their associated tokens.

> This is printed as part of the table data tag's class.

> \LWR@columntype@mctype@<type> is defined by \LWR@modifycolumntype.

```
9966 \newcommand*{\LWR@printmccoltype}[1]{%
9967     \LWR@traceinfo{lwr@printmccoltype -#1-}%
```

> Get one token of the column spec:

```
9968     \StrChar{#1}{\arabic{LWR@tablemulticolspos}}[\LWR@strresult]%
```

> Detokenize to avoid problems with special characters:

```
9969     \edef\LWR@strresult{\detokenize\expandafter{\LWR@strresult}}%
```

> Add to the HTML tag depending on which column type is found:

```
9970     \ifcsdef{LWR@columntype@mctype@\LWR@strresult}%
9971         {\csuse{LWR@columntype@mctype@\LWR@strresult}}%
9972         {\boolfalse{LWR@mcolvertbaronleft}}%
9973     \LWR@traceinfo{lwr@printmccoltype done}%
9974 }
```

\LWR@printmccoldata@other {⟨*num args to skip*⟩} {⟨*entire colspec*⟩}

> For @, !, >, <, print the next token without paragraph tags:

```
9975 \newcommand*{\LWR@printmccoldata@other}[2]{%
9976     \defaddtocounter{LWR@tablemulticolspos}{1}%
9977     \StrChar{#2}{\arabic{LWR@tablemulticolspos}}[\LWR@strresult]%
9978     \LWR@strresult%
```

A valid column data type was found:

```
9979     \booltrue{LWR@validtablecol}%
9980 }
```

\LWR@printmccoldata@skip {⟨*num args to skip*⟩} {⟨*entire colspec*⟩}

Nothing to print for this column type.

```
9981 \newcommand*{\LWR@printmccoldata@skip}[2]{%
9982     \defaddtocounter{LWR@tablemulticolspos}{#1}%
```

A valid column data type was found:

```
9983     \booltrue{LWR@validtablecol}%
9984 }
```

For \LWR@printmccoldata@..., {⟨*num args to skip*⟩} is provided by \LWR@columntype@mcdata@<coltype> when it was defined by \LWR@modifycolumntype. \entire colspec is provided by \LWR@printmccoldata when it uses \LWR@columntype@mcdata@<coltype>.

\LWR@printmccoldata@normal {⟨*num args to skip*⟩} {⟨*entire colspec*⟩}

```
9985 \newcommand*{\LWR@printmccoldata@normal}[2]{%
9986     \LWR@multicoltext%
9987     \defaddtocounter{LWR@tablemulticolspos}{#1}%
9988 }
```

\LWR@printmccoldata@paragraph {⟨*num args to skip*⟩} {⟨*entire colspec*⟩}

```
9989 \newcommand*{\LWR@printmccoldata@paragraph}[2]{%
9990     \LWR@startpars%
9991     \LWR@multicoltext%
9992     \defaddtocounter{LWR@tablemulticolspos}{#1}%
9993     \LWR@stoppars%
9994 }
```

\LWR@printmccoldata {⟨*entire colspec*⟩}

Print the data for any valid column type found.

```
9995 \newcommand*{\LWR@printmccoldata}[1]{%
9996     \LWR@traceinfo{lwr@printmccoldata -#1}%
```

Not yet found a valid column type:

```
9997     \boolfalse{LWR@validtablecol}%
```

Get one token of the column spec, into a local copy in case nested.

```
9998        \StrChar{#1}{\arabic{LWR@tablemulticolspos}}[\LWR@strresult]%
9999        \edef\LWR@printmccoldatatoken{\LWR@strresult}%
```

Print the text depending on which column type is found. Also handles @, >, < as it comes to them.

```
10000       \ifcsdef{LWR@columntype@mcdata@\LWR@printmccoldatatoken}%
10001           {\csuse{LWR@columntype@mcdata@\LWR@printmccoldatatoken}{#1}}%
10002           {}%
```

If an unknown column type, print the text:

```
10003       \ifbool{LWR@validtablecol}{}{\LWR@multicoltext{}}%
```

Tracing:

```
10004       \LWR@traceinfo{lwr@printmccoldata done}%
10005 }
```

\parsemulticolumnalignment {⟨*1: colspec*⟩} {⟨*2: printresults csname*⟩}

Scan the multicolumn specification and execute the printfunction for each entry.

Note that the spec for a p{spec} column, or @, >, <, is a token list which will NOT match l, c, r, or p.

```
10006 \newcommand*{\LWR@parsemulticolumnalignment}[2]{%
10007     \defcounter{LWR@tablemulticolspos}{1}%
10008     \StrLen{#1}[\LWR@strresult]%
10009     \defcounter{LWR@tablemulticolswidth}{\LWR@strresult}%
```

Scan across the tokens in the column spec:

```
10010     \whileboolexpr{%
10011         not test {%
10012             \ifnumcomp{\value{LWR@tablemulticolspos}}{>}%
10013                 {\value{LWR@tablemulticolswidth}}%
10014         }%
10015     }%
10016     {%
```

Execute the assigned print function for each token in the column spec:

```
10017         \csuse{#2}{#1}%
```

Move to the next token in the column spec:

```
10018         \defaddtocounter{LWR@tablemulticolspos}{1}%
10019     }%
10020 }
```

77.22.2 Multicolumn factored code

\LWR@addmulticolvertrulecolor

```
10021 \newcommand*{\LWR@addmulticolvertrulecolor}{%
```

No vertical rules if finishing the tabular with a row of empty cells:

```
10022    \ifbool{LWR@tabularmutemods}{}{%
```

Left side:

```
10023        \ifnumcomp{\value{LWR@mcolvertbarsl}}{=}{1}{%
10024            \LWR@tdaddstyle%
10025            border-left: 1px solid \LWR@vertruleHTMLcolor%
10026        }{}%
10027        \ifnumcomp{\value{LWR@mcolvertbarsl}}{>}{1}{%
10028            \LWR@tdaddstyle%
10029            border-left: 4px double \LWR@vertruleHTMLcolor%
10030        }{}%
10031        \ifnumcomp{\value{LWR@mcolvertbarsldash}}{=}{1}{%
10032            \LWR@tdaddstyle%
10033            border-left: 1px dashed \LWR@vertruleHTMLcolor%
10034        }{}%
10035        \ifnumcomp{\value{LWR@mcolvertbarsldash}}{>}{1}{%
10036            \LWR@tdaddstyle%
10037            border-left: 2px dashed \LWR@vertruleHTMLcolor%
10038        }{}%
```

Right side:

```
10039        \ifnumcomp{\value{LWR@mcolvertbarsr}}{=}{1}{%
10040            \LWR@tdaddstyle%
10041            border-right: 1px solid \LWR@vertruleHTMLcolor%
10042        }{}%
10043        \ifnumcomp{\value{LWR@mcolvertbarsr}}{>}{1}{%
10044            \LWR@tdaddstyle%
10045            border-right: 4px double \LWR@vertruleHTMLcolor%
10046        }{}%
10047        \ifnumcomp{\value{LWR@mcolvertbarsrdash}}{=}{1}{%
10048            \LWR@tdaddstyle%
10049            border-right: 1px dashed \LWR@vertruleHTMLcolor%
10050        }{}%
10051        \ifnumcomp{\value{LWR@mcolvertbarsrdash}}{>}{1}{%
10052            \LWR@tdaddstyle%
10053            border-right: 2px dashed \LWR@vertruleHTMLcolor%
10054        }{}%
10055    }%
10056 }
```

```
10057 \newcommand{\LWR@multicoltext}{}
```

To find multicolumn right trim:

```
10058 \newcounter{LWR@lastmulticolumn}
```

\LWR@domulticolumn [⟨*1: vpos*⟩] [⟨*2: #rows*⟩] {⟨*3: numLaTeXcols*⟩} {⟨*4: numHTMLcols*⟩} {⟨*5: colspec*⟩}
{⟨*6: text*⟩}

```
10059 \NewDocumentCommand{\LWR@domulticolumn}{o o m m m +m}{%
10060     \LWR@traceinfo{LWR@domulticolumn -#1- -#2- -#4- -#5-}%
```

Remember the text to be inserted, and when used remember that a valid column type was found:

```
10061    \renewcommand{\LWR@multicoltext}{%
10062        #6%
10063        \booltrue{LWR@validtablecol}%
10064    }%
```

Expand the preamble and save it.

```
10065    \LWR@expandpreamble{#5}%
10066    \edef\LWR@origmccolspec{\the\@temptokena}%
```

Compute the rightmost column to be included. This is used to create the right trim.

```
10067    \defcounter{LWR@lastmulticolumn}{\value{LWR@tableLaTeXcolindex}}%
10068    \defaddtocounter{LWR@lastmulticolumn}{#3}%
10069    \defaddtocounter{LWR@lastmulticolumn}{-1}%
```

Row processing:

```
10070    \LWR@maybenewtablerow%
```

Begin the opening table data tag:

```
10071    \LWR@htmltag{%
10072        td colspan=\textquotedbl#4\textquotedbl\ %

10073        \IfValueT{#2}{ % rows?
10074            rowspan=\textquotedbl#2\textquotedbl\ %
10075        }%

10076        class=\textquotedbl{}td%
```

Print the column type and vertical bars:

```
10077        \defcounter{LWR@mcolvertbarsl}{0}%
10078        \defcounter{LWR@mcolvertbarsr}{0}%
10079        \defcounter{LWR@mcolvertbarsldash}{0}%
10080        \defcounter{LWR@mcolvertbarsrdash}{0}%
10081        \booltrue{LWR@mcolvertbaronleft}%
10082    \LWR@parsemulticolumnalignment{\LWR@origmccolspec}{\LWR@printmccoltype}%
```

If this column has a cmidrule, add "rule" to the end of the HTML class tag.

If this position had a "Y" then add "rule" for a horizontal rule:

```
10083        \LWR@subaddcmidruletrim%
10084            {%
10085                \LWR@getexparray{LWR@trimlrules}%
10086                    {\arabic{LWR@tableLaTeXcolindex}}%
10087            }%
10088            {%
10089                \LWR@getexparray{LWR@trimrrules}%
10090                    {\arabic{LWR@lastmulticolumn}}%
10091            }%
```

Also add vertical bar class.

```
10092        \ifnumcomp{\value{LWR@mcolvertbarsl}}{=}{1}{ tvertbarl}{}%
10093        \ifnumcomp{\value{LWR@mcolvertbarsl}}{>}{1}{ tvertbarldouble}{}%
10094        \ifnumcomp{\value{LWR@mcolvertbarsr}}{=}{1}{ tvertbarr}{}%
10095        \ifnumcomp{\value{LWR@mcolvertbarsr}}{>}{1}{ tvertbarrdouble}{}%
10096        \ifnumcomp{\value{LWR@mcolvertbarsldash}}{=}{1}{ tvertbarldash}{}%
10097        \ifnumcomp{\value{LWR@mcolvertbarsldash}}{>}{1}%
10098            { tvertbarldoubledash}{}%
10099        \ifnumcomp{\value{LWR@mcolvertbarsrdash}}{=}{1}{ tvertbarrdash}{}%
10100        \ifnumcomp{\value{LWR@mcolvertbarsrdash}}{>}{1}%
10101            { tvertbarrdoubledash}{}%
```

Close the class tag's opening quote:

```
10102        \textquotedbl{}%
```

```
10103        \LWR@tdstartstyles%
```

Style for vertical position:

```
10104        \IfValueT{#1}{% vpos?
10105            \ifstrequal{#1}{b}%
10106                {%
10107                    \LWR@tdaddstyle%
10108                    \LWR@print@mbox{vertical-align:bottom}%
10109                }{}%
10110            \ifstrequal{#1}{t}%
10111                {%
10112                    \LWR@tdaddstyle%
10113                    \LWR@print@mbox{vertical-align:top}%
10114                }{}%
10115        }% vpos?
```

Style for row colors:

```
10116        \LWR@addtabularrowcolor%
```

Other styles:

```
10117        \LWR@addcmidrulewidth%
10118        \LWR@addcdashline%
10119        \LWR@addtabularhrulecolor%
10120        \LWR@addmulticolvertrulecolor%
10121        \LWR@addformatwpalignment{\LWR@origmccolspec}%
10122        \LWR@tdendstyles%
10123    }% end of the opening table data tag
10124    \boolfalse{LWR@intabularmetadata}%
10125    \LWR@parsemulticolumnalignment{\LWR@origmccolspec}{LWR@printmccoldata}%
10126 }
```

77.22.3 Multicolumn

\LWR@htmlmulticolumn {⟨*numcols*⟩} {⟨*alignment*⟩} {⟨*text*⟩}

```
10127 \NewDocumentCommand{\LWR@htmlmulticolumn}{m m +m}%
10128 {%
```

Figure out how many extra HTML columns to add for @ and ! columns:

```
10129    \LWR@tabularhtmlcolumns{\arabic{LWR@tableLaTeXcolindex}}{#1}%
```

Create the multicolumn tag:

```
10130    \LWR@domulticolumn{#1}{\arabic{LWR@tabhtmlcoltotal}}{#2}{#3}%
```

Move to the next LATEX column:

```
10131    \defaddtocounter{LWR@tableLaTeXcolindex}{#1}%
10132    \defaddtocounter{LWR@tableLaTeXcolindex}{-1}%
```

Skip any trailing @ or ! columns for this cell:

```
10133    \booltrue{LWR@skipatbang}%
10134 }
```

77.22.4 Longtable captions

longtable captions use \multicolumn.

Per the caption package. User-redefinable float type.

```
10135 \providecommand*{\LTcaptype}{table}
```

\LWR@longtabledatacaptiontag * [⟨*toc entry*⟩] {⟨*caption*⟩}

```
10136 \NewDocumentCommand{\LWR@longtabledatacaptiontag}{s o +m}
10137 {%
```

Remember the latest name for \nameref:

```
10138    \IfValueTF{#2}{% optional given?
10139        \ifblank{#2}% optional empty?
10140        {\LWR@setlatestname{#3}}% empty
10141        {\LWR@setlatestname{#2}}% given and non-empty
10142    }% optional given
10143    {\LWR@setlatestname{#3}}% no optional
```

Create a multicolumn across all the columns:

Figure out how many extra HTML columns to add for @ and ! columns found between the first and the last column:

```
10144    \LWR@tabularhtmlcolumns{1}{\arabic{LWR@tabletotalLaTeXcols}}%
```

Create the multicolumn tag. The caption will be centered by the CSS caption class.

```
10145    \LWR@domulticolumn{\arabic{LWR@tabletotalLaTeXcols}}%
10146        {\arabic{LWR@tabhtmlcoltotal}}%
10147        {p}%
10148    {% \LWR@domulticolumn
10149    \IfBooleanTF{#1}% star?
```

Star version, show a caption but do not make a LOT entry:

```
10150     {% yes star
10151         \LWR@figcaption%
10152         \LWR@isolate{#3}%
10153         \endLWR@figcaption%
10154     }%
10155     {% No star:
```

Not the star version:

Don't step the counter if \caption[]{A caption.}

```
10156         \ifbool{LWR@starredlongtable}%
10157         {%
10158             \ifblank{#2}% TOC entry
10159             {}%
10160             {%
10161                 \refstepcounter{\LTcaptype}%
10162                 \protected@edef\@currentlabel{%
10163                     \@nameuse{p@\LTcaptype}\@nameuse{the\LTcaptype}%
10164                 }%
10165             }%
10166         }{}%
```

Create an HTML caption. Afterwards, maybe make a LOT entry.

```
10167         \LWR@figcaption%
10168         \LWR@isolate{\@nameuse{fnum@\LTcaptype}}%
10169         \CaptionSeparator%
10170         \LWR@isolate{#3}%
10171         \endLWR@figcaption%
```

See if an optional caption was given:

```
10172         \ifblank{#2}% TOC entry empty
```

if the optional caption was given, but empty, do not form a TOC entry

```
10173         {}%
```

If the optional caption was given, but might only be []:

```
10174         {% TOC entry not empty
10175             \IfNoValueTF{#2}% No TOC entry?
```

The optional caption is []:

```
10176             {% No TOC entry
10177                 \addcontentsline%
10178                 {\@nameuse{ext@\LTcaptype}}%
10179                 {\LTcaptype}%
10180                 {%
10181                     \protect\numberline%
10182                 {\LWR@isolate{\@nameuse{p@\LTcaptype}}\@nameuse{the\LTcaptype}}%
10183                     {\ignorespaces \LWR@isolate{#3}\protect\relax}%
10184                 }%
10185             }% end of No TOC entry
```

The optional caption has text enclosed:

```
10186            {% yes TOC entry
10187                \addcontentsline%
10188                {\@nameuse{ext@\LTcaptype}}%
10189                {\LTcaptype}%
10190                {%
10191                    \protect\numberline%
10192                {\LWR@isolate{\@nameuse{p@\LTcaptype}}\@nameuse{the\LTcaptype}}%
10193                    {\ignorespaces \LWR@isolate{#2}\protect\relax}%
10194                }%
10195            }% end of yes TOC entry
10196        }% end of TOC entry not empty
10197    }% end of no star
```

Skip any trailing @ or ! columns for this cell:

```
10198    \booltrue{LWR@skipatbang}%
10199    }% end of \LWR@domulticolumn
10200    \defaddtocounter{LWR@tableLaTeXcolindex}{\value{LWR@tabletotalLaTeXcols}}%
10201    \defaddtocounter{LWR@tableLaTeXcolindex}{-1}
10202
10203 }
```

77.22.5   Counting HTML tabular columns

The LaTeX specification for a table includes a number of columns separated by the & character. These columns differ in content from line to line. Additional virtual columns may be specified by the special @ and ! columns. These columns are identical from line to line, but may be skipped during a multicolumn cell.

For HTML output, @ and ! columns are placed into their own tabular columns. Thus, a LaTeX \multicolumn command may span several additional @ and ! columns in HTML output. These additional columns must be added to the total number of columns spanned by an HTML multi-column data cell.

```
10204 \newcounter{LWR@tabhtmlcolindex}
10205 \newcounter{LWR@tabhtmlcolend}
10206 \newcounter{LWR@tabhtmlcoltotal}
```

\LWR@subtabularhtmlcolumns {⟨*index*⟩}

Factored from \LWr@tabularhtmlcolumns, which follows.

```
10207 \newcommand*{\LWR@subtabularhtmlcolumns}[1]{%
```

Temporarily define a macro equal to the @ specification for this column:

```
10208    \edef\LWR@atbangspec{\LWR@getexparray{LWR@colatspec}{#1}}%
```

If the @ specification is not empty, add to the count:

```
10209    \ifdefempty{\LWR@atbangspec}%
10210        {}%
10211        {\defaddtocounter{LWR@tabhtmlcoltotal}{1}}%
```

Likewise for the ! columns:

```
10212      \edef\LWR@atbangspec{\LWR@getexparray{LWR@colbangspec}{#1}}%
10213      \ifdefempty{\LWR@atbangspec}%
10214          {}%
10215          {\defaddtocounter{LWR@tabhtmlcoltotal}{1}}%
10216 }
```

\LWR@tabularhtmlcolumns {⟨*starting L&ATEX column*⟩} {⟨*number L&ATEX columns*⟩}

Compute the total number of HTML columns being spanned, considering the starting L&ATEX table column and the number of L&ATEX tabular columns being spanned. Any @ and ! columns within this span are included in the total count. The resulting number of HTML columns is returned in the counter LWR@tabhtmlcoltotal.

```
10217 \newcommand*{\LWR@tabularhtmlcolumns}[2]{%
```

Count the starting index, compute ending index, and begin with the count being the L&ATEX span, to which additional @ and ! columns may be added:

```
10218      \defcounter{LWR@tabhtmlcolindex}{#1}%
10219      \defcounter{LWR@tabhtmlcoltotal}{#2}%
10220      \defcounter{LWR@tabhtmlcolend}{#1}%
10221      \defaddtocounter{LWR@tabhtmlcolend}{#2}%
```

If at the left edge, add the at/bang columns for the left edge:

```
10222      \ifnumcomp{\value{LWR@tabhtmlcolindex}}{=}{1}{%
10223          \LWR@subtabularhtmlcolumns{leftedge}%
10224      }{}%
```

Walk across the L&ATEX columns looking for @ and ! columns:

```
10225      \whileboolexpr{%
10226          test {%
10227          \ifnumcomp{\value{LWR@tabhtmlcolindex}}{<}{\value{LWR@tabhtmlcolend}}%
10228          }%
10229      }%
10230      {%
10231          \LWR@subtabularhtmlcolumns{\arabic{LWR@tabhtmlcolindex}}%
10232          \defaddtocounter{LWR@tabhtmlcolindex}{1}%
10233      }% whiledo
10234 }
```

```
10235 \end{warpHTML}
```

77.23   Multirow if not loaded

A default defintion in case multirow is not loaded. This is used during table parsing.

```
10236 \begin{warpHTML}
10237 \newcommand{\multirow}[2][c]{}
10238 \end{warpHTML}
```

77.24 Multicolumnrow

A print-mode version is defined here, and is also used during HTML output while inside a lateximage.

See section 436 for the HTML versions.

for HTML & PRINT: 10239 \begin{warpall}

\multicolumnrow {⟨*1:cols*⟩} {⟨*2:halign*⟩} [⟨*3:vpos*⟩] {⟨*4:numrows*⟩} [⟨*5:bigstruts*⟩] {⟨*6:width*⟩} [⟨*7:fixup*⟩] {⟨*8:text*⟩}

For discussion of the use of \DeclareExpandableDocumentCommand, see: https://tex.stackexchange.com/questions/168434/problem-with-abbreviation-of-multirow-and-multicolumn-latex

\AtBeginDocument to adjust after the user may have loaded multirow, which requires several tests to determine which version is loaded and thus which options are available.

10240 \AtBeginDocument{

\@ifundefined{@xmultirow} determines if multirow was never loaded.

Null action if not loaded:

```
10241 \@ifundefined{@xmultirow}
10242 {
10243 \DeclareExpandableDocumentCommand{\LWR@print@multicolumnrow}%
10244     {+m +m +O{c} +m +O{0} +m +O{0pt} +m}%
10245     {}%
10246 }% no version of multirow was loaded
10247 {% \@xmultirow defined, so some version of multirow was loaded
```

\IfPackageLoadedTF{multirow} determines if v2.0 or later of multirow was used, which included the \ProvidesPackage macro.

The print version:

```
10248 \IfPackageLoadedTF{multirow}{% v2.0 or newer
10249 \IfPackageAtLeastTF{multirow}{2016/09/01}% 2016/09/27 for v2.0
10250 {% v2.0+:
10251 \DeclareExpandableDocumentCommand{\LWR@print@multicolumnrow}%
10252     {+m +m +O{c} +m +O{0} +m +O{0pt} +m}%
10253     {\multicolumn{#1}{#2}{\@xmultirow[#3]{#4}[#5]{#6}[#7]{#8}}}%
10254 }
10255 {% loaded but older, probably not executed:
10256 \DeclareExpandableDocumentCommand{\LWR@print@multicolumnrow}%
10257     {+m +m +O{c} +m +O{0} +m +O{0pt} +m}%
10258     {\multicolumn{#1}{#2}{\@xmultirow{#4}[#5]{#6}[#7]{#8}}}%
10259 }
10260 }% packageloaded{multirow}
```

If not \IfPackageLoadedTF{multirow} but \@xmultirow is defined, then this must be v1.6 or earlier, which did not \ProvidesPackage{multirow}, and did not have the vposn option.

```
10261 {% v1.6 or older did not \ProvidePackage
```

```
10262 \DeclareExpandableDocumentCommand{\LWR@print@multicolumnrow}%
10263     {+m +m +O{c} +m +O{0} +m +O{0pt} +m}%
10264     {\multicolumn{#1}{#2}{\@xmultirow{#4}[#5]{#6}[#7]{#8}}}%
10265 }
10266
10267 }% \@ifundefined{@xmultirow}
10268
10269 \providecommand*{\multicolumnrow}{\LWR@print@multicolumnrow}
10270
10271 }% AtBeginDocument

10272 \end{warpall}
```

77.25   Utility macros inside a table

```
10273 \begin{warpHTML}
```

Used to prevent opening a tabular data cell if the following token is one which does not create tabular data:

```
10274 \newcommand*{\LWR@donothing}{}
```

In case array is not loaded:

```
10275 \let\firsthline\relax
10276 \let\lasthline\relax
10277 \newcommand*{\firsthline}{}
10278 \newcommand*{\lasthline}{}
```

In case bigdelim is not loaded:

```
10279 \newcommand*{\ldelim}{}
10280 \newcommand*{\rdelim}{}
```

```
10281 \end{warpHTML}
```

77.26   Special-case tabular markers

```
10282 \begin{warpall}
```

\TabularMacro   Place this just before inserting a custom macro in a table data cell. Doing so tells lwarp not to automatcally start a new HTML table data cell yet. See section 8.10.1.

```
10283 \newcommand*{\TabularMacro}{}
```

```
10284 \end{warpall}
```

\ResumeTabular   Used to resume tabular entries after resuming an environment.

⚠ tabular inside another environment   When creating a new environment which contains a tabular environment, lwarp's emulation of the tabular does not automatically resume when the containing enviroment ends, resulting in corrupted HTML rows. To fix this, use \ResumeTabular as follows. This is ignored in print mode.

```
\StartDefiningTabulars %   (& is used in a definition)
\newenvironment{outerenvironment}
{
  \tabular{cc}
left & right \\
}
{
  \TabularMacro\ResumeTabular
  left & right \\
  \endtabular
}
\StopDefiningTabulars
```

for HTML output: 10285 \begin{warpHTML}

10286 \newcommand*{\ResumeTabular}{%
10287   \boolfalse{LWR@exitingtabular}%
10288   \boolfalse{LWR@tabularmutemods}%
10289   \boolfalse{LWR@tabularfinalrow}%
10290   \LWR@getmynexttoken%
10291 }

10292 \end{warpHTML}

for PRINT output: 10293 \begin{warpprint}

10294 \newcommand*{\ResumeTabular}{}

10295 \end{warpprint}

77.27   Checking for a new table cell

for HTML output: 10296 \begin{warpHTML}

\LWR@tabledatacolumntag   Open a new HTML table cell unless the next token is for a macro which does not create data, such as \hline, \toprule, etc:

10297 \newcommand*{\LWR@tabledatacolumntag}%
10298 {%
10299   \LWR@traceinfo{LWR@tabledatacolumntag}%

        \show\LWR@mynexttoken to see what tokens to look for

If not any of the below, start a new table cell:

10300   \global\let\LWR@mynextaction\LWR@tabledatasinglecolumntag%

If find \end, exit the tabular:

10301   \ifdefequal{\LWR@mynexttoken}{\end}%
10302   {%
10303   \booltrue{LWR@tabularfinalrow}%
10304   \booltrue{LWR@exitingtabular}%
10305   }{}%
```

longtable can have a caption in a cell

```
10306    \ifdefequal{\LWR@mynexttoken}{\caption}%
10307        {\global\let\LWR@mynextaction\LWR@donothing}{}%
```

Look for other things which would not start a table cell:

```
10308    \ifdefequal{\LWR@mynexttoken}{\multicolumn}%
10309        {\global\let\LWR@mynextaction\LWR@donothing}{}%
10310    \ifdefequal{\LWR@mynexttoken}{\multirow}%
10311        {\global\let\LWR@mynextaction\LWR@donothing}{}%
10312    \ifdefequal{\LWR@mynexttoken}{\multicolumnrow}%
10313        {\global\let\LWR@mynextaction\LWR@donothing}{}%
10314    \ifdefequal{\LWR@mynexttoken}{\noalign}%
10315        {\global\let\LWR@mynextaction\LWR@donothing}{}%
```

If an \mrowcell, this is a cell to be skipped over:

```
10316    \ifdefequal{\LWR@mynexttoken}{\mrowcell}%
10317        {\global\let\LWR@mynextaction\LWR@donothing}{}%
```

If an \mcolrowcell, this is a cell to be skipped over:

```
10318    \ifdefequal{\LWR@mynexttoken}{\mcolrowcell}%
10319        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10320    \ifdefequal{\LWR@mynexttoken}{\TabularMacro}%
10321        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10322    \ifdefequal{\LWR@mynexttoken}{\hline}%
10323        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10324    \ifdefequal{\LWR@mynexttoken}{\firsthline}%
10325        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10326    \ifdefequal{\LWR@mynexttoken}{\lasthline}%
10327        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10328    \ifdefequal{\LWR@mynexttoken}{\toprule}%
10329        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10330    \ifdefequal{\LWR@mynexttoken}{\midrule}%
10331        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10332    \ifdefequal{\LWR@mynexttoken}{\cmidrule}%
10333        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10334    \ifdefequal{\LWR@mynexttoken}{\morecmidrules}%
10335        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10336    \ifdefequal{\LWR@mynexttoken}{\specialrule}%
10337        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10338    \ifdefequal{\LWR@mynexttoken}{\cline}%
10339        {\global\let\LWR@mynextaction\LWR@donothing}{}%
```

```
10340    \ifdefequal{\LWR@mynexttoken}{\bottomrule}%
10341        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10342    \ifdefequal{\LWR@mynexttoken}{\hhline}%
10343        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10344    \ifdefequal{\LWR@mynexttoken}{\rowcolor}%
10345        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10346    \ifdefequal{\LWR@mynexttoken}{\arrayrulecolor}%
10347        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10348    \ifdefequal{\LWR@mynexttoken}{\doublerulesepcolor}%
10349        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10350    \ifdefequal{\LWR@mynexttoken}{\warpprintonly}%
10351        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10352    \ifdefequal{\LWR@mynexttoken}{\warpHTMLonly}%
10353        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10354    \ifdefequal{\LWR@mynexttoken}{\ldelim}%
10355        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10356    \ifdefequal{\LWR@mynexttoken}{\rdelim}%
10357        {\global\let\LWR@mynextaction\LWR@donothing}{}%
```

For arydshln:

```
10358    \ifdefequal{\LWR@mynexttoken}{\hdashline}%
10359        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10360    \ifdefequal{\LWR@mynexttoken}{\cdashline}%
10361        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10362    \ifdefequal{\LWR@mynexttoken}{\firsthdashline}%
10363        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10364    \ifdefequal{\LWR@mynexttoken}{\lasthdashline}%
10365        {\global\let\LWR@mynextaction\LWR@donothing}{}%
```

Ignore an empty line between rows:

```
10366    \ifdefequal{\LWR@mynexttoken}{\par}%
10367        {%
10368            \global\let\LWR@mynextaction\LWR@donothing%
10369        }{}%
```

No action for an \end token.

Add similar to the above for any other non-data tokens which might appear in the table.

Start the new table cell if was not any of the above:

```
10370    \LWR@traceinfo{LWR@tabledatacolumntag: done, about to do LWR@mynextaction}%
10371    \LWR@mynextaction%
10372 }
```

10373 \end{warpHTML}

77.28 \mrowcell

10374 \begin{warpall}

\mrowcell The user must insert \mrowcell into any \multirow cells which must be skipped.
⚠ multirow cells This command has no action during print output.

10375 \newcommand*{\mrowcell}{}

10376 \end{warpall}

77.29 \mcolrowcell

10377 \begin{warpall}

\mcolrowcell The user must insert \mcolrowcell into any \multicolumnrow cells which must
⚠ multirow cells be skipped. This command has no action during print output.

10378 \newcommand*{\mcolrowcell}{}

10379 \end{warpall}

77.30 HTML tabular environment

10380 \begin{warpHTML}

These are default defininitions in case booktabs is not loaded, and are not ex-
pected to used, but must exist as placeholders. memoir may have already loaded
booktabs.

10381 \providecommand*{\toprule}[1][]{\hline}
10382 \providecommand*{\midrule}[1][]{\hline}
10383 \providecommand*{\cmidrule}{\cline}
10384 \providecommand*{\bottomrule}[1][]{\hline}
10385 \providecommand*{\addlinespace}[1][]{}
10386 \providecommand*{\morecmidrules}{}
10387 \providecommand*{\specialrule}[3]{\hline}

\noalign {⟨*text*⟩} Redefined for use inside tabular.

10388 \LetLtxMacro\LWR@orignoalign\noalign
10389
10390 \newcommand{\LWR@tabularnoalign}[1]{%
10391   \advance\rownum\m@ne%
10392   \LetLtxMacro\LWR@save@xcolorrowHTMLcolor\LWR@xcolorrowHTMLcolor%
10393   \renewcommand*{\LWR@xcolorrowHTMLcolor}{}%
10394   \multicolumn{\value{LWR@tabletotalLaTeXcols}}{l}{#1} \\
10395   \LetLtxMacro\LWR@xcolorrowHTMLcolor\LWR@save@xcolorrowHTMLcolor%
10396   % \@rowc@lors%
10397   \LWR@getmynexttoken%
10398 }

\LWR@HTMLhline The definition of \hline depends on whether tabls has been loaded. If so, optional
space below the line may be specified, but will be ignored.

```
10399 \AtBeginDocument{
10400
10401 \IfPackageLoadedTF{lwarp-tabls}
10402 {
10403     \newcommand*{\LWR@HTMLhline}[1][]{%
10404         \ifbool{FormatWP}%
10405             {\LWR@docmidrule{1-\arabic{LWR@tabletotalLaTeXcols}}}%
10406             {\defaddtocounter{LWR@hlines}{1}}%
10407         \LWR@getmynexttoken}%
10408 }
10409 {
10410     \newcommand*{\LWR@HTMLhline}{%
10411         \ifbool{FormatWP}%
10412             {\LWR@docmidrule{1-\arabic{LWR@tabletotalLaTeXcols}}}%
10413             {\defaddtocounter{LWR@hlines}{1}}%
10414         \LWR@getmynexttoken}%
10415 }
10416
10417 }% AtBeginDocument
```

\LWR@HTMLcline {⟨*columns*⟩}

```
10418 \NewDocumentCommand{\LWR@HTMLcline}{m}%
10419 {%
10420     \LWR@docmidrule{#1}%
10421     \LWR@maybenewtablerow%
10422     \LWR@getmynexttoken%
10423 }%
```

\LWR@tabular@warpprintonly {⟨*contents*⟩}

Only process the contents if producing printed output. Modified inside a tabular
to grab the next token.

```
10424 \newcommand{\LWR@tabular@warpprintonly}[1]{%
10425     \ifbool{warpingprint}{#1}{}%
10426     \LWR@getmynexttoken%
10427 }
```

\LWR@nullifyNoAutoSpacing For babel-french, turn off auto spacing at the start of the tabular, then nullify the
autospacing commands inside the tabular, since they were not compatible with
the tabular parsing code for each cell, which uses xstring.

```
10428 \AtBeginDocument{
10429 \@ifundefined{NoAutoSpacing}%
10430 {% no babel-french
10431     \newcommand*{\LWR@nullifyNoAutoSpacing}{}
10432 }% no babel-french
10433 {% yes babel-french
10434     \newcommand*{\LWR@nullifyNoAutoSpacing}{%
10435         \NoAutoSpacing%
10436         \renewcommand*{\NoAutoSpacing}{}%
10437         \renewcommand*{\LWR@FBcancel}{}%
10438     }
```

```
10439 }% yes babel-french
10440 }% AtBeginDocument
```

tabular (*env.*) <direction> [⟨*vertposition*⟩] {⟨*colspecs*⟩}

The <direction> is from plext for Japanese documents, and is ignored.

```
10441 \StartDefiningTabulars
10442
10443 \NewDocumentCommand{\LWR@HTML@@tabular}{d<> o m}
10444 {%
10445     \LWR@traceinfo{LWR@HTML@@tabular started}%
```

⚠ <table> inside <span> In LaTeX, a tabular may be placed inside a minipage, but in HTML a <table> may not be inside a <span>. Since there may be several nested <span>s, with an unknown number of other objects between, it is hard to undo all these <span>s before the <table> then redo them after. The broswer probably compensates for this situation, but formatting may be lost inside the <table> because several things are neutralized inside a <span>. Furthermore, in the HTML output, the entire <table> is placed on a single line of HTML code, since the line breaking commands are neutralized inside a <span>. Since this is such a sloppy situation, a warning is issued here instructing the user to please isolate the <span> to print-only.

```
10446     \LWR@spanwarnformat{tabular}%
10447     \addtocounter{LWR@tabulardepth}{1}%
```

Not yet started a table row:

```
10448     \boolfalse{LWR@startedrow}%
```

Not yet doing any rules:

```
10449     \defcounter{LWR@hlines}{0}%
10450     \defcounter{LWR@hdashedlines}{0}%
10451     \boolfalse{LWR@doingtbrule}%
10452     \boolfalse{LWR@doingcmidrule}%
```

For babel-french, turn off auto spacing one time, then nullify the autospacing commands since were not compatible with the tabular parsing code.

```
10453     \LWR@nullifyNoAutoSpacing%
```

Have not yet found the end of tabular command. Unmute the @ and ! columns.

```
10454     \boolfalse{LWR@exitingtabular}%
10455     \boolfalse{LWR@tabularmutemods}%
```

Not adding final row for the lower border:

```
10456     \boolfalse{LWR@tabularfinalrow}%
```

Error if failed to use \mrowcell or \mcolrowcell when needed.

```
10457     \boolfalse{LWR@usedmultirow}%
10458     \boolfalse{LWR@foundmrowcell}%
```

In case of nesting:

```
10459    \renewcommand*{\LWR@multicoltext}{}%
10460    \booltrue{LWR@intabularmetadata}%
```

New PDF page, unless in a \multirow:

```
10461    \ifbool{LWR@in@multirow@par}%
10462        {\leavevmode\LWR@orignewline}%
10463        {\LWR@forcenewpage}%
```

In case of nesting, locally no longer in a \multirow:

```
10464    \boolfalse{LWR@in@multirow@par}%
```

Create the table tag:

```
10465    \LWR@htmlblocktag{table}%
```

Parse the table columns:

```
10466    \LWR@parsetablecols{#3}%
```

Table col spec is: \LWR@tablecolspec which is a string of llccrr, etc.

Do not place the table inside a paragraph:

```
10467    \LWR@stoppars%
```

Without at least one header cell, some screen readers think that the table is just for page layout, and do not read it as data. Add a hidden row with a single non-empty header cell to tell the screen readers that this is a table of data for the user.

```
10468    \LWR@htmltag{tr style=\textquotedbl{}display:none\textquotedbl}%
10469    \LWR@htmltag{th}.\LWR@htmltag{/th}%
10470    \LWR@htmltag{/tr}%
10471    \LWR@orignewline%
10472    \LWR@forceemptyline%
```

Track column #:

```
10473    \defcounter{LWR@tableLaTeXcolindex}{1}%
```

Have not yet added data in this column:

```
10474    \global\boolfalse{LWR@tabularcelladded}%
```

Start looking for midrules:

```
10475    \LWR@clearmidrules%
```

\\ becomes a macro to end the table row:

```
10476    \LetLtxMacro{\\}{\LWR@tabularendofline}%
```

\warpprintonly inside a tabular must grab the next token.

```
10477    \LetLtxMacro\warpprintonly\LWR@tabular@warpprintonly%
```

The following adjust for colortbl.

```
10478    \LetLtxMacro\arrayrulecolor\arrayrulecolornexttoken%
10479    \LetLtxMacro\doublerulesepcolor\doublerulesepcolornexttoken%
10480    \def\LWR@columnHTMLcolor{}%
10481    \def\LWR@rowHTMLcolor{}%
10482    \def\LWR@cellHTMLcolor{}%
10483    \@rowcolors%
```

The vertical rules are set to the color active at the start of the tabular. \arrayrulecolor will then affect horizontal rules inside the tabular, but not the vertical rules.

```
10484    \ifdefvoid{\LWR@ruleHTMLcolor}%
10485        {\edef\LWR@vertruleHTMLcolor{black}}%
10486        {\edef\LWR@vertruleHTMLcolor{\LWR@origpound\LWR@ruleHTMLcolor}}%
```

Tracking the depth of cell color <div>s:

```
10487    \defcounter{LWR@cellcolordepth}{0}%
```

The following may appear before a data cell is created, so after doing their actions, we look ahead with \LWR@getmynextoken to see if the next token might create a new data cell:

The optional parameter for \hline supports the tabls package.

```
10488    \LWR@traceinfo{LWR@@HTML@tabular: redefining macros}%
10489    \LetLtxMacro\noalign\LWR@tabularnoalign%
10490    \LetLtxMacro\hline\LWR@HTMLhline%
10491    \LetLtxMacro\cline\LWR@HTMLcline%

10492    \DeclareDocumentCommand{\hdashline}{o}{%
10493        \ifbool{FormatWP}%
10494            {\LWR@docdashline{1-\arabic{LWR@tabletotalLaTeXcols}}}%
10495            {\defaddtocounter{LWR@hdashedlines}{1}}%
10496        \LWR@getmynexttoken%
10497    }%

10498    \DeclareDocumentCommand{\cdashline}{m}{%
10499        \LWR@docdashline{##1}\LWR@getmynexttoken%
10500    }%

10501    \DeclareDocumentCommand{\firsthdashline}{o}{%
10502        \ifbool{FormatWP}%
10503            {\LWR@docdashline{1-\arabic{LWR@tabletotalLaTeXcols}}}%
10504            {\defaddtocounter{LWR@hdashedlines}{1}}%
10505        \LWR@getmynexttoken%
10506    }%

10507    \DeclareDocumentCommand{\lasthdashline}{o}{%
10508        \ifbool{FormatWP}%
10509            {\LWR@docdashline{1-\arabic{LWR@tabletotalLaTeXcols}}}%
10510            {\defaddtocounter{LWR@hdashedlines}{1}}%
10511        \LWR@getmynexttoken%
10512    }%
```

The following create data cells and will have no more data in this cell, so we do not want to look ahead for a possible data cell, so do not want to use \LWR@getmynexttoken.

```
10513    \renewcommand{\multicolumn}{\LWR@htmlmulticolumn}%
10514    \renewcommand*{\mrowcell}{%
10515        \LWR@maybenewtablerow%
10516        \LWR@tabularleftedge%
10517        \booltrue{LWR@skippingmrowcell}%
10518        \booltrue{LWR@foundmrowcell}%
10519    }%
10520    \renewcommand*{\mcolrowcell}{%
10521        \LWR@maybenewtablerow%
10522        \booltrue{LWR@skippingmcolrowcell}%
10523        \booltrue{LWR@foundmrowcell}%
10524    }%
10525    \LetLtxMacro\caption\LWR@longtabledatacaptiontag%
```

Reset for new processing:

```
10526    \boolfalse{LWR@tableparcell}%
10527    \boolfalse{LWR@skippingmrowcell}%
10528    \boolfalse{LWR@skippingmcolrowcell}%
10529    \boolfalse{LWR@skipatbang}%
10530    \boolfalse{LWR@emptyatbang}%
```

Set & for its special meaning inside the tabular:

```
10531    \StartDefiningTabulars%
10532    \protected\gdef&{\LWR@tabularampersand}%
```

Locally force any minipages to be fullwidth, until the end of the `tabular`:

```
10533    \booltrue{LWR@forceminipagefullwidth}%
```

Nest one level deeper of tabular paragraph handling:

```
10534    \addtocounter{LWR@tabularpardepth}{1}%
```

Look ahead for a possible table data cell:

```
10535    \LWR@traceinfo{LWR@@HTML@tabular: about to LWR@getmynexttoken}%
10536    \LWR@getmynexttoken%
10537 }%
```

Ending the environment:

```
10538 \newcommand*{\LWR@HTML@endtabular}
10539 {%
10540    \LWR@traceinfo{LWR@HTML@endtabular}%
```

Unnest one level of tabular paragraph handling:

```
10541    \addtocounter{LWR@tabularpardepth}{-1}%
```

Finish a row which is not yet done:

```
10542    \ifboolexpr{%
```

```
10543          test {%
10544              \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{<}%
10545                  {\value{LWR@tabletotalLaTeXcols}}
10546          } or %
10547          (%
10548              bool{LWR@intabularmetadata} and%
10549              not bool{LWR@tabularcelladded} and%
10550              test {%
10551                  \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{=}%
10552                      {\value{LWR@tabletotalLaTeXcols}}%
10553              }%
10554          )%
10555      }%
10556      {%


10557          \booltrue{LWR@tabularfinalrow}%
10558          \LWR@tabularfinishrow%
10559          \boolfalse{LWR@tabularfinalrow}%
10560      }%
10561      {%
10562          \LWR@closetabledatacell%
10563      }%
10564      \ifbool{LWR@startedrow}%
10565          {\LWR@htmltag{/tr}\LWR@orignewline}%
10566          {}%
```

xcolor row color support:

```
10567      \@rowc@lors%


10568      \LWR@htmlblocktag{/table}%
10569      \boolfalse{LWR@intabularmetadata}%
```

Unnest one level of tabular:

```
10570      \addtocounter{LWR@tabulardepth}{-1}%
```

Restore & to its usual meaning:

```
10571      \ifnumequal{\value{LWR@tabulardepth}}{0}{%
10572          \protected\gdef&{\LWR@origampmacro}%
10573          \StopDefiningTabulars%
10574      }{}%
```

Error if used \multirow or \multicolumnrow without using \mrowcell or \mcolrowcell.

```
10575      \ifbool{LWR@usedmultirow}{%
10576          \ifbool{LWR@foundmrowcell}%
10577              {\relax}%
10578              {%
10579                  \PackageError{lwarp}%
10580                  {%
10581                  When using \protect\multirow, \protect\multicolumnrow,\MessageBreak
10582                      or the bigdelim package,\MessageBreak
10583                  place \protect\mrowcell\space or \protect\mcolrowcell\MessageBreak
10584                      in empty cells which are to be skipped.\MessageBreak
10585                      See the Lwarp package documentation:\MessageBreak
```

```
10586                          "Special cases and limitations" -> "Tabular"
10587                  }%
10588                  {%
10589                          See the Lwarp package documentation:\MessageBreak
10590                          "Special cases and limitations" -> "Tabular".
10591                  }%
10592              }%
10593      }{}%

10594      \LWR@traceinfo{LWR@HTML@endtabular finished}%
10595 }
10596
10597 \csletcs{LWR@HTML@endtabular*}{LWR@HTML@endtabular}
10598
10599 \StopDefiningTabulars
```

siunitx may redefine `tabular`, so set the following later:

```
10600 \AtBeginDocument{
10601      \LetLtxMacro\LWR@origendtabular\endtabular
10602      \csletcs{LWR@origendtabular*}{endtabular*}
10603      \LWR@formatted{@tabular}
10604      \LWR@formatted{endtabular}
10605      \LWR@formatted{endtabular*}
10606 }

10607 \end{warpHTML}
```

78   Cross-references

Sectioning commands have been emulated from scratch, so the cross-referencing commands are custom-written for them. Emulating both avoids several layers of patches.

*_html.aux (*file*)   A new entry in `*_html.aux` is used to remember section name, file, and lateximage depth and number for each label:

```
\newlabel{<labelname>@lwarp}{{<section name>}{<filename>}
                                   {<limagedepth>}{<limagenumber>}}
```

Table 16 shows the data structures related to cross-referencing.

for HTML output: 10608 \begin{warpHTML}

78.1   Setup

\@currentlabelname   To remember the most recently defined section name, description, or caption, for \nameref.

10609 \def\@currentlabelname{\linkhomename}%

\LWR@stripperiod {⟨*text*⟩} [⟨.⟩]

Table 16: Cross-referencing data structures

Original LATEX: (print and HTML)

\refstepcounter: Steps the counter, sets \@currentcounter, \@currentlabel.

\@currentcounter: Counter type <ctr>, such as "section".

\@currentlabel: \p@<ctr>\the<ctr> Updated by \refstepcounter.

\label: Writes to the .aux file:
\newlabel{<label>}{{\@currentlabel}{\thepage}{name}{Href}{}}

\newlabel: When the .aux file is read, sets \r@<label>.

\r@<label>: Set to: {\@currentlabel}{\thepage}{name}{Href}{}

\ref: Returns the first part of \r@<label>.

\pageref: Returns the second part of \r@<label>.

Added by lwarp: (HTML only)

\label: Adds HTML tags (section 78.3), and two more .aux entries (section 78.2), for \r@<label>@lwarp and \r@<label>@lwarp@image. (nameref changes to \ref, etc. are undone \AtBeginDocument.)

\newlabel: {Href} is changed to {<jobname>-autopage-#}, or the label entered by the user. When the .aux file is read, used to set \r@<label>, and then \r@<label>@lwarp, and \r@<label>@lwarp@image.

\r@<label>@lwarp: Set to {{nameref}{pageref}{fileref}{}{}}:

   \LWR@nameref: The section or object name for this label.
   \LWR@currentautosecpageref: The LWR@currentautosecpage for this label.
   \LWR@htmlfileref: The filenumber or name for this label.

\r@<label>@lwarp@image: Set to {{depth}{number}{}{}{}}:

   \LWR@lateximagedepthref: The lateximagedepth for this label.
   \LWR@lateximagenumberref: The lateximagenumber for this label.

\nameref: Emualted from hyperref for lwarp. See section 78.4.

\ref and \nameref: Adds HTML tags. See section 78.4.

Added by amsmath: (print and HTML)

\label: Execution is delayed until the math environment is completed.

\ltx@label: LATEX \label, (HTML: patched by lwarp,) later patched by cleveref.

Added by cleveref: (print and HTML)

\refstepcounter: Added: sets \cref@currentlabel.

\cref@currentlabel: (<type>=<ctr> unless an alias is used):
[<type>][\arabic{<ctr>}][<parent ctrs>]{\p@<ctr>\the<ctr>}
Also see section 62.4 for use with footnotes.

\label: Also writes to the .aux file:
\newlabel{<label>@cref}{{\cref@currentlabel}{\thepage}}{}{}{}

\newlabel: Unchanged. When the .aux file is read, sets \r@<label>@cref.

\r@<label>@cref: Set to: {{\cref@currentlabel}{\thepage}{}{}{}}

Utility functions: See \cref@getlabel, \cref@gettype, \cref@getcounter, \cref@getprefix.

Cross-referencing names: \crefname and \Crefname assign human-readable names for references to this counter type.

Additionally patched by lwarp: (HTML only)

\cref, etc.: Modified for lwarp. See section 206.

\label inside math: See section 85.7.1.

Footnotes: See \noteentry in section 62.4.

Removes a trailing period.

```
10610 \def\LWR@stripperiod#1.\ltx@empty#2\@nil{#1}%
```

\LWR@setlatestname {⟨*object name*⟩}

Removes \label, strips any final period, and remembers the result.

```
10611 \newcommand*{\LWR@setlatestname}[1]{%
```

Remove \label and other commands from the name, the strip any final period. See gettitlestring.

```
10612     \GetTitleStringExpand{#1}%
10613     \edef\@currentlabelname{\detokenize\expandafter{\GetTitleStringResult}}%
10614     \edef\@currentlabelname{%
10615         \expandafter\LWR@stripperiod\@currentlabelname%
10616         \ltx@empty.\ltx@empty\@nil%
10617     }%
10618 }
```

78.2   New lwarp labels.

*_html.aux (*file*)   New entries in *_html.aux are used to remember section name, file, and lateximage depth and number for each label:

```
\newlabel{<labelname>@lwarp}{{<section name>}{<auto page ref>}{<file name>}{}{}}
\newlabel{<labelname>@lwarp@image}{{<limagedepth>}{<limagenumber>}{}{}{}}
```

The LATEX kernel reserves the fifth argument of \r@<label> for future use. Therefore, lwarp adds two *_html.aux entries, one with three and another with two arguments, to write a total of five argument without colliding with the reserved argument.

See:
http://tex.stackexchange.com/questions/57194/
extract-section-number-from-equation-reference

\LWR@setref {⟨*args list*⟩} {⟨*selector*⟩} {⟨*label*⟩}

\@setref without the \null (\hbox), and without the warning messages. Each caused problems with lwarp references. The regular reference will still cause the warning.

```
10619 \def\LWR@setref#1#2#3{%
10620   \ifx#1\relax%
10621     ??%
10622   \else%
10623     \expandafter#2#1%
10624   \fi}
```

\LWR@nameref {⟨*label*⟩} Returns the section name for this label:

```
10625 \newcommand*{\LWR@nameref}[1]{%
```

```
10626        \begingroup%
10627        \LWR@nullifyfootnotes%
10628        \expandafter\LWR@setref\csname r@#1@lwarp\endcsname\LWR@firstoffive{#1}%
10629        \endgroup%
10630 }
```

\LWR@currentautosecpageref {⟨*label*⟩} Returns the LWR@currentautosecpage for this label:

```
10631 \newcommand*{\LWR@currentautosecpageref}[1]{%
10632    \expandafter\LWR@setref\csname r@#1@lwarp\endcsname\LWR@secondoffive{#1}%
10633 }
```

\LWR@htmlfileref {⟨*label*⟩} Returns the file number or name for this label:

```
10634 \newcommand*{\LWR@htmlfileref}[1]{%
10635    \expandafter\LWR@setref\csname r@#1@lwarp\endcsname\LWR@thirdoffive{#1}%
10636 }
```

\LWR@lateximagedepthref {⟨*label*⟩} Returns the lateximagedepth for this label. This determines if the HTML
link will be to a lateximage or just a text link.

```
10637 \newcommand*{\LWR@lateximagedepthref}[1]{%
10638    \expandafter\LWR@setref\csname r@#1@lwarp@image\endcsname\LWR@firstoffive{#1}%
10639 }
```

\LWR@lateximagenumberref {⟨*label*⟩} Returns the lateximagenumber for this label:

```
10640 \newcommand*{\LWR@lateximagenumberref}[1]{%
10641    \expandafter\LWR@setref\csname r@#1@lwarp@image\endcsname\LWR@secondoffive{#1}%
10642 }
```

\LWR@write@lwarplabel {⟨*label*⟩} Sanitize the name and then creates the label:

```
10643 \newcommand*{\LWR@write@lwarplabel}[1]{%
10644    \LWR@traceinfo{LWR@write@lwarplabel !#1!}%
10645    \LWR@setlatestname{\@currentlabelname}%
10646        \@bsphack%
10647        \protected@write\@auxout{}%
10648            {%
10649                \string\newlabel{#1@lwarp}{%
10650                    {\@currentlabelname}%
10651                    {\theLWR@currentautosecpage}%
10652                    {%
10653                        \ifbool{FileSectionNames}%
10654                            {\LWR@thisfilename}%
10655                            {\arabic{LWR@htmlfilenumber}}%
10656                    }%
10657                    {}%
10658                    {}%
10659                }%
10660            }%
10661        \protected@write\@auxout{}%
10662            {%
10663                \string\newlabel{#1@lwarp@image}{%
10664                    {\arabic{LWR@lateximagedepth}}%
10665                    {\arabic{LWR@lateximagenumber}}%
```

```
10666                              {}%
10667                              {}%
10668                              {}%
10669                        }%
10670                  }%
10671            \@esphack%
10672 }
```

78.3   Labels

\LWR@label@subcreatetag Creates the tag from \LWR@sanitized.

```
10673 \newcommand*{\LWR@label@subcreatetag}{%
10674    \LWR@htmltag{a \LWR@print@mbox{id=\textquotedbl\LWR@sanitized\textquotedbl}}%
10675    \LWR@htmltag{/a}%
10676 }
```

\LWR@label@inmathcomment

```
10677 \newcommand*{\LWR@label@inmathcomment}{%
10678    \ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
10679    {%
```

The combined LATEX & HTML label is printed in a \mbox field:

```
10680        \mbox{%
```

Shift the label over to the right side of the environment to avoid over-printing the math:

```
10681            \ifdef{\totwidth@}{\ifbool{LWR@amsmultline}{}{\hspace*{\totwidth@}}}{}%
```

Temporarily end the HTML comment, insert the LATEX & HTML label, then resume the HTML comment. \@firstofone is required to remove extra braces introduced by the amsmath package.)

```
10682                \LWR@htmlclosecomment%
10683                \LWR@label@subcreatetag%
10684                \LWR@htmlopencomment%
10685            }% mbox
10686    }% mathjax
10687    {%
10688        \LWR@label@subcreatetag%
10689    }%
10690 }
```

\LWR@label@createtag {⟨label⟩}   Creates an HTML id tag.

Used by \LWR@new@label and \hyperdef.

\detokenize is used to allow underscores in the labels.

```
10691 \newcommand*{\LWR@label@createtag}[1]{%
10692    \LWR@traceinfo{LWR@label@createtag !#1!}%
```

Create an HTML id tag unless are inside a lateximage, since it would appear in the image:

```
10693     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
10694     {}%
10695     {% not lateximage
```

If not doing a lateximage, create an HTML ID tag.

```
10696         \LWR@sanitize{#1}%
10697         \ifbool{LWR@insidemathcomment}%
10698         {% inside HTML math comment
10699             \LWR@label@inmathcomment%
10700         }% inside HTML math comment
10701         {% not inside HTML math comment
10702             \ifbool{LWR@doingstartpars}%
10703             {% pars allowed
10704                 \ifbool{LWR@doingapar}%
10705                 {% par started
10706                     \LWR@label@subcreatetag%
10707                 }% par started
10708                 {% par not started
10709                     \LWR@stoppars%
10710                     \LWR@label@subcreatetag%
10711                     \LWR@startpars%
10712                 }% par not started
10713             }% pars allowed
10714             {% pars not allowed
10715                     \LWR@label@subcreatetag%
10716             }% pars not allowed
10717         }% not inside HTML math comment
10718     }% not lateximage
10719 }
```

\LWR@new@label {⟨*label*⟩}

> \label during HTML output when not in SVG math mode, removing extra spaces around the label, as done by a regular LATEX \label.
>
> The is also used during a lateximage, including SVG math, since the special label handling is required, but \LWR@label@createtag does not generate HTML tags inside a lateximage.
>
> clevereref later encases this to add its own cross-referencing.
>
> nameref patches are undone \AtBeginDocument.

```
10720 \newcommand*{\LWR@new@label}[1]{%
10721     \LWR@traceinfo{LWR@new@label: starting}%
10722     \LWR@traceinfo{LWR@new@label: !#1!}%
10723 % \@bsphack%
```

Create a traditional LATEX label, as modified by cleveref:

```
10724     \LWR@orig@label{#1}%
```

Create a special label which holds the section number, section name, LWR@htmlfilenumber, LWR@lateximagedepth, and LWR@lateximagenumber:

```
10725     \LWR@traceinfo{%
10726         LWR@new@label: filesectionnames is %
10727         \ifbool{FileSectionNames}{true}{false}%
10728     }%
10729     \LWR@traceinfo{%
10730         LWR@new@label: LWR@thisfilename is !\LWR@thisfilename!%
10731     }%
10732     \LWR@traceinfo{%
10733         LWR@new@label: LWR@htmlfilenumber is \arabic{LWR@htmlfilenumber}%
10734     }%
10735     \LWR@write@lwarplabel{#1}%
10736     \LWR@label@createtag{#1}%
10737     % \@esphack%
10738     \LWR@traceinfo{LWR@new@label: done}%
10739 }
```

78.4   References

\LWR@addlinktitle

```
10740 \newcommand*{\LWR@addlinktitle}{%
10741     \ifdefvoid{\LWR@ThisAltText}{}{ % space
10742         title=\textquotedbl\LWR@ThisAltText\textquotedbl\ % space
10743         \gdef\LWR@ThisAltText{}%
10744     }%
10745 }
```

\LWR@startref {⟨*label*⟩}   (Common code for \ref and \nameref.)

Open an HTML tag reference to a filename, # character, and a label.

```
10746 \newcommand*{\LWR@startref}[1]
10747 {%
10748     \LWR@sanitize{#1}%
10749     \LWR@traceinfo{LWR@startref A: !#1!}%
```

Create the filename part of the link:

```
10750     \LWR@htmltag{a href=\textquotedbl%
10751     \LWR@traceinfo{LWR@startref B}%
10752     \LWR@print@mbox{\LWR@htmlrefsectionfilename{#1}}%
10753     \LWR@traceinfo{LWR@startref C}%
10754     \LWR@origpound%
```

Create the destination id:

See if LWR@lateximagedepth is unknown:

```
10755     \LWR@traceinfo{LWR@startref D: !#1!}%
10756     \ifcsundef{r@#1@lwarp}%
```

"??" if LWR@lateximagedepth is unknown, so create a link with an unknown destination:

```
10757     {%
10758         \LWR@traceinfo{LWR@startref D0: ??}%
10759         ??%
10760     }%
```

If `LWR@lateximagedepth` is known. Use a lateximage if the depth is greater than zero, or a regular link otherwise:

(Using xifthen \ifthenelse here failed in some cases, but etoolbox \ifnumgreater works.)

```
10761     {%
10762         \ifnumgreater{\LWR@lateximagedepthref{#1}}{0}%
10763             {%

10764                 lateximage-\BaseJobname-\LWR@lateximagenumberref{#1}%
10765             }%
10766             {%
10767                 \LWR@traceinfo{LWR@startref D3}%
```

\detokenize is used to allow underscores in the labels:

```
10768                 \LWR@print@mbox{\LWR@sanitized}%
10769             }%
10770     }%
10771     \LWR@traceinfo{LWR@startref E}%
```

Closing quote:

```
10772     \textquotedbl%
```

Maybe add a title:

```
10773     \LWR@addlinktitle%
10774     }%
10775     \LWR@traceinfo{LWR@startref F}%
10776 }
```

\LWR@subnewref {⟨*label*⟩} {⟨*label or sub@label*⟩}

Factored for the subfig package. Uses the original label for the hyper-reference, but prints its own text, such as "1(b)".

```
10777 \NewDocumentCommand{\LWR@subnewref}{m m}{%
10778     \LWR@traceinfo{LWR@subnewref #1 #2}%
10779     \LWR@startref{#1}%
10780     \LWR@print@ref{#2}%
10781     \LWR@htmltag{/a}%
10782 }
```

\ref * {⟨*label*⟩}

\ref is redefined to \LWR@HTML@ref, except inside the text part of a \hyperref, where it is redefined to \LWR@ref@ignorestar.

\LWR@HTML@ref * {⟨*label*⟩} Create an internal document reference link, or without a link if starred per hyperref.

The HTML version:

```
10783 \NewDocumentCommand{\LWR@HTML@ref}{s m}{%
10784     \LWR@traceinfo{LWR@HTML@ref !#2!}%
10785     \IfBooleanTF{#1}%
10786         {\LWR@print@ref{#2}}%
10787         {\LWR@subnewref{#2}{#2}}%
10788 }
10789
10790 \AtBeginDocument{% **8*
10791 \LWR@formatted{ref}
10792 }
10793
10794 \NewDocumentCommand{\LWR@HTML@Ref}{s m}{%
10795     \LWR@traceinfo{LWR@HTML@Ref !#2!}%
10796     \IfBooleanTF{#1}%
10797         {\LWR@print@Ref{#2}}%
10798         {\LWR@subnewref{#2}{#2}}%
10799 }
10800
10801 \AtBeginDocument{% **8*
10802 \LWR@formatted{Ref}
10803 }
```

\LWR@refwithsection * {⟨*label*⟩}

Creates a reference, printing the section number as the text. Used for back references.

```
10804 \NewDocumentCommand{\LWR@refwithsection}{s m}{%
10805     \LWR@traceinfo{LWR@refwithsection !#2!}%
```

If starred, just use the text without a hyperlink:

```
10806     \IfBooleanTF{#1}%
10807         {\LWR@print@ref{\BaseJobname-autopage-\LWR@currentautosecpageref{#2}}}%
```

If not starred: Check for a reference to the start of the document. (Generated by backref.

```
10808         {% not starred
10809             \ifstrequal{#2}{Doc-Start}%
10810                 {%
10811                     \LWR@startref{\BaseJobname-autopage-1}%
10812                     *%
10813                     \LWR@htmltag{/a}%
10814                 }%
10815                 {% not Doc-Start
```

Open the reference:

```
10816                     \LWR@startref{#2}%
```

Add the text of the link.

Check for and handle an undefined reference:

```
10817                              \edef\@tempa{\LWR@currentautosecpageref{#2}}%
10818                              \ifdefstring{\@tempa}{??}%
10819                                  {??}%
```

For a defined reference:

```
10820                                  {% not ??
```

Set \@tempa to \r@<label>, which is {section number}{page number}{name}{Href}{}.

```
10821                                  \edef\@tempa{\csexpandonce{r@\BaseJobname-autopage-\@tempa}}%
10822                                  \expandafter\LWR@edeffirstoffive\@tempa%
```

Check the section number alone:

If the reference has no section number print an asterisk:

```
10823                                      \expandafter\ifblank\expandafter{\@tempa}%
10824                                          {*}
```

If there is a section number, print it:

```
10825                                          {%
10826                                              \LWR@print@ref{%
10827                                      \BaseJobname-autopage-\LWR@currentautosecpageref{#2}%
10828                                              }%
10829                                          }%
10830                                  }% not ??
```

Close the reference:

```
10831                              \LWR@htmltag{/a}%
10832                          }% not Doc-Start
10833                      }% not starred
10834 }
```

For MATHJAX:

```
10835 \CustomizeMathJax{\let\LWRref\ref}
10836 \CustomizeMathJax{\renewcommand{\ref}{\ifstar\LWRref\LWRref}}
```

\pagerefPageFor Text for page references.

```
10837 \newcommand*{\pagerefPageFor}{see }
```

\pageref * {⟨label⟩} Create an internal document reference, or just the unlinked number
               if starred, per hyperref.

```
10838 \NewDocumentCommand{\LWR@new@pageref}{s m}{%
10839     \IfBooleanTF{#1}%
10840         {(\pagerefPageFor\LWR@print@ref{#2})}%
10841         {(\cpageref{#2})}%
10842 }
```

\nameref {⟨*label*⟩}

> nameref may have already defined \nameref. Redefine it here.

```
10843 \providecommand{\nameref}[1]{}%
10844
10845 \renewrobustcmd*{\nameref}[1]{%
10846     \LWR@traceinfo{nameref}%
10847     \LWR@startref{#1}%
10848     \LWR@traceinfo{nameref B}%
10849     \LWR@nameref{#1}%
10850     \LWR@traceinfo{nameref C}%
10851     \LWR@htmltag{/a}%
10852     \LWR@traceinfo{nameref: done}%
10853 }
```

\Nameref {⟨*label*⟩} In print, adds the page number. In HTML, does not.

> Overwrites nameref definition if already defined.

```
10854 \LetLtxMacro\Nameref\nameref
```

\NR@gettitle {⟨*text*⟩} From nameref, used by caption.

```
10855 \def\NR@gettitle#1{%
10856   \GetTitleString{#1}%
10857   \let\@currentlabelname\GetTitleStringResult%
10858 }
```

78.5   Hyper-references

⚠️   Note that the code currently only sanitizes the underscore character. Additional characters should be rendered inert as well. See the hyperref.sty definition of \gdef\hyper@normalise for an example.

hyperref (*Pkg*)

⚠️   Do not tell other packages that hyperref is emulated. Some packages patch various commands if hyperref is present, which will probably break something, and the emulation already handles whatever may be emulated anyhow.

```
10859 % DO NOT TELL OTHER PACKAGES TO ASSUME HYPERREF, lest they attempt to patch it:
10860 % \EmulatesPackage{hyperref}[2015/08/01]% Disabled.  Do not do this.
```

> Emulates hyperref:

\@currentHref Added to support backref.

```
10861 \AtBeginDocument{
10862     \def\@currentHref{\BaseJobname-autopage-\theLWR@previousautopagelabel}%
10863 }
```

\LWR@linkcatcodes Sets catcodes before processing macros which have hyperlinks as arguments.

```
10864 \newcommand*{\LWR@linkcatcodes}{%
10865     \catcode`\#=12%
10866     \catcode`\%=12%
10867     \catcode`\&=12%
10868     \catcode`\~=12%
10869     \catcode`\_=12%
```

For babel-french:

```
10870     \LWR@hook@processingtags%
10871 }
```

\LWR@linkmediacatcodes   Sets catcodes before processing macros which have hyperlinks as arguments. Mod-
                         ified for multimedia links.

```
10872 \newcommand*{\LWR@linkmediacatcodes}{%
10873     \catcode`\#=12%
10874     \catcode`\%=12%
10875 %   \catcode`\&=12% left alone for splitting flash variables
10876     \catcode`\~=12%
10877     \catcode`\_=12%
```

For babel-french:

```
10878     \LWR@hook@processingtags%
10879 }
```

\LWR@subhyperref {⟨URL⟩}

Starts a link for \LWR@hrefb. A group must have been opened first, with nulli-
fied catcodes. The text name is printed afterwards, after the group is closed and
catcodes restored.

```
10880 \NewDocumentCommand{\LWR@subhyperref}{m}{%
10881     \LWR@traceinfo{LWR@subhyperref !#1!}%
10882     \edef\tmpb{\detokenize\expandafter{#1}}%
10883     \LWR@HTMLsanitize@tmpb%
10884     \LWR@htmltag{%
10885         a href=\textquotedbl\tmpb\textquotedbl\ % space
10886         \LWR@addlinktitle % space
10887         target=\textquotedbl\_{}blank\textquotedbl\ % space
10888     }%
10889 }
```

\LWR@subhyperreftext@sanizited {⟨text⟩}

Finishes the hyperref for \LWR@hrefb. Catcodes must have been restored already.
To be used after \LWR@subhyperref, and after its group has been closed.

```
10890 \newcommand{\LWR@subhyperreftext@sanitized}[1]{%
10891     \edef\tmpb{#1}%
10892     \LWR@HTMLsanitize@tmpb%
10893     \tmpb%
10894     \LWR@htmltag{/a}%
10895     \LWR@ensuredoingapar%
10896 }
```

LWR@subhyperreftext@unsanitized {⟨*text*⟩}

> Finishes the hyperref for \LWR@hrefb. Catcodes must have been restored already.
> To be used after \LWR@subhyperref, and after its group has been closed.

```
10897 \newcommand{\LWR@subhyperreftext@unsanitized}[1]{%
10898     #1%
10899     \LWR@htmltag{/a}%
10900     \LWR@ensuredoingapar%
10901 }
```

\LWR@subhyperrefclass {⟨*URL*⟩} {⟨*text*⟩} {⟨*htmlclass*⟩}

```
10902 \NewDocumentCommand{\LWR@subhyperrefclass}{m +m m}{%
10903     \LWR@htmltag{%
10904         a % space
10905       href=\textquotedbl\begingroup\@sanitize#1\endgroup\textquotedbl\ % space
10906         class=\textquotedbl#3\textquotedbl\ % space
10907         \LWR@addlinktitle % space
10908     }\LWR@orignewline%
10909     #2%
10910     \LWR@htmltag{/a}%
10911     \LWR@ensuredoingapar%
10912 }
```

\LWR@href [⟨*options*⟩] {⟨*URL*⟩} {⟨*text*⟩}

> Create a link with accompanying text. The accompanyting text is sanitized for
> HTML.

```
10913 \DeclareDocumentCommand{\LWR@hrefb}{O{} m}{%
10914     \LWR@ensuredoingapar%
10915     \LWR@subhyperref{#2}%
10916     \endgroup% restore catcodes
```

> If use \LWR@subhyperreftext@sanitized here, some forms of text may not expand
> correctly, and thus break.

```
10917     \LWR@subhyperreftext@unsanitized% takes the following text as an argument
10918 }
10919
10920 \newrobustcmd*{\LWR@href}{%
10921     \begingroup%
10922     \LWR@linkcatcodes%
10923     \LWR@hrefb%
10924 }
```

> A version which sanitizes both the URL and the text. Used by \LWR@url.

```
10925 \DeclareDocumentCommand{\LWR@hrefb@sanitized}{O{} m}{%
10926     \LWR@ensuredoingapar%
10927     \LWR@subhyperref{#2}%
10928     \endgroup% restore catcodes
```

> Used by \LWR@url to sanitize the text argument before printing.

```
10929     \LWR@subhyperreftext@sanitized% takes the following text as an argument
```

```
10930 }
10931
10932 \newrobustcmd*{\LWR@href@sanitized}{%
10933     \begingroup%
10934     \LWR@linkcatcodes%
10935     \LWR@hrefb@sanitized%
10936 }
```

\LWR@href@partsanitized [⟨*options*⟩] {⟨*URL*⟩} {⟨*text*⟩}

> Create a link with accompanying text. The accompanyting text is not sanitized,
> for use internally with algorithmically derived tags.

```
10937 \DeclareDocumentCommand{\LWR@hrefb@partsanitized}{O{} m}{%
10938     \LWR@ensuredoingapar%
10939     \LWR@subhyperref{#2}%
10940     \endgroup% restore catcodes
10941     \LWR@subhyperreftext@unsanitized% takes the following text as an argument
10942 }
10943
10944 \newrobustcmd*{\LWR@href@partsanitized}{%
10945     \begingroup%
10946     \LWR@linkcatcodes%
10947     \LWR@hrefb@partsanitized%
10948 }
```

\LWR@nolinkurl {⟨*URL*⟩}

> Print the name of the link without creating the link:

```
10949 \newcommand*{\LWR@nolinkurlb}[1]{%
10950     \LWR@ensuredoingapar%
10951     \edef\tmpb{#1}%
10952     \LWR@HTMLsanitize@tmpb%
10953     \tmpb%
10954     \endgroup%
10955 }
10956
10957 \newrobustcmd*{\LWR@nolinkurl}{%
10958     \begingroup%
10959     \LWR@linkcatcodes%
10960     \LWR@nolinkurlb%
10961 }
```

\LWR@url {⟨*URL*⟩}

> Create a link whose text name is the address of the link.

> The url package may redefine \url, so it is \let to \LWR@urlahere and also rede-
> fined by lwarp-url.

```
10962 \DeclareDocumentCommand{\LWR@urlb}{m}{%
10963     \LWR@ensuredoingapar%
10964     \LWR@href@sanitized{#1}{#1}%
10965     \endgroup%
10966 }
10967
10968 \newrobustcmd*{\LWR@url}{%
```

```
10969     \begingroup%
10970     \LWR@linkcatcodes%
10971     \LWR@urlb%
10972 }
```

\LWR@subinlineimage {⟨*1: <alt> tag*⟩} {⟨*2: class*⟩} {⟨*3: filename*⟩} {⟨*4: extension*⟩} {⟨*5: css style*⟩} {⟨*6: aria role*⟩}

Creates the HTML tag for the image. Factored from lateximage.

To allow sanitization of the alt tag, if the alt tag is detokenized then it must also be expanded before being used here.

```
10973 \newcommand*{\LWR@subinlineimage}[6]{%
10974     \begingroup%
10975     \boolfalse{LWR@HTMLsanitize@tmpb@removebackslash}%
10976     \ifblank{#6}% ARIA role
10977         {\renewcommand*{\LWR@tempone}{}}%
10978      {\renewcommand*{\LWR@tempone}{role=\textquotedbl#6\textquotedbl\LWR@indentHTML}}%
10979     \ifblank{#1}% alt text
10980     {% alt is file name
10981         \LWR@htmltag{img \LWR@indentHTML
10982             src=\textquotedbl#3.#4\textquotedbl \LWR@indentHTML
10983             alt=\textquotedbl#3\textquotedbl \LWR@indentHTML
10984             \LWR@tempone
10985             style=\textquotedbl#5\textquotedbl \LWR@indentHTML
10986             class=\textquotedbl#2\textquotedbl \LWR@orignewline
10987         }%
10988     }%
10989     {% alt is as given
10990         \def\tmpb{#1}%
10991         \LWR@HTMLsanitize@tmpb%
10992         \LWR@htmltag{img \LWR@indentHTML
10993             src=\textquotedbl#3.#4\textquotedbl \LWR@indentHTML
10994             alt=\textquotedbl\tmpb\textquotedbl \LWR@indentHTML
10995             \LWR@tempone
10996             style=\textquotedbl#5\textquotedbl \LWR@indentHTML
10997             class=\textquotedbl#2\textquotedbl \LWR@orignewline
10998         }%
10999     }%
11000     \endgroup%
11001 }
```

```
11002 \end{warpHTML}
```

Table 17: Float data structures

For each <type> of float (figure, table, etc.) there exists the following:

counter <type>: A counter called <type>, such as figure, table.

\<type>name: Name. \figurename prints "Figure", etc.

\ext@<type>: File extension. \ext@figure prints "lof", etc.

\fps@<type>: Placement.

\the<type>: Number. \thetable prints the number of the table, etc.

\p@<type>: Parent's number. Prints the number of the [within] figure, etc.

\fnum@<type>: Prints the figure number for the caption.
        \<type>name \the<type>, "Figure 123".

\<type>: Starts the float environment. \figure or \begin{figure}

\end<type>: Ends the float environment. \endfigure or \end{figure}

\tf@<ext>: The LaTeX file identifier for the output file.

LWR@have<type>: A boolean remembering whether a \listof was requested for a float of this type.

File with extension lo<f,t,a-z>: An output file containing the commands to build the \listof<type> "table-of-contents" structure.

Cross-referencing names: For cleveref's \cref and related, \crefname and \Crefname assign human-readable names for references to this float type.

79 Floats

Floats are supported, although partially through emulation.

Table 17 shows the data structure associated with each <type> of float.

79.1 Float environment

for HTML output: 11003 \begin{warpHTML}

\LWR@floatbegin {⟨*type*⟩} [⟨*placement*⟩]   Begins a \newfloat environment.

11004 \NewDocumentCommand{\LWR@floatbegin}{m o}{%

Warn if starting a float inside a <span>:

11005   \LWR@spanwarninvalid{float}%

11006   \ifbool{FormatWP}{\newline}{}%
11007   \LWR@stoppars%

There is a new float, so increment the unique float counter:

```
11008     \addtocounter{LWR@thisautoid}{1}%
11009     \booltrue{LWR@freezethisautoid}%
```

```
11010     \begingroup%
```

Settings while inside the environment:

```
11011     \LWR@print@raggedright%
```

Open an HTML figure tag. The figure is assigned a class equal to its type, and another class according to the float package style, if used. Note that \csuse returns an empty string if \LWR@floatstyle@<type> is not defined.

```
11012   \LWR@htmltag{%
11013       figure id=\textquotedbl%
11014           \LWR@print@mbox{autoid-\arabic{LWR@thisautoid}}%
11015       \textquotedbl\ % space
11016       class=\textquotedbl#1 \@nameuse{LWR@floatstyle@#1}\textquotedbl%
11017   }%
11018   \ifbool{FormatWP}{%
11019       \LWR@orignewline%
11020       \LWR@BlockClassWP{}{}{wp#1}%
11021   }{}%
```

Update the caption type:

```
11022     \renewcommand*{\@captype}{#1}%
```

Mark the float for a word processor conversion:

```
11023     \LWR@startpars%
11024     \ifboolexpr{bool{FormatWP} and bool{WPMarkFloats}}{%
11025
11026     === begin #1 ===
11027
11028     }{}%
```

After each \LWR@floatbegin, look for \centering, etc next, using \LWR@floatalignment.

```
11029 }
```

For koma-script. The following does not work for tables.

```
11030 \AtBeginDocument{
11031
11032 \IfPackageLoadedTF{tocbasic}{
11033
11034 \appto\figure@atbegin{%
11035     \LWR@futurenonspacelet\LWR@mynexttoken\LWR@floatalignment%
11036 }
11037
11038 }{}% tocbasic
11039
11040 }% AtBeginDocument
```

\@xfloat Support packages which create floats directly.

\@xdlbfloat

Look for \centering, etc using \LWR@floatalignment.

```
11041 \AtBeginDocument{
11042     \def\@xfloat #1[#2]{%
11043         \LWR@floatbegin{#1}[#2]
11044         \LWR@futurenonspacelet\LWR@mynexttoken\LWR@floatalignment%
11045     }
11046     \def\@xdblfloat #1[#2]{%
11047         \LWR@floatbegin{#1}[#2]
11048         \LWR@futurenonspacelet\LWR@mynexttoken\LWR@floatalignment%
11049     }
11050 }
```

\LWR@floatend Ends a \newfloat environment.

```
11051 \newcommand*{\LWR@floatend}{%
```

If saw a \centering, finish the center environment:

```
11052     \LWR@endfloatalignment%
```

Mark the float end for a word processor conversion:

```
11053     \ifboolexpr{bool{FormatWP} and bool{WPMarkFloats}}{%
11054
11055     === end ===
11056
11057     }{}%
11058     \LWR@stoppars%
```

Close an HTML figure tag:

```
11059     \ifbool{FormatWP}{\endLWR@BlockClassWP}{}%
11060     \LWR@htmlelementend{figure}%
11061     \endgroup%
11062     \boolfalse{LWR@freezethisautoid}%
11063     \LWR@startpars%
11064     \ifbool{FormatWP}{\newline}{}%
11065 }
```

\end@float Support packages which create floats directly.

\end@dlbfloat

```
11066 \AtBeginDocument{
11067     \let\end@float\LWR@floatend
11068     \let\end@dblfloat\LWR@floatend
11069 }
```

79.2 Float tracking

LWR@thisautoid (*Ctr*) A sequential counter for all floats and theorems. This is used to identify the float or theorem then reference it from the List of Figures and List of Tables.

```
11070 \newcounter{LWR@thisautoid}
```

LWR@thisautoidWP (*Ctr*) A sequential counter for all word processor conversion <div>s. This is used to convince LIBREOFFICE to form a frame around this element.

```
11071 \newcounter{LWR@thisautoidWP}
```

LWR@freezethisautoid (*bool*) Prevents multiple increments of \LWR@thisautoid inside a float.

```
11072 \newbool{LWR@freezethisautoid}
11073 \boolfalse{LWR@freezethisautoid}
```

\LWR@forcenewautoidanchor Adds a new <autoid> anchor.

```
11074 \newcommand*{\LWR@forcenewautoidanchor}{%
11075     \addtocounter{LWR@thisautoid}{1}%
11076     \ifbool{LWR@doingapar}%
11077     {%
11078         \LWR@htmltag{a id=\textquotedbl%
11079             \LWR@print@mbox{autoid-\arabic{LWR@thisautoid}}}%
11080             \textquotedbl\ }% space
11081         \LWR@htmltag{/a }%
11082     }%
11083     {%
11084         \LWR@stoppars%
11085         \LWR@htmltag{a id=\textquotedbl%
11086             \LWR@print@mbox{autoid-\arabic{LWR@thisautoid}}}%
11087         \textquotedbl\ }% space
11088         \LWR@htmltag{/a }%
11089         \LWR@startpars%
11090     }%
11091 }
```

\LWR@newautoidanchor Sometimes adds a new <autoid> anchor.

```
11092 \newcommand*{\LWR@newautoidanchor}{%
11093     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
11094         {}%
11095         {\ifbool{LWR@freezethisautoid}{}{\LWR@forcenewautoidanchor}}%
11096 }
```

\@captype Remembers which float type is in use.

```
11097 \newcommand*{\@captype}{}
```

\LWR@floatalignmentname Set to center, flushleft, or flushright if saw \centering, \raggedright, or \raggedleft.

```
11098 \newcommand*{\LWR@floatalignmentname}{}
```

\LWR@floatalignment If sees a \centering, \raggedleft, or \raggedright, creates a center, flushright, or flushleft environment.

```
11099 \newcommand*{\LWR@floatalignment}{%
11100     \ifdefstrequal{\LWR@mynexttoken}{\centering}{%
11101         \center%
11102         \renewcommand*{\LWR@floatalignmentname}{center}%
11103     }{}%
```

```
11104     \ifdefstrequal{\LWR@mynexttoken}{\raggedright}{%
11105         \flushleft%
11106         \renewcommand*{\LWR@floatalignmentname}{flushleft}%
11107     }{}%
11108     \ifdefstrequal{\LWR@mynexttoken}{\raggedleft}{%
11109         \flushright%
11110         \renewcommand*{\LWR@floatalignmentname}{flushright}%
11111     }{}%
11112 }
```

\LWR@endfloatalignment Closes an environment from \LWR@floatalignment.

```
11113 \newcommand*{\LWR@endfloatalignment}{%
11114     \ifdefvoid{\LWR@floatalignmentname}%
11115         {}%
11116         {\@nameuse{end\LWR@floatalignmentname}}%
11117     \renewcommand*{\LWR@floatalignmentname}{}%
11118 }
```

79.3   Caption inside a float environment

\CaptionSeparator How to separate the float number and the caption text, if not defined by the user. In most cases, caption's settings are used instead.

```
11119 \AtBeginDocument{\providecommand*{\CaptionSeparator}{:~}}
```

\@caption   {⟨*posn*⟩} [⟨*name*⟩] {⟨*long name*⟩}

\@makecaption   {⟨*name and num*⟩} {⟨*text*⟩}

Prints the float type and number, the caption separator, and the caption text.

\@caption is provided here in case caption is not loaded, and is based on the nameref package.

```
11120 \AtBeginDocument{
11121     \IfPackageLoadedTF{caption}{}{
11122         \let\LWR@orig@caption\@caption%
11123         \long\def\@caption#1[#2]{%
```

Warn if using a caption inside a <span>:

```
11124             \LWR@spanwarnformat{caption}%

11125             \LWR@setlatestname{#2}%
11126             \LWR@orig@caption{#1}[{#2}]% also takes third argument
11127         }%
11128
11129         \renewcommand{\@makecaption}[2]{%
11130             \LWR@traceinfo{@makecaption}%
11131             \caption@begin{\@captype}%
11132             \LWR@isolate{#1}%
11133             \edef\LWR@tempone{#1}%
11134             \ifdefvoid{\LWR@tempone}{}{\CaptionSeparator}%
11135             \LWR@isolate{#2}%
```

```
11136            \caption@end%
11137            \LWR@traceinfo{@makecaption: done}%
11138        }%
11139    }
11140 }
```

79.4   Caption and LOF linking and tracking

When a new HTML file is marked in the LaTeX PDF file, or at the start of a new section, the LaTeX PDF page number at that point is stored in LWR@currentautosecfloatpage, (and the associated filename is remembered by the special LaTeX labels). This page number is used to generate an autopage HTML <id> in the HTML output at the start of the new HTML file or section. Meanwhile, there is a float counter used to generate an HTML autoid <id> at the start of the float itself in the HTML file. The autopage and autoid values to use for each float are written to the .lof, etc. files just before each float's entry. These values are used by \l@figure, etc. to create the HTML links in the List of Figures, etc.

LWR@nextautoid (*Ctr*)   Tracks autoid for floats. Tracks autopage for floats.

LWR@nextautopage (*Ctr*)
These are updated per float as the .lof, .lot file is read.

```
11141 \newcounter{LWR@nextautoid}
11142 \newcounter{LWR@nextautopage}
```

\LWRsetnextfloat   {⟨*autopage*⟩} {⟨*float autoid*⟩}

*_html.lof (*file*)   This is written to the *_html.lof or *_html.lot file just before each float's usual

*_html.lot (*file*)   entry. The autopage and the float's autoid are remembered for \l@figure to use when creating the HTML links.

```
11143 \newcommand*{\LWRsetnextfloat}[2]{%
11144    \setcounter{LWR@nextautopage}{#1}%
11145    \setcounter{LWR@nextautoid}{#2}%
11146 }
```

LWR@figcaption (*env.*) An HTML <figcaption> is not allowed in places where LaTeX does allow a figure caption, such as inside a longtable where the tabular has already started, or inside a center environment. Therefore, a <div> of class figurecaption is used instead.

```
11147 \newenvironment*{LWR@figcaption}
11148    {%
11149        \ifbool{FormatWP}{%
11150            \BlockClass[font-style:italic]{figurecaption}%
11151        }{%
11152            \BlockClass{figurecaption}%
11153        }%
```

Inside the caption, temporarily prevent underfull \hbox warnings, such as when the caption contains a math SVG image.

```
11154            \hbadness=10000\relax%
```

```
11155    }%
11156    {\endBlockClass}
```

\LWR@HTML@caption@begin {⟨*type*⟩}

> Low-level code to create HTML tags for captions.

> The print versions are from the caption package, if loaded.

```
11157 \newcommand*{\LWR@HTML@caption@begin}[1]
11158 {%
11159     \LWR@traceinfo{LWR@HTML@caption@begin}%
```

> Keep par and minipage changes local:

```
11160     \begingroup%
```

> No need for a minipage or \parbox inside the caption:

```
11161     \RenewDocumentEnvironment{minipage}{O{t} o O{t} m}{}{}%
11162     \RenewDocumentCommand{\parbox}{O{t} O{} O{t} m +m}{##5}%
```

> Enclose the original caption code inside an HTML tag:

```
11163     \LWR@figcaption%
11164     \LWR@traceinfo{LWR@HTML@caption@begin: about to LWR@origcaption@begin}%
11165     \LWR@print@caption@begin{#1}%
11166     \LWR@traceinfo{LWR@HTML@caption@begin: done}%
11167 }
```

\LWR@HTML@caption@end Low-level patches to create HTML tags for captions.

```
11168 \newcommand*{\LWR@HTML@caption@end}
11169 {%
11170     \LWR@traceinfo{LWR@HTML@caption@end}%
11171     \LWR@print@caption@end%
```

> Closing tag:

```
11172     \endLWR@figcaption%
11173     \endgroup%
11174     % \leavevmode% avoid bad space factor (0) error
11175     \LWR@traceinfo{LWR@HTML@caption@end: done}%
11176 }
```

\caption@begin Low-level patches to create HTML tags for captions. These are assigned \AtBeginDocument
\caption@end so that other packages which modify captions will have already been loaded before
saving the print-mode version.

> Print versions are provided here in case caption is not loaded.

```
11177 \AtBeginDocument{
11178     \providecommand{\caption@begin}[1]{}
11179     \LWR@formatted{caption@begin}
11180
11181     \providecommand{\caption@end}{}
11182     \LWR@formatted{caption@end}
11183 }
```

\captionlistentry   Tracks the float number for this caption used outside a float. Patched to create an
HTML anchor.

```
11184 \AtBeginDocument{%
11185 \IfPackageLoadedTF{caption}{
11186     \let\LWR@origcaptionlistentry\captionlistentry
11187
11188     \renewcommand*{\captionlistentry}{%
11189         \LWR@ensuredoingapar%
11190         \LWR@origcaptionlistentry%
11191     }
11192     \def\LWR@LTcaptionlistentry{%
11193         \LWR@ensuredoingapar%
11194         \LWR@forcenewautoidanchor%
11195         \bgroup%
11196         \@ifstar{\egroup\LWR@LT@captionlistentry}% gobble *
11197             {\egroup\LWR@LT@captionlistentry}%
11198     }%
11199
11200     \def\LWR@LT@captionlistentry#1{%
11201         \caption@listentry\@firstoftwo[\LTcaptype]{#1}%
11202     }%
11203 }% caption loaded
11204 {% caption not loaded
11205     \newcommand{\captionlistentry}[2][]{}%
11206     \newcommand{\LWR@LT@captionlistentry}[2][]{}%
11207 }
11208 }% AtBeginDocument
```

\addcontentsline   Patched to write the autopage and autoid before each float's entry. No changes
if writing .toc For a theorem, automatically defines \ext@<type> as needed, to
mimic and reuse the float mechanism.

f

```
11209 \let\LWR@origaddcontentsline\addcontentsline
11210
11211 \renewcommand*{\addcontentsline}[3]{%
11212     \ifstrequal{#1}{toc}{}{% not TOC
11213     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
11214         {}%
11215         {\LWR@newautoidanchor}%
11216     \ifcsvoid{ext@#2}{\csdef{ext@#2}{#1}}{}%
11217     \addtocontents{\@nameuse{ext@#2}}{%
11218         \protect\LWRsetnextfloat%
11219         {\arabic{LWR@currentautosecfloatpage}}%
11220         {\arabic{LWR@thisautoid}}%
11221     }%
11222 }% not TOC
11223     \LWR@origaddcontentsline{#1}{#2}{#3}%
11224 }
```

capt-of (*Pkg*)   Either package provides \captionof, which is later patched at the beginning of
caption (*Pkg*)

the document.

\captionof Patched to handle paragraph tags.

```
11225 \RequirePackage{capt-of}
11226
11227 \AtBeginDocument{
11228     \let\LWR@origcaptionof\captionof
11229
11230     \renewcommand*{\captionof}{%
11231         \LWR@stoppars%
11232         \LWR@origcaptionof%
11233     }
11234 }% AtBeginDocument


11235 \end{warpHTML}
```

80   Table of Contents, LOF, LOT

This section controls the generation of the TOC, LOF, and LOT.

The `.toc`, `.lof`, and `.lot` files are named by the source code \jobname.

In HTML, the printed tables are placed inside a `<div>` of class toc, lof, or lot.

A "sidetoc" is provided which prints a subset of the TOC on the side of each page other than the homepage.

The regular LaTeX infrastructure is used for TOC, along with some patches to generate HTML output.

for HTML output: 11236 \begin{warpHTML}

80.1   Reading and printing the TOC

\LWR@myshorttoc {⟨*toc/lof/lot/sidetoc*⟩}

Reads in and prints the TOC/LOF/LOT at the current position. While doing so, makes the @ character into a normal letter to allow formatting commands in the section names.

Unlike in regular LaTeX, the file is not reset after being read, since the sideTOC may be referred to again in each HTML page.

```
11237 \newcommand*{\LWR@myshorttoc}[1]{%
11238     \LWR@traceinfo{LWR@myshorttoc: #1}%
```

Only if the file exists:

```
11239     \IfFileExists{\jobname.#1}{%
11240         \LWR@traceinfo{LWR@myshorttoc: loading}%
```

⚠ Many of the commands in the file will have @ characters in them, so @ must be

made a regular letter.

```
11241          \begingroup%
11242          \makeatletter%
```

Disable \ref to avoid nested HTML references.

```
11243        \LetLtxMacro\ref\LWR@print@ref%
11244        \LWR@disablepinyin%
```

Read in the TOC file:

```
11245          \@input{\jobname.#1}%
11246          \endgroup%
11247      }%
11248      {}%
11249      \LWR@traceinfo{LWR@myshorttoc: done}%
11250 }
```

\LWR@subtableofcontents {⟨*toc/lof/lot*⟩} {⟨*sectionstarname*⟩}

Places a TOC/LOF/LOT at the current position.

```
11251 \NewDocumentCommand{\LWR@subtableofcontents}{m m}{%
```

Closes previous levels:

```
11252      \@ifundefined{chapter}%
11253          {\LWR@closeprevious{section}}%
11254          {\LWR@closeprevious{chapter}}%
```

Prints any pending footnotes so that they appear above the potentially large TOC:

```
11255      \LWR@printpendingfootnotes%
```

Place the list into its own chapter (if defined) or section:

```
11256      \@ifundefined{chapter}{\section*{#2}}{\chapter*{#2}}%
```

Create a new HTML nav containing the TOC/LOF/LOT:

```
11257      \LWR@htmlelementclass{nav}{#1}%
```

Create the actual list:

```
11258      \LWR@myshorttoc{#1}%
```

Close the nav:

```
11259      \LWR@htmlelementclassend{nav}{#1}%
11260 }
```

\@starttoc {⟨*ext*⟩}

Patch \@starttoc to encapsulate the TOC inside HTML tags:

```
11261 \let\LWR@orig@starttoc\@starttoc
11262
11263 \renewcommand{\@starttoc}[1]{
11264     \LWR@htmlelementclass{nav}{#1}%
11265     \LWR@orig@starttoc{#1}%
11266     \LWR@htmlelementclassend{nav}{#1}%
11267 }
```

LWR@copiedsidetoc (*bool*) Used to only copy the TOC file to the sidetoc a single time.

(listings and perhaps other packages would re-use \tableofcontents for their own purposes, causing the sidetoc to be copied more than once, and thus end up empty.)

```
11268 \newbool{LWR@copiedsidetoc}
11269 \boolfalse{LWR@copiedsidetoc}
```

\tableofcontents Patch \tableofcontents, etc. to print footnotes first. newfloat uses \listoffigures for all future float types.

```
11270 \AtBeginDocument{
11271
11272 \LetLtxMacro\LWR@origtableofcontents\tableofcontents
11273
11274 \renewcommand*{\tableofcontents}{%
```

Do not print the table of contents if formatting for a word processor, which will presumably auto-generate its own updated table of contents:

```
11275     \ifboolexpr{bool{FormatWP} and bool{WPMarkTOC}}{
11276
11277     === table of contents ===
11278
11279     }
11280     {
```

Copy the .toc file to .sidetoc for printing the sideTOC. The original .toc file is renewed when \tableofcontents is finished.

```
11281         \ifbool{LWR@copiedsidetoc}{}{%
11282             \LWR@copyfile{\jobname.toc}{\jobname.sidetoc}%
11283             \booltrue{LWR@copiedsidetoc}%
11284         }%
11285         \LWR@printpendingfootnotes
```

Disable \ref to avoid nested HTML references.

```
11286         \begingroup%
11287         \LetLtxMacro\ref\LWR@print@ref%
11288         \LWR@disablepinyin%
11289         \LWR@origtableofcontents%
11290         \endgroup%
11291     }
11292 }% \tableofcontents
11293
11294 }% AtBeginDocument
```

\listoffigures

```
11295 \let\LWR@origlistoffigures\listoffigures
11296
11297 \renewcommand*{\listoffigures}{
11298     \ifboolexpr{bool{FormatWP} and bool{WPMarkLOFT}}{
11299
11300     === list of figures ===
11301
11302     }
11303     {
11304         \LWR@printpendingfootnotes
```

Disable \ref to avoid nested HTML references.

```
11305         \begingroup%
11306         \LetLtxMacro\ref\LWR@print@ref%
11307         \LWR@disablepinyin%
11308         \LWR@origlistoffigures%
11309         \endgroup%
11310     }
11311 }
```

\listoftables

```
11312 \let\LWR@origlistoftables\listoftables
11313
11314 \renewcommand*{\listoftables}{
11315     \ifboolexpr{bool{FormatWP} and bool{WPMarkLOFT}}{
11316
11317     === list of tables ===
11318
11319     }
11320     {
11321         \LWR@printpendingfootnotes
```

Disable \ref to avoid nested HTML references.

```
11322         \begingroup%
11323         \LetLtxMacro\ref\LWR@print@ref%
11324         \LWR@disablepinyin%
11325         \LWR@origlistoftables%
11326         \endgroup%
11327     }
11328 }
```

80.2 TOC commands

\LWR@listof {⟨*type*⟩} {⟨*title*⟩}

Emulate the \listof command from the float package (section 281). Used to create lists of custom float types. Also used to redefine the standard LATEX \listoffigures and \listoftables commands, and in tocloft and memoir.

```
11329 \NewDocumentCommand{\LWR@listof}{m +m}{%
11330     \@ifundefined{l@#1}{%
```

```
11331        \csdef{l@#1}##1##2{\hypertocfloat{1}{#1}{\@nameuse{ext@#1}}{##1}{##2}}%
11332    }{}%
11333    \LWR@subtableofcontents{\@nameuse{ext@#1}}{#2}%
11334    \expandafter\newwrite\csname tf@\csname ext@#1\endcsname\endcsname%
11335    \immediate\openout \csname tf@\csname ext@#1\endcsname\endcsname%
11336        \jobname.\@nameuse{ext@#1}\relax%
11337 }
```

80.3   Side TOC

The "side TOC" is a table-of-contents positioned to the side.

It may be renamed by redefining \sidetocname, and may contain paragraphs.

Per table 18, CSS may be used to format the sideTOC.

Table 18: CSS related to the sideTOC

div.sidetoccontainer: The entire sideTOC.

div.sidetoctitle: The title.

div.sidetoccontents: The table of contents.

```
11338 \end{warpHTML}
```

for HTML & PRINT: `11339 \begin{warpall}`

SideTOCDepth (*Ctr*)   Controls how deep the side-TOC gets. Use a standard LaTeX section level similar to tocdepth. Warn if parts of the website may be inaccessible.

```
11340 \newcounter{SideTOCDepth}
11341 \setcounter{SideTOCDepth}{1}
11342
11343 \AtEndDocument{%
11344     \ifnumcomp{\value{SideTOCDepth}}{<}{\value{FileDepth}}{
11345         \PackageWarningNoLine{lwarp}
11346         {%
11347             SideTOCDepth is less than FileDepth,\MessageBreak
11348             so some website pages may be inaccessible%
11349         }
11350     }{}
11351 }
```

\sidetocname   Holds the default name for the sideTOC.

```
11352 \newcommand{\sidetocname}{Contents}
```

```
11353 \end{warpall}
```

for HTML output: `11354 \begin{warpHTML}`

\LWR@sidetoc   Creates the actual side-TOC.

```
11355 \newcommand*{\LWR@sidetoc}{%
```

```
11356        \LWR@forcenewpage
11357        \LWR@stoppars
11358
```

The entire sideтос is placed into a nav of class sidetoc.

```
11359        \LWR@htmlelementclass{div}{sidetoccontainer}
11360        \LWR@htmlelementclass{nav}{sidetoc}
11361
11362        \setcounter{tocdepth}{\value{SideTOCDepth}}
```

The title is placed into a <div> of class sidetoctitle, and may contain paragraphs.

```
11363        \begin{BlockClass}{sidetoctitle}
11364        \ifcsvoid{thetitle}{}{\InlineClass{sidetocthetitle}{\thetitle}\par}
11365        \sidetocname
11366        \end{BlockClass}
```

The table of contents is placed into a <div> of class sidetoccontents.

```
11367        \begin{BlockClass}{sidetoccontents}
11368        \LinkHome
11369
11370        \LWR@myshorttoc{sidetoc}
11371        \end{BlockClass}
11372        \LWR@htmlelementclassend{nav}{sidetoc}
11373        \LWR@htmlelementclassend{div}{sidetoccontainer}
11374 }
```

80.4   Low-level тос line formatting

\numberline {⟨*number*⟩}

(Called from each line in the .aux, .lof files.)

Record this section number for further use:

```
11375 \newcommand*{\LWR@numberline}[1]{%
11376        \LWR@sectionnumber{#1}\quad%
11377 }
11378
11379 \LetLtxMacro\numberline\LWR@numberline
```

\LWR@maybetocdata   Replaced by tocdata. Adds author name.

```
11380 \newcommand*{\LWR@maybetocdata}{}
```

\hypertoc {⟨*1: depth*⟩} {⟨*2: type*⟩} {⟨*3: name*⟩} {⟨*4: page*⟩}

Called by \l@section, etc. to create a hyperlink to a section.

The autopage label is always created just after the section opens.

#1: Depth.

#2: section, subsection, etc.

#3: The text of the caption.

#4: Page number, which for HTML is the LWR@currentautosecpage when the section was created.

```
11381 \NewDocumentCommand{\hypertoc}{m m +m m}{%
11382     \LWR@traceinfo{hypertoc !#1!#2!#3!#4!}%
```

Respond to tocdepth:

```
11383     \ifnumcomp{#1}{>}{\value{tocdepth}}%
11384         {}%
11385         {%
11386             \LWR@startpars%
```

Create an HTML link to <filename>#autosec-(page), with the name, of the given HTML class.

\BaseJobname is added to the label in case xr or xr-hyper are used.

```
11387             \LWR@subhyperrefclass{%
11388                 \LWR@htmlrefsectionfilename{\BaseJobname-autopage-#4}%
11389                     \LWR@origpound\LWR@print@mbox{autosec-#4}%
11390             }{#3}{toc#2}%

11391                 \LWR@maybetocdata%

11392                 \LWR@stoppars%
11393         }%
11394     \LWR@traceinfo{hypertoc done}%
11395 }
```

lofdepth (*Ctr*)   TOC depth for figures.

```
11396 \IfClassLoadedTF{memoir}{}{
11397     \newcounter{lofdepth}
11398     \setcounter{lofdepth}{1}
11399 }
```

lotdepth (*Ctr*)   TOC depth for tables.

```
11400 \IfClassLoadedTF{memoir}{}{
11401     \newcounter{lotdepth}
11402     \setcounter{lotdepth}{1}
11403 }
```

\hypertocfloat {⟨*1: depth*⟩} {⟨*2: type*⟩} {⟨*3: ext of parent*⟩} {⟨*4: caption*⟩} {⟨*5: page*⟩}

#1 is depth

#2 is figure, table, etc.

#3 is lof, lot, of the parent.

#4 the text of the caption

#5 page number

```
11404 \newcommand{\hypertocfloat}[5]{%
```

If some float-creation package has not yet defined the float type's lofdepth counter, etc, define it here:

```
11405    \@ifundefined{c@#3depth}{%
11406        \newcounter{#3depth}%
11407        \setcounter{#3depth}{1}%
11408    }{}%
```

Respond to lofdepth, etc.:

```
11409    \LWR@traceinfo{hypertocfloat depth is #1 #3depth is \arabic{#3depth}}%
11410    \ifthenelse{\cnttest{#1}{<=}{\arabic{#3depth}}}%
11411        {%
11412            \LWR@startpars%
```

Create an HTML link to filename#autoid-(float number), with text of the caption, of the given HTML class.

\BaseJobname is added to the label in case xr or xr-hyper are used.

```
11413            \LWR@subhyperrefclass{%
11414            \LWR@htmlrefsectionfilename{%
11415                \BaseJobname-autopage-\arabic{LWR@nextautopage}%
11416            }%
11417            \LWR@origpound\LWR@print@mbox{autoid-\arabic{LWR@nextautoid}}}%
11418            {#4}{toc#2}%


11419            \LWR@maybetocdata%


11420            \LWR@stoppars%
11421        }%
11422        {}%
11423 }
```

Automatically called by \contentsline:

\l@book {⟨*name*⟩} {⟨*page*⟩}

Uses \DeclareDocumentCommand in case the class does not happen to have a \book.

```
11424 \DeclareDocumentCommand{\l@book}{m m}{\hypertoc{-2}{book}{#1}{#2}}
```

\l@part {⟨*name*⟩} {⟨*page*⟩}

Uses \DeclareDocumentCommand in case the class does not happen to have a \part.

```
11425 \DeclareDocumentCommand{\l@part}{m m}{\hypertoc{-1}{part}{#1}{#2}}
```

\l@chapter {⟨*name*⟩} {⟨*page*⟩}

Uses \DeclareDocumentCommand in case the class does not happen to have a \chapter.

```
11426 \@ifundefined{chapter}
11427 {}
11428 {
11429 \DeclareDocumentCommand{\l@chapter}{m m}
11430     {\hypertoc{0}{chapter}{#1}{#2}}
11431 }
```

\l@section {⟨*name*⟩} {⟨*page*⟩}

```
11432 \renewcommand{\l@section}[2]{\hypertoc{1}{section}{#1}{#2}}
```

\l@subsection {⟨*name*⟩} {⟨*page*⟩}

```
11433 \renewcommand{\l@subsection}[2]{\hypertoc{2}{subsection}{#1}{#2}}
```

\l@subsubsection {⟨*name*⟩} {⟨*page*⟩}

```
11434 \renewcommand{\l@subsubsection}[2]{\hypertoc{3}{subsubsection}{#1}{#2}}
```

\l@paragraph {⟨*name*⟩} {⟨*page*⟩}

```
11435 \renewcommand{\l@paragraph}[2]{\hypertoc{4}{paragraph}{#1}{#2}}
```

\l@subparagraph {⟨*name*⟩} {⟨*page*⟩}

```
11436 \renewcommand{\l@subparagraph}[2]{\hypertoc{5}{subparagraph}{#1}{#2}}
```

\l@figure {⟨*name*⟩} {⟨*page*⟩}

```
11437 \renewcommand{\l@figure}[2]{\hypertocfloat{1}{figure}{lof}{#1}{#2}}
```

\l@table {⟨*name*⟩} {⟨*page*⟩}

```
11438 \renewcommand{\l@table}[2]{\hypertocfloat{1}{table}{lot}{#1}{#2}}
```

```
11439 \end{warpHTML}
```

81 Index and glossary

See:
http://tex.stackexchange.com/questions/187038/
        how-to-mention-section-number-in-index-created-by-imakeidx

Index links are tracked by the counter LWR@autoindex. This counter is used to create a label for each index entry, and a reference to this label for each entry in

the index listing. This method allows each index entry to link directly to its exact position in the document.

11440 `\begin{warpHTML}`

11441 `\newcounter{LWR@autoindex}`
11442 `\setcounter{LWR@autoindex}{0}`
11443
11444 `\newcounter{LWR@autoglossary}`
11445 `\setcounter{LWR@autoglossary}{0}`

`\IndexPageSeparator`   User-adjustable delimiters for page and range separators in the `*.ind` files.
`\IndexRangeSeparator`
11446 `\newcommand*{\IndexPageSeparator}{, }`
11447 `\newcommand*{\IndexRangeSeparator}{--}`

theindex (*env.*)

11448 `\@ifundefined{chapter}`
11449 `    {\newcommand*{\LWR@indexsection}[1]{\section*{#1}}}`
11450 `    {\newcommand*{\LWR@indexsection}[1]{\chapter*{#1}}}`
11451
11452
11453 `\AtBeginDocument{`
11454
11455 `\renewenvironment*{theindex}{%`
11456 `    \LWR@indexsection{\indexname}%`
11457 `    \LetLtxMacro\item\LWR@indexitem%`
11458 `    \LetLtxMacro\subitem\LWR@indexsubitem%`
11459 `    \LetLtxMacro\subsubitem\LWR@indexsubsubitem%`
11460 `}{}`
11461
11462 `}% AtBeginDocument`

`\LWR@indexitem` [⟨*index key*⟩]   The optional argument is added to support repeatindex.

11463 `\newcommand{\LWR@indexitem}[1][\@empty]{`
11464
11465 `    \InlineClass{indexitem}{\LWR@htmlcomment{}}#1%`
11466 `}`

`\LWR@indexsubitem`

11467 `\newcommand{\LWR@indexsubitem}{`
11468
11469 `    \InlineClass{indexsubitem}{\LWR@htmlcomment{}}%`
11470 `}`

`\LWR@indexsubsubitem`

11471 `\newcommand{\LWR@indexsubsubitem}{`
11472
11473 `    \InlineClass{indexsubsubitem}{\LWR@htmlcomment{}}%`
11474 `}`

\LWR@xindex@modifyentry {⟨*indexing term*⟩}

> If using *xindex*, modifies the pipe character to become \hyperindexformat. The in-
> dexing term is split into two argument at the pipe, then fed to \LWR@xindex@modifyentrysub.

```
11475 \NewDocumentCommand{\LWR@xindex@modifyentry}{>{\SplitArgument{1}{|}}m}
11476     {\LWR@xindex@modifyentrysub#1}
```

Handle left and right parenthesis range argument, or add a hyperindexformat
clause.

```
11477 \newcommand*{\LWR@xindex@modifyentrysub}[2]{%
11478     \edef\LWR@tempone{#1}%
11479     \edef\LWR@temptwo{#2}%
11480     \IfValueTF{#2}{%
11481         \ifx#2(%
11482             \appto\LWR@tempone{|(}%
11483         \else%
11484             \ifx#2)%
11485                 \appto\LWR@tempone{|)}%
11486             \else%
11487                 \appto\LWR@tempone{%
11488                     |hyperindexformat\LWRleftbrace%
11489                     \LWRbackslash#2%
11490                     \LWRrightbrace%
11491                 }%
11492             \fi%
11493         \fi%
11494     }%
11495     {}%
11496 }
```

LWR@xindex@tricked (*bool*)   Used to track xindex creation. See next.

```
11497 \newbool{LWR@xindex@tricked}
11498 \boolfalse{LWR@xindex@tricked}
```

\@wrindex {⟨*indexing term*⟩}   Redefined to write the LWR@autoindex counter instead of
page.

> If using *xindex*, the first line is a comment including a special phrase which tricks
> *xindex* into thinking that hyperref was used.

```
11499 \def\LWR@wrindex#1{%
11500     \ifbool{LWR@xindex}{%
11501         \ifbool{LWR@xindex@tricked}{}{%
11502             \protected@write\@indexfile{}%
11503             {%
11504                 \LWRpercent\space hyperpage\LWRrightbrace%
11505                 \LWRpercent\space trick xindex to assume hyperref%
11506             }%
11507             \global\booltrue{LWR@xindex@tricked}%
11508         }%
11509         \LWR@xindex@modifyentry{#1}%
11510     }{%
11511         \def\LWR@tempone{#1}%
11512     }%
11513     \addtocounter{LWR@autoindex}{1}%
```

```
11514     \protected@write\@indexfile{}%
11515   {\string\indexentry{\LWR@tempone}{\arabic{LWR@autoindex}}}%
```

The label is assigned after the file write to avoid conflict with cleveref.

```
11516     \label{LWRindex-\arabic{LWR@autoindex}}%
11517     \endgroup%
11518     \@esphack%
11519 }
11520
11521 \AtBeginDocument{
11522 \let\@wrindex\LWR@wrindex
11523 }
```

\@wrglossary {⟨*term*⟩}   Redefined to write the LWR@autoglossary counter instead of page.

```
11524 \def\@wrglossary#1{%
11525     \addtocounter{LWR@autoglossary}{1}%
11526     \LWR@new@label{LWRglossary-\theLWR@autoglossary}%
11527     \protected@write\@glossaryfile{}%
11528     {\string\glossaryentry{#1}{\theLWR@autoglossary}}%
11529     \endgroup%
11530     \@esphack%
11531 }
```

\LWR@indexnameref@anonref {⟨*LWR@autoindex*⟩}

Displays a reference link where there no \ref available.

```
11532 \newcommand*{\LWR@indexnameref@anonref}[1]{%
11533     \LWR@startref{LWRindex-#1}%
11534     (*)%
11535     \LWR@htmltag{/a}%
11536 }
```

\LWR@indexnameref@ref {⟨*LWR@autoindex*⟩}

Creates \ref-style index references. To avoid an unwanted space if there is nothing to reference, the reference is checked first.

```
11537 \newcommand*{\LWR@indexnameref@ref}[1]{%
11538     \edef\LWR@thisref{\csuse{r@LWRindex-#1}}%
11539     \ifdefvoid{\LWR@thisref}{}{%
11540         \edef\LWR@thisref{\expandafter\@firstoffive\LWR@thisref}%
11541         \ifdefvoid{\LWR@thisref}%
11542             {\LWR@indexnameref@anonref{#1}}%
11543             {\ref{LWRindex-#1}}%
11544     }%
11545 }
```

\LWR@indexnameref@refnameref {⟨*LWR@autoindex*⟩}

Creates \ref-style index references. To avoid an unwanted space if there is nothing to reference, the reference is checked first. For links to starred or ?? objects, only the name is used.

```
11546 \newcommand*{\LWR@indexnameref@refnameref}[1]{%
```

```
11547    \edef\LWR@thisref{\csuse{r@LWRindex-#1}}%
11548    \ifdefvoid{\LWR@thisref}{}{%
11549        \edef\LWR@thisref{\expandafter\@firstoffive\LWR@thisref}%
11550        \ifdefvoid{\LWR@thisref}{}{%
11551            \ifdefstring{\LWR@thisref}{(*)}%
11552                {}%
11553                {\ref{LWRindex-#1} }% space
11554        }%
11555    }%
11556    \nameref{LWRindex-#1}%
11557 }
```

\LWR@indexnameref@cref {⟨*LWR@autoindex*⟩}

> Creates \cref-style index references. If no numbered reference is available,
> a \nameref is used instead. If the reference is ??, which will be changed by
> \LWR@indexnameref to become (*), then the link is changed to show (*).

```
11558 \newcommand*{\LWR@indexnameref@cref}[1]{%
11559    \edef\LWR@thisref{\csuse{r@LWRindex-#1}}%
11560    \ifdefvoid{\LWR@thisref}{%
11561        \nameref{LWRindex-#1}%
11562    }{%
11563        \edef\LWR@thisref{\expandafter\@firstoffive\LWR@thisref}%
11564        \ifdefvoid{\LWR@thisref}{%
11565            \nameref{LWRindex-#1}%
11566        }{%
11567            \ifdefstring{\LWR@thisref}{(*)}{%
11568                \LWR@indexnameref@anonref{#1}%
11569            }{%
11570                \cref{LWRindex-#1}%
11571            }%
11572        }%
11573    }%
11574 }
```

\LWR@indexnameref@crefnameref {⟨*LWR@autoindex*⟩}

> Creates \cref-style index references. If no numbered reference is available,
> a \nameref is used instead. If the reference is ??, which will be changed by
> \LWR@indexnameref to become (*), then the link is changed to show only the
> name.

```
11575 \newcommand*{\LWR@indexnameref@crefnameref}[1]{%
11576    \edef\LWR@thisref{\csuse{r@LWRindex-#1}}%
11577    \ifdefvoid{\LWR@thisref}%
11578        {}%
11579        {%
11580            \edef\LWR@thisref{\expandafter\@firstoffive\LWR@thisref}%
11581            \ifdefvoid{\LWR@thisref}%
11582                {}%
11583                {%
11584                    \ifdefstring{\LWR@thisref}{(*)}%
11585                        {}%
11586                        {\cref{LWRindex-#1}} % space
11587                }%
11588        }%
11589    \nameref{LWRindex-#1}%
11590 }
```

\LWR@indexnameref {⟨*LWR@autoindex*⟩}

>   Creates a hyperlink based on the given entry's autoindex.

```
11591 \newcommand*{\LWR@indexnameref}[1]{%
11592     {% group
```

>   Temporarily redefine caption's \caption@xref because it was printing ?? in the
>   indexes, and also causing error on expansion:

```
11593         \ifdef{\caption@xref}{%
11594             \renewcommand*{\caption@xref}[2]{(*)}%
11595         }{}%

11596         \ifdefstring{\LWR@IndexRef}{ref}{%
11597             \LWR@indexnameref@ref{#1}%
11598         }{%
11599         \ifdefstring{\LWR@IndexRef}{nameref}{%
11600             \nameref{LWRindex-#1}%
11601         }{%
11602         \ifdefstring{\LWR@IndexRef}{refnameref}{%
11603             \LWR@indexnameref@refnameref{#1}%
11604         }{%
11605         \ifdefstring{\LWR@IndexRef}{cref}{%
11606             \LWR@indexnameref@cref{#1}%
11607         }{%
11608         \ifdefstring{\LWR@IndexRef}{crefnameref}{%
11609             \LWR@indexnameref@crefnameref{#1}%
11610         }{%
11611         \ifdefstring{\LWR@IndexRef}{autoref}{%
11612             \LWR@indexnameref@cref{#1}%
11613         }{% text string
11614             \LWR@startref{LWRindex-#1}%
11615             \LWR@IndexRef%
11616             \LWR@htmltag{/a}%
11617         }}}}}}%
11618     }% group
11619 }
```

\LWR@doindexentrysubsub {⟨*range start: LWR@autoindex, or macros.*⟩} {⟨*range end or blank*⟩}

>   Creates a hyperlink, or handles \see, \textbf, etc.

```
11620 \newrobustcmd{\LWR@doindexentrysubsub}[2]{%
11621     \IfInteger{#1}%
11622         {\LWR@indexnameref{#1}}%
11623         {#1}%
11624     \IfValueT{#2}{%
11625         \IndexRangeSeparator%
11626         \IfInteger{#2}%
11627             {\LWR@indexnameref{#2}}%
11628             {#2}%
11629     }%
11630 }
```

\LWR@doindexentrysub {⟨*range delimiter*⟩} {⟨*LWR@autoindex or macros, possible a range*⟩}

```
11631 \NewDocumentCommand{\LWR@doindexentrysub}{m >{\SplitArgument{1}{#1}}m}
```

11632   {\LWR@doindexentrysubsub#2}

\LWR@doindexentry {⟨*LWR@autoindex or macros, possible a range*⟩}

11633 \newcommand*{\LWR@doindexentry}[1]{%
11634   \relax% required
11635   \expandafter\LWR@doindexentrysub\expandafter{\IndexRangeSeparator}{#1}%
11636 }

\LWR@hyperindexrefnullified   Handles macros commonly seen inside an \index entry. Each macro is redefined to create and format a link to its entry.

⚠   index formatting   To handle additional macros:

      \appto\LWR@hyperindexrefnullified{. . . }

11637 \newcommand{\LWR@hyperindexrefnullified}{%
11638   \renewrobustcmd{\emph}[1]{\LWR@HTML@emph{\LWR@doindexentry{##1}}}%
11639   \renewrobustcmd{\textbf}[1]{\LWR@HTML@textbf{\LWR@doindexentry{##1}}}%
11640   \renewrobustcmd{\texteb}[1]{\LWR@HTML@texteb{\LWR@doindexentry{##1}}}%
11641   \renewrobustcmd{\textlg}[1]{\LWR@HTML@textlg{\LWR@doindexentry{##1}}}%
11642   \renewrobustcmd{\textrm}[1]{\LWR@HTML@textrm{\LWR@doindexentry{##1}}}%
11643   \renewrobustcmd{\textsf}[1]{\LWR@HTML@textsf{\LWR@doindexentry{##1}}}%
11644   \renewrobustcmd{\texttt}[1]{\LWR@HTML@texttt{\LWR@doindexentry{##1}}}%
11645   \renewrobustcmd{\textup}[1]{\LWR@HTML@textup{\LWR@doindexentry{##1}}}%
11646   \renewrobustcmd{\textsc}[1]{\LWR@HTML@textsc{\LWR@doindexentry{##1}}}%
11647   \renewrobustcmd{\textulc}[1]{\LWR@HTML@textulc{\LWR@doindexentry{##1}}}%
11648   \renewrobustcmd{\textsi}[1]{\LWR@HTML@textsi{\LWR@doindexentry{##1}}}%
11649   \renewrobustcmd{\textit}[1]{\LWR@HTML@textit{\LWR@doindexentry{##1}}}%
11650   \renewrobustcmd{\textsl}[1]{\LWR@HTML@textsl{\LWR@doindexentry{##1}}}%
11651 }

\hyperindexref {⟨*list of LWR@autoindex, commas, and ranges*⟩}

\hyperindexref{LWR@autoindex} is inserted into *.ind by the *makeindex* style file lwarp.ist or the *xindy* style file lwarp.xdy. For *xindex*, \hyperpage is inserted, which is \let to \hyperindexref. For *gindex*, \addindexitem and related are inserted, which are defined to use \hyperindexref.

The argument is split at commas, and also for ranges, then passed to \LWR@hyperindexrefsub.

11652 \newcommand*{\hyperindexref}[1]{%
11653   \relax% required
11654   \expandafter\LWR@hyperindexref@comma\expandafter{\IndexPageSeparator}{#1}%
11655 }

\LWR@hyperindexref@comma {⟨*separator*⟩} {⟨*list of args*⟩}

The list is split at commas, and passed to \LWR@hyperindexref@@comma.

11656 \NewDocumentCommand{\LWR@hyperindexref@comma}
11657   {m >{\SplitList{#1}} m}
11658   {%

Used to place the separtor between each entry, but not before the first.

```
11659          \def\LWR@hyperindexref@thiscomma{}%
11660          \def\LWR@hyperindexref@nextcomma{#1}%
```

Each comma-delimited entry is now passed individually to \LWR@hyperindexref@@comma.

```
11661          \ProcessList{#2}\LWR@hyperindexref@@comma%
11662     }
```

\LWR@hyperindexref@@comma {⟨*arg, perhaps with a range*⟩}

A comma separator is placed if not the first item, then the range is parsed.

```
11663 \newcommand*{\LWR@hyperindexref@@comma}[1]{%
11664     \LWR@hyperindexref@thiscomma%
11665     \renewcommand{\LWR@hyperindexref@thiscomma}{\LWR@hyperindexref@nextcomma}%
11666     \expandafter\LWR@hyperindexref@range\expandafter{\IndexRangeSeparator}{#1}%
11667 }
```

\LWR@hyperindexref@range {⟨*range delimiter*⟩} {⟨*arg*⟩}

```
11668 \NewDocumentCommand{\LWR@hyperindexref@range}
11669     {m >{\SplitArgument{1}{#1}} m}
11670     {\LWR@hyperindexrefsub#2}
```

\LWR@hyperindexrefsub {⟨*range start: LWR@autoindex*⟩} {⟨*range end, or* -NoValue-⟩}

Handles the start and end of a range, if applicable.

```
11671 \newcommand*{\LWR@hyperindexrefsub}[2]{%
11672     \LWR@hyperindexrefsubtwo{#1}%
11673     \IfValueT{#2}{%
11674         \IndexRangeSeparator%
11675         \LWR@hyperindexrefsubtwo{#2}%
11676     }%
11677 }
```

\LWR@hyperindexrefsubtwo {⟨*LWR@autoindex*⟩}

```
11678 \newcommand*{\LWR@hyperindexrefsubtwo}[1]{%
```

In long index lines with numerous entries, *makeindex* can insert a newline before the page number, resulting in an extra space before the first digit. If the first character is a space, remove it first.

```
11679     \edef\LWR@tempone{#1}%
11680     \IfBeginWith{\LWR@tempone}{ }{%
11681         \StrGobbleLeft{\LWR@tempone}{1}[\LWR@tempone]%
11682     }{}%
```

If a numeric entry, create a link. If not numeric, such as \see, use the entry as-is. \emph, \textit, etc. have been redefined above to create and format the entry.

```
11683     \IfInteger{\LWR@tempone}%
11684         {\LWR@indexnameref{\LWR@tempone}}%
11685         {%
11686             \begingroup%
```

```
11687            \LWR@hyperindexrefnullified%
11688            #1%
11689            \endgroup%
11690        }%
11691 }
```

\hyperpage Emulate hyperref.

```
11692 \LetLtxMacro\hyperpage\hyperindexref
```

\nohyperpage Emulate hyperref.

```
11693 \def\nohyperpage#1{}
```

\hyperindexformat Emulate hyperref.

```
11694 \def\hyperindexformat#1#2{%
11695     #1{\hyperpage{#2}}%
11696 }%
```

```
11697 \end{warpHTML}
```

for PRINT output: A null command for print mode, in case hyperref was not used:

```
11698 \begin{warpprint}
11699 \newcommand{\hyperindexref}[1]{#1}
11700 \end{warpprint}
```

for HTML & PRINT: For the glossaries package, try to prevent an error where \glo@name was not found:

```
11701 \begin{warpall}
11702 \providecommand{\glo@name}{}
11703 \end{warpall}
```

82   Bibliography presentation

for HTML output: 11704 \begin{warpHTML}

\bibliography {⟨*filenames*⟩}   At one time this was modified to read \BaseJobname.bbl, which meant the HTML version could not resolve until the print version was also present. This also confused multibib. It has been reverted to the original to use \jobname.bbl.

\@biblabel {⟨*text-refnumber*⟩}

```
11705 \renewcommand{\@biblabel}[1]{[#1]\quad}
```

thebibliography (*env.*) To emphasize document titles in the bibliography, the following redefines \em inside thebibliography to gather everything until the next closing brace, then display these tokens with \textit.

Adapted from embracedef.sty, *which is by* Takayuki Yato*:*
    https://gist.github.com/zr-tex8r/b72555e3e7ad2f0a37f1

```
11706 \AtBeginDocument{
11707
11708 \AtBeginEnvironment{thebibliography}{
11709
11710 \providecommand*{\LWR@newem}[1]{\textit{#1}}
11711
11712 \renewrobustcmd{\em}{%
11713   \begingroup
11714     \gdef\LWR@em@after{\LWR@em@finish\LWR@newem}%
11715     \afterassignment\LWR@em@after
11716     \toks@\bgroup
11717 }
11718
11719 \def\LWR@em@finish#1{%
11720     \xdef\LWR@em@after{\noexpand#1{\the\toks@}}%
11721   \endgroup
11722   \LWR@em@after\egroup
11723 }
11724
11725 }% \AtBeginEnvironment{thebibliography}
11726
11727 }% \AtBeginDocument
```

```
11728 \end{warpHTML}
```

83   Restoring original formatting

11729 \begin{warpHTML}

\LWR@restoreMathJaxformatting A few macros (ref: tcolorbox) must be treated separately while printing the HTML comment for a MathJax expression. These are set here, to which other functions may be appended.

```
11730 \newcommand*{\LWR@restoreMathJaxformatting}{}
```

\LWR@restoreorigformatting Used to temporarily restore the print-mode meaning of a number of formatting, graphics, and symbols-related macros while generating SVG math or a lateximage.

Must be used inside a group.

Sets \LWR@formatting to print until the end of the group.

A number of packages will \appto additional actions to this macro.

Various packages add to this macro using \appto.

```
11731   \newcommand*{\LWR@restoreorigformatting}{%
11732     \LWR@traceinfo{LWR@restoreorigformatting}%
```

Numerous macros change their print/HTML meaning depending on \LWR@formatting:

```
11733     \renewcommand*{\LWR@formatting}{print}%
11734     \linespread{1}%
```

```
11735    \setbool{LWR@doingparhooks}{false}%
```

```
11736    \def\color@endgroup{\endgraf\endgroup}%
```

```
11737    \LetLtxMacro\hfil\LWR@orighfil%
11738    \let\hss\LWR@orighss%
11739    \let\llap\LWR@origllap%
11740    \let\rlap\LWR@origrlap%
11741    \let\hfilneg\LWR@orighfilneg%
```

```
11742    \let\,\LWR@origcomma% disable HTML short unbreakable space
11743    \let\textless\LWR@origtextless%
11744    \let\textgreater\LWR@origtextgreater%
```

```
11745    \let\&\LWR@origampersand%
11746    \let\%\LWR@origpercent%
```

```
11747    \LetLtxMacro\em\LWR@origem%
11748    \LetLtxMacro\normalfont\LWR@orignormalfont%
11749    \let\sp\LWR@origsp%
11750    \let\sb\LWR@origsb%
11751    \LetLtxMacro\underline\LWR@origunderline%
11752    \LetLtxMacro~\LWR@origtilde%
11753    \LetLtxMacro\nobreakspace\LWR@orignobreakspace%
```

\endtabular must be restored to its original, instead of relying on lwarp's \LWR@formatted mechanism:

```
11754    \LetLtxMacro\endtabular\LWR@origendtabular%
11755    \csletcs{endtabular*}{LWR@origendtabular*}%
```

```
11756    \LetLtxMacro\noalign\LWR@orignoalign%
11757    \LetLtxMacro\hline\LWR@orighline%
```

```
11758    \let\newline\LWR@orignewline%
11759    \LetLtxMacro\includegraphics\LWR@origincludegraphics%
```

```
11760    \LetLtxMacro\@ensuredmath\LWR@origensuredmath%
```

```
11761    \let\math\LWR@orig@math%
11762    \let\endmath\LWR@orig@endmath%
11763    \let\displaymath\LWR@orig@displaymath%
11764    \let\enddisplaymath\LWR@orig@enddisplaymath%
11765 %
11766    \LWR@restoreorigaccents%
11767    \LWR@restoreoriglists%
```

```
11768    \let\@mpfootnotetext\LWR@orig@mpfootnotetext%
```

```
11769    \LWR@hook@processingtags%
```

To enable MATHJAX-specific nullification, used for tcolorbox:

```
11770    \ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
11771        {\LWR@restoreMathJaxformatting}%
11772        {}%
11773 }
```

```
11774 \end{warpHTML}
```

84 Nullifying filename formatting

The following are used to nullify certain macros and environments while convert-
ing section names to file names.

`11775 \begin{warpHTML}`

Also commonly used are \@empty, \@gobble, and \@firstofone.

```
11776 \newcommand*{\LWR@dash}{-}
```

\LWR@nullfonts Removes formatting during filename operations, file references, and HTML com-
ments.

⚠ Use only inside a group.

The following are *not* made robust, since they must be expanded to their nullified
versions.

```
11777 \catcode`\$=\active% redefining $ below
11778 \catcode`\_=12% redefining \_ below
11779 \newcommand*{\LWR@nullfonts}{%
```

Various built-in symbols.

```
11780    \renewcommand*{\$}{-}%
11781    \renewcommand*{\%}{-}%
11782    \renewcommand*{\_}{-}%
11783    \renewcommand*{\}}{-}%
11784    \renewcommand*{\{}{-}%
11785    \renewcommand*{\&}{-}% used to be 'and'
11786    \renewcommand*{\#}{-}%
11787    \renewcommand*{\,}{-}%
11788    \renewcommand*{~}{-}%
11789 %
11790 % accents:
11791    \renewcommand*{\`}[1]{##1}%
11792    \renewcommand*{\'}[1]{##1}%
11793    \renewcommand*{\^}[1]{##1}%
11794    \renewcommand*{\~}[1]{##1}%
11795    \renewcommand*{\=}[1]{##1}%
11796    \renewcommand*{\u}[1]{##1}%
11797    \renewcommand*{\.}[1]{##1}%
11798    \renewcommand*{\"}[1]{##1}%
11799    \renewcommand*{\H}[1]{##1}%
11800    \renewcommand*{\v}[1]{##1}%
11801    \renewcommand*{\d}[1]{##1}%
11802    \renewcommand*{\c}[1]{##1}%
```

```
11803        \renewcommand*{\b}[1]{##1}%
11804        \renewcommand*{\t}[1]{##1}%
11805 %
11806        \let\newline\LWR@dash%
11807        \let\textasciicircum\LWR@dash%
11808        \let\textasciitilde\LWR@dash%
11809        \let\textasteriskcentered\LWR@dash%
11810        \let\textbackslash\LWR@dash%
11811        \let\textbar\LWR@dash%
11812        \let\textbardbl\LWR@dash%
11813        \let\textbigcircle\LWR@dash%
11814        \let\textbraceleft\LWR@dash%
11815        \let\textbraceright\LWR@dash%
11816        \let\textbullet\LWR@dash%
11817        \let\textcopyright\LWR@dash%
11818        \let\textdagger\LWR@dash%
11819        \let\textdaggerdbl\LWR@dash%
11820        \let\textdollar\LWR@dash%
11821        \let\textellipsis\LWR@dash%
11822        \let\textemdash\LWR@dash%
11823        \let\textendash\LWR@dash%
11824        \let\textexclamdown\LWR@dash%
11825        \let\textgreater\LWR@dash%
11826        \let\textless\LWR@dash%
11827        \let\textordfeminine\LWR@dash%
11828        \let\textordmasculine\LWR@dash%
11829        \let\textparagraph\LWR@dash%
11830        \let\textperiodcentered\LWR@dash%
11831        \let\textpertenthousand\LWR@dash%
11832        \let\textperthousand\LWR@dash%
11833        \let\textquestiondown\LWR@dash%
11834        \let\textquotedblleft\LWR@dash%
11835        \let\textquotedblright\LWR@dash%
11836        \let\textquoteleft\LWR@dash%
11837        \let\textquoteright\LWR@dash%
11838        \let\textregistered\LWR@dash%
11839        \let\textsection\LWR@dash%
11840        \let\textsterling\LWR@dash%
11841        \let\texttrademark\LWR@dash%
11842        \let\textunderscore\LWR@dash%
11843        \let\textvisiblespace\LWR@dash%
11844        \let\copyright\LWR@dash%
11845        \let\dag\LWR@dash%
11846        \let\ddag\LWR@dash%
11847        \let\dots\LWR@dash%
11848        \let\P\LWR@dash%
11849        \let\pounds\LWR@dash%
11850        \let\S\LWR@dash%
11851 %
11852        \renewcommand*{\aa}{a}%
11853        \renewcommand*{\AA}{A}%
11854        \renewcommand*{\AE}{AE}%
11855        \renewcommand*{\ae}{ae}%
11856        \renewcommand*{\dh}{d}%
11857        \renewcommand*{\DH}{D}%
11858        \renewcommand*{\DJ}{D}%
11859        \renewcommand*{\dj}{d}%
11860        \renewcommand*{\IJ}{IJ}%
11861        \renewcommand*{\ij}{ij}%
11862        \renewcommand*{\L}{L}%
```

```
11863      \renewcommand*{\l}{l}%
11864      \renewcommand*{\NG}{NG}%
11865      \renewcommand*{\ng}{ng}%
11866      \renewcommand*{\O}{O}%
11867      \renewcommand*{\o}{o}%
11868      \renewcommand*{\oe}{oe}%
11869      \renewcommand*{\OE}{OE}%
11870      \renewcommand*{\ss}{ss}%
11871      \renewcommand*{\SS}{SS}%
11872      \renewcommand*{\th}{th}%
11873      \renewcommand*{\TH}{TH}%
11874 %
11875      \let\guillemotleft\@empty%
11876      \let\guilsinglleft\@empty%
11877      \let\quotedblbase\@empty%
11878      \let\textquotedbl\@empty%
11879      \let\guillemotright\@empty%
11880      \let\guilsinglright\@empty%
11881      \let\quotesinglbase\@empty%

11882      \renewcommand*{\HTMLunicode}[1]{}%
11883      \renewcommand*{\HTMLentity}[1]{}%

11884      \renewcommand{\textsuperscript}[1]{##1}%
11885      \renewcommand{\textsubscript}[1]{##1}%

11886      \renewcommand{\underline}[1]{##1}%

11887      \RenewDocumentCommand{\hspace}{s m}{}%

11888      \RenewDocumentCommand{\LWR@htmlspanclass}{o D(){} m +m}{{##4}}%
11889      \DeclareExpandableDocumentCommand{\InlineClass}{D{(}{)}{} o m +m}{{##4}}%
```

Nullify math macros.

```
11890      \def\(##1\){}%
11891      \def\[##1\]{}%
11892      \RenewDocumentCommand{\LWR@subsingledollar}{s m m m}{}%
```

Nullify logos:

```
11893      \renewcommand*{\TeX}{TeX}%
11894      \renewcommand*{\LaTeX}{LaTeX}%
11895      \renewcommand*{\LaTeXe}{LaTeX2e}%
11896      \renewcommand*{\LuaTeX}{LuaTeX}%
11897      \renewcommand*{\LuaLaTeX}{LuaLaTeX}%
11898      \renewcommand*{\XeTeX}{XeTeX}%
11899      \renewcommand*{\XeLaTeX}{XeLaTeX}%
11900      \renewcommand*{\ConTeXt}{ConTeXt}%
11901      \renewcommand*{\BibTeX}{BibTeX}%
11902      \renewcommand*{\MakeIndex}{MakeIndex}%
11903      \renewcommand*{\AmS}{AmS}%
11904      \renewcommand*{\MiKTeX}{MiKTeX}%
11905      \renewcommand*{\LyX}{LyX}%
```

Use the simpler form with \texorpdfstring:

```
11906        \def\texorpdfstring{\expandafter\@secondoftwo}%
11907 }
11908 \catcode`\$=3%
11909 \catcode`\_=8%
```

\FilenameNullify {⟨*redefinitions*⟩}

Adds more nullifying definitions for filename generation.

```
11910 \newcommand*{\FilenameNullify}[1]{%
11911      \appto{\LWR@nullfonts}{#1}%
11912 }
```

```
11913 \end{warpHTML}
```

85   Math

85.1   Limitations

See Math, section 8.7.

85.2   HTML alt tag names

Redefinable names for the HTML alt tags, for translation according to the reader's native language.

for HTML & PRINT: 11914 \begin{warpall}

\AltTextOpen The opening part of HTML alt tag for an image. The default is a left parenthesis.
Default: (

```
11915 \newcommand*{\AltTextOpen}{(}
```

\AltTextClose The closing part of HTML alt tag for an image. The default is a right parenthesis.
Default: (

```
11916 \newcommand*{\AltTextClose}{)}
```

\ImageAltText The HTML alt tag for an image.
Default: image

```
11917 \newcommand*{\ImageAltText}{image}
```

\MathImageAltText The HTML alt tag for an SVG math image.
Default: "math image"

```
11918 \newcommand*{\MathImageAltText}{math image}
```

\LWR@ThisAltText The HTML alt tag for the next image. Cleared after use, and also after each lateximage, \LWR@subsingledollar, and each use of MATHJAX.

```
11919 \newcommand*{\LWR@ThisAltText}{}
```

\ThisAltText {⟨text⟩}

Assigns the HTML alt tag for the next image generated by lwarp, such as a lateximage, picture, or svg math.

```
11920 \newcommand*{\ThisAltText}[1]{%
11921     \renewcommand{\LWR@ThisAltText}{#1}%
11922 }
```

\PackageDiagramAltText Appended to the lateximage HTML alt tag for the images generated by many
Default: "diagram" packages.

```
11923 \newcommand*{\PackageDiagramAltText}{diagram}
```

```
11924 \end{warpall}
```

85.3   Inline and display math

for HTML output:
```
11925 \begin{warpHTML}
```

LWR@externalfilecnt (*Ctr*) Counter for the external files which are generated and then referenced from the HTML:

```
11926 \newcounter{LWR@externalfilecnt}
```

LWR@indisplaymathimage (*bool*) True if processing display math for svg output. Inside a lateximage, display math is only set to print-mode output if LWR@indisplaymathimage is false. Used to avoid nullifying display math before it has been completed.

```
11927 \newbool{LWR@indisplaymathimage}
```

LWR@insidemathcomment (*bool*) True while inside an HTML comment which is displaying a math environment. Used to undo the comment for a moment while creating a \label, so that the label's HTML tags will be seen by HTML.

```
11928 \newbool{LWR@insidemathcomment}
11929 \boolfalse{LWR@insidemathcomment}
```

LWR@xfakebold (*bool*) True if xfakebold \setBold is in use.

```
11930 \newbool{LWR@xfakebold}
11931 \boolfalse{LWR@xfakebold}
```

\LWR@orig@setBold Redefined by lwarp-xfakebold.

```
11932 \newcommand*{\LWR@orig@setBold}{}
```

\LWR@orig@unsetBold   Redefined by **lwarp-xfakebold**.

> 11933 `\newcommand*{\LWR@orig@unsetBold}{}`

\LWR@applyxfakebold   Redefined by **lwarp-xfakebold**.

> 11934 `\newcommand*{\LWR@applyxfakebold}{}`

\LWR@setcurrentfont   Sets the actual LaTeX font to that which was selected for HTML output. Ex: In HTML mode, \bfseries sets \LWR@f@series to "bf". This sets the PDF output here for use inside a lateximage.

> 11935 `\newcommand*{\LWR@setcurrentfont}{%`
> 11936 `    \LWR@traceinfo{Using font family \LWR@f@family}%`
> 11937 `    \@nameuse{LWR@print@\LWR@f@family family}%`
> 11938 `    \LWR@traceinfo{Using font series \LWR@f@series}%`
> 11939 `    \@nameuse{LWR@print@\LWR@f@series series}%`
> 11940 `    \LWR@traceinfo{Using font shape \LWR@f@shape}%`
> 11941 `    \@nameuse{LWR@print@\LWR@f@shape shape}%`
> 11942 `    \LWR@traceinfo{Using font caps shape \LWR@f@shapecaps}%`
> 11943 `    \@nameuse{LWR@print@\LWR@f@shapecaps shape}%`
> 11944 `}`

\\$ Plain dollar signs appearing in the HTML output may be interpreted by MATHJAX to be math shifts. For a plain text dollar \\$, use an HTML entity to avoid it being interpreted by MATHJAX, unless are inside a lateximage, in which case it will not be seen by MATHJAX.

> 11945 `\let\LWR@origtextdollar\$`
> 11946
> 11947 `\renewcommand*{\$}{%`
> 11948 `    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%`
> 11949 `        {\LWR@origtextdollar}%`
> 11950 `        {\HTMLunicode{00024}}%`
> 11951 `}`

lwarp_baseline_marker.png   *(file)*
lwarp_baseline_marker.eps   *(file)*

A marker to be used to help *pdfcrop* identify the inline math baseline and width. If either **graphicx** or **graphics** is loaded, this marker is placed at the lower left and lower right corners of the inline math. *pdfcrop* is then able to identify the width of the image, and also the height of an image such as a horizontal dash which does not otherwise touch the baseline.

A marker with alpha or opacity of 0% is not registered by *pdfcrop*, so the marker is a small square block of 1% alpha, which seems to work while still being effectively invisible in the final SVG image.

If **graphicx** is loaded, this marker is sized as a tiny 1 sp square. If **graphics** is loaded, this marker is used at its default size of around .25 pt. If neither graphics package is loaded, the marker is replaced by a 10 sp horizontal space, and there is no assistance for determining baseline or width of the inline math image. The best results are obtained when using **graphicx**.

\LWR@addbaselinemarker   Places a small marker in an SVG inline image. If **graphics** or **graphicx** are loaded, the marker is a mostly transparent image. If neither is loaded, no marker is used.

```
11952 \AtBeginDocument{
11953
11954 \ifpdf
11955     \newcommand*{\LWR@baselinename}{lwarp_baseline_marker.png}
11956 \else
11957     \ifXeTeX
11958         \newcommand*{\LWR@baselinename}{lwarp_baseline_marker.png}
11959     \else
11960         \newcommand*{\LWR@baselinename}{lwarp_baseline_marker.eps}
11961     \fi
11962 \fi
11963
11964 \IfFileExists{\LWR@baselinename}%
11965 {
11966     \IfPackageLoadedTF{graphicx}{
11967         \newcommand*{\LWR@addbaselinemarker}{%
11968             \LWR@origincludegraphics{\LWR@baselinename}%
11969         }
11970     }{
11971         \IfPackageLoadedTF{graphics}{
11972             \newcommand*{\LWR@addbaselinemarker}{%
11973                 \LWR@origincludegraphics{\LWR@baselinename}%
11974             }
11975         }{
11976             \newcommand*{\LWR@addbaselinemarker}{%
11977                 \global\booltrue{LWR@warnbaselinemarker}%
11978             }
11979             \AtEndDocument{
11980                 \ifbool{LWR@warnbaselinemarker}{
11981                     \PackageNoteNoLine{lwarp}{%
11982                         Load graphicx or graphics for improved\MessageBreak
11983                         SVG math sizing and baselines%
11984                     }
11985                 }{}
11986             }
11987         }
11988     }
11989 }{% lwarp_baseline_marker.png or .eps is not present
11990     \newcommand*{\LWR@addbaselinemarker}{%
11991         \global\booltrue{LWR@warnbaselinemarker}%
11992     }
11993     \AtEndDocument{
11994         \ifbool{LWR@warnbaselinemarker}{
11995             \PackageWarningNoLine{lwarp}{%
11996                 File \LWR@baselinename\space is not installed\MessageBreak
11997                 alongside the lwarp-*.sty files, so\MessageBreak
11998                 SVG math sizing and baselines may not be accurate}
11999         }{}
12000     }
12001 }
12002
12003 }% AtBeginDocument
```

LWR@warnbaselinemarker (*bool*)   True if the math baseline marker was ever called for, but graphics or graphicx were not loaded.

```
12004 \newbool{LWR@warnbaselinemarker}
12005 \boolfalse{LWR@warnbaselinemarker}
```

LWR@unknownmathsize (*bool*)   If TikZ or other objects are used inside math mode, the resulting image may exceed the TEX box, resulting in an incorrect measurement of the size of the resulting image. If this is so, the HTML styles for image size and depth will be neutralized.

```
12006 \newbool{LWR@unknownmathsize}
```

\LWR@singledollarmeasure {⟨*math expression*⟩}

Measures the size of the image of the math expression.

(In some circumstances svg math is used even if MATHJAX is preferred.)

svg math: \LWR@origensuredmath is part of argument #4.

svg math \ensuremath: \LWR@origensuredmath is part of argument #4.

svg dynamic math: \LWR@origensuredmath is part of argument #4.

MATHJAX: Argument #4 is the contents of the math expression without \LWR@origensuredmath. This case is handled above.

MATHJAX \ensuremath: \LWR@origensuredmath is part of argument #4.

MATHJAX dynamic math: Argument #4 is the contents of the math expression without \LWR@origensuredmath, so \LWR@origensuredmath is added below.

\ifmmode: Included "just in case".

Factored from \LWR@subsingledollarsvg.

```
12007 \newcommand*{\LWR@singledollarmeasure}[1]{%
12008     \begingroup%
```

Temporarily disable formatting while measuring the image parameters:

```
12009     \LWR@restoreorigformatting%
12010     \RenewDocumentEnvironment{lateximage}{s o s t? o o d()}{}{}% inside group
12011     \LWR@print@normalsize%
```

Temporarily set font for the HTML PDF output:

```
12012     \LWR@setcurrentfont%
```

lateximagedepth must be nested to avoid generating paragraph tags. $\mathcal{A}_{\mathcal{M}}\mathcal{S}$ math modifies the \text macro such that \addtocounter does not always occur as expected. Lower-level code is used instead.

```
12013     \global\advance\c@LWR@lateximagedepth 1\relax%
```

Typeset the math in a box. While doing so, some macros or environments may set LWR@unknownmathsize, in which case this will be used to cancel the HTML styles being generated here.

```
12014     \boolfalse{LWR@unknownmathsize}%
12015     \ifmmode%
12016         \global\sbox{\LWR@singledollarbox}{#1}%
12017     \else%
```

```
12018        \ifbool{LWR@dynamicmath}{%
12019            \ifbool{mathjax}{%
12020                \global\sbox{\LWR@singledollarbox}%
12021                    {\LWR@origensuredmath{#1}}%
12022            }{%
12023                \global\sbox{\LWR@singledollarbox}{#1}%
12024            }%
12025        }{%
12026            \global\sbox{\LWR@singledollarbox}{#1}%
12027        }%
12028    \fi%
```

Add a small and almost transparent marker at the depth of the image.

A math minus sign has the same depth as a plus, even though it does not draw anything below the baseline. This means that *pdfcrop* would crop the image without depth. The marker below the baseline is seen by *pdfcrop* and preserves the depth.

```
12029    \global\sbox{\LWR@singledollarbox}{%
12030        \usebox{\LWR@singledollarbox}%
12031        \raisebox{-\dp\LWR@singledollarbox}{%
12032            \LWR@addbaselinemarker%
12033        }%
12034    }%
```

More low-level code to undo the counter change.

```
12035    \global\advance\c@LWR@lateximagedepth -1\relax% Due to AmS \text macro.
```

Measure the depth:

```
12036    \setlength{\LWR@singledollardepth}{%
12037        \LateximageFontScale\dp\LWR@singledollarbox%
12038    }%
```

Make the length a global change:

```
12039    \global\LWR@singledollardepth=\LWR@singledollardepth%
```

Likewise for width:

```
12040    \setlength{\LWR@singledollarwidth}{%
12041        \LateximageFontScale\wd\LWR@singledollarbox%
12042    }%
12043    \global\LWR@singledollarwidth=\LWR@singledollarwidth%
```

Likewise for total height:

```
12044    \setlength{\LWR@singledollarheight}{%
12045        \LateximageFontScale\ht\LWR@singledollarbox%
12046    }%
12047    \addtolength{\LWR@singledollarheight}{%
12048        \LateximageFontScale\dp\LWR@singledollarbox%
12049    }%
12050    \global\LWR@singledollarheight=\LWR@singledollarheight%

12051    \endgroup%
12052 }
```

\LWR@subsingledollarsvg * {⟨*2: alt text*⟩} {⟨*3: add'l hashing*⟩} {⟨*4: math expression*⟩}

For inline math. Uses svg math. The image is measured and ajusted to the baseline of the html output, and placed inside a lateximage.

(In some circumstances svg math is used even if MathJax is preferred.)

Factored from \LWR@subsingledollar.

```
12053 \newcommand*{\LWR@subsingledollarsvg}[4]{%
12054     \LWR@traceinfo{LWR@subsingledollartsvg}%
```

Measure the depth, width, and height of the math image:

```
12055     \LWR@singledollarmeasure{#4}%
```

Set a style for the the height or width. The em unit is used so that the math scales according to the user's selected font size.

Start with the greater of the width or the height, biased towards the width:

```
12056     \ifdimgreater{\LWR@singledollarwidth}{.7\LWR@singledollarheight}{%
12057         \def\LWR@singledollarstyle{%
12058             width:\LWR@convertto{em}{\the\LWR@singledollarwidth} em%
12059         }%
12060     }{%
12061         \def\LWR@singledollarstyle{%
12062             height:\LWR@convertto{em}{\the\LWR@singledollarheight} em%
12063         }%
12064     }%
```

If a very narrow width, use the height.

```
12065     \ifdimless{\LWR@singledollarwidth}{.2em}%
12066     {%
12067         \def\LWR@singledollarstyle{%
12068             height:\LWR@convertto{em}{\the\LWR@singledollarheight} em%
12069         }%
12070     }%
12071     {}%
```

If very wide and short, use the width:

```
12072     \ifdimless{\LWR@singledollarheight}{.2em}%
12073     {%
12074         \def\LWR@singledollarstyle{%
12075             width:\LWR@convertto{em}{\the\LWR@singledollarwidth} em%
12076         }%
12077     }%
12078     {}%
```

If there is significant text depth, add the depth to the style.

```
12079     \ifdimgreater{\LWR@singledollardepth}{0.05ex}{%
12080         \def\LWR@singledollardepthstyle{%
12081             \ ; % extra space
12082             \LWR@print@mbox{%
12083             vertical-align:-\LWR@convertto{em}{\the\LWR@singledollardepth} em%
12084             } % extra space
```

```
12085        }%
12086    }{%
12087        \def\LWR@singledollardepthstyle{}%
12088    }%
```

If using certain TikZ actions inside math, the resulting image may exceed the TEX boundaries, so the HTML size styles may be incorrect, and must be neutralized.

```
12089    \ifbool{LWR@unknownmathsize}{%
12090        \def\LWR@singledollarstyle{}%
12091        \def\LWR@singledollardepthstyle{}%
12092    }{}%
```

Create the lateximage using the alternate tag and the computed size and depth. The star causes lateximage to use an MD5 hash as the filename. When hashing, also include the current font and color in the hash.

```
12093    \ifbool{LWR@dynamicmath}{%
12094        \LWR@traceinfo{subsingledollarsvg: dynamic}%
12095        \begin{lateximage}% no hashing
12096            [\MathImageAltText]% alt tag
12097            ?% Do not detokenize the alt tag
12098            []% no add'l hashing
12099            [\LWR@singledollarstyle \LWR@singledollardepthstyle]% CSS
12100            (math)% ARIA
12101    }{% not dynamic math
12102        \LWR@traceinfo{subsingledollarsvg: static}%
12103        \IfValueTF{#1}{% #1 True
12104            \LWR@findcurrenttextcolor% sets \LWR@tempcolor
```

Support for xfakebold:

```
12105            \ifbool{LWR@xfakebold}%
12106                {\def\LWR@tempone{Y}}%
12107                {\def\LWR@tempone{N}}%

12108            \LWR@traceinfo{subsingledollarsvg about to lateximage}%
12109            \begin{lateximage}*% use hashing
12110                [#2]% alt
12111                *% do not add open/closing braces
12112                ?% Do not detokenize the alt tag
12113                [% addl' hashing
12114                    #3%
12115                    FM\LWR@f@family%
12116                    SR\LWR@f@series%
12117                    SH\LWR@f@shape%
12118                    SHC\LWR@f@shapecaps%
12119                    CL\LWR@tempcolor%
12120                    FB\LWR@tempone% xfakebold
12121                ]%
12122                [\LWR@singledollarstyle \LWR@singledollardepthstyle]% CSS
12123                (math)% ARIA
12124                \LWR@traceinfo{subsingledollar did lateximage}%
12125        }{% #1 False
12126            \begin{lateximage}% no hashing
12127                [#2]% alt
12128                ?% Do not detokenize the alt tag
12129                []% no add'l hashing
12130                [\LWR@singledollarstyle \LWR@singledollardepthstyle]% CSS
```

```
12131              (math)% ARIA
12132         }%
12133    }% not dynamic math
```

Place small and almost transparent markers on the baseline at the left and right edges of the image. These markers are seen by *pdfcrop*, and force vertically-centered objects such as a dash to be raised off the baseline in the cropped image, and also force the total width and left / right margins to be correct. (Except that in some fonts a character may exceed the bounding box, and thus may appear wider than expected when converted to an image.)

```
12134    \LWR@addbaselinemarker%
```

Support for xfakebold:

```
12135    \LWR@applyxfakebold%
```

Typeset the contents:

```
12136    \usebox{\LWR@singledollarbox}%
```

The closing baseline marker:

```
12137    \LWR@addbaselinemarker%
```

```
12138    \end{lateximage}%
12139 %
12140 }
```

\LWR@subsingledollar * {⟨*2: alt text*⟩} {⟨*3: add'l hashing*⟩} {⟨*4: math expression*⟩}

For inline math. Uses MATHJAX, or for svg math the image is measured and ajusted to the baseline of the HTML output, and placed inside a lateximage.

svg math: \LWR@origensuredmath is part of argument #4.

svg math \ensuremath: \LWR@origensuredmath is part of argument #4.

svg dynamic math: \LWR@origensuredmath is part of argument #4.

MATHJAX: Argument #4 is the contents of the math expression without \LWR@origensuredmath. This case is handled above.

MATHJAX \ensuremath: \LWR@origensuredmath is part of argument #4.

MATHJAX dynamic math: Argument #4 is the contents of the math expression without \LWR@origensuredmath, so \LWR@origensuredmath is added below.

image filename hashing   If starred, a hashed filename is used.If so, the hash is based on the alt tag and also the additional hashing argument.

This may be used to provide an expression with a simple alt tag but also enough additional information to provide a unique hash.

An example is when the expression is a complicated TEX expression, which would not copy/paste well. A simplified tag may be used, while the complicated expression is used in the additional hashing argument to ensure a unique image.

Another example is when the expression is simple, but the image depends on options. These options may be decoded into text form and included in the additional hashing argument in order to make the hash unique according to the set of options, even if the simple alt tag is still the same.

```
12141 \newlength{\LWR@singledollarwidth}
12142 \newlength{\LWR@singledollarheight}
12143 \newlength{\LWR@singledollardepth}
12144
12145 \newsavebox{\LWR@singledollarbox}
12146
12147 \NewDocumentCommand{\LWR@subsingledollar}{s m m m}{%
12148     \LWR@traceinfo{LWR@subsingledollar !#2!}%
12149     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
12150     {%
12151         \LWR@traceinfo{LWR@subsingledollar: already in a lateximage}%
12152             #4% contents
12153     }%
12154     {% not in a lateximage
12155         \begingroup%
```

Support for xfakebold:

```
12156         \LWR@applyxfakebold%
```

MATHJAX cannot parse the often complicated TEX expressions which appear in the various uses of \ensuredmath. \ensuremath forces the alt tag to "(math image)", as translated according to \MathImageAltText. If this is the case, force the use of a lateximage even if MATHJAX. Likewise for siunitx if parse-numbers=false.

If MATHJAX, or if formatting math for a word processor, and not \ensuredmath, and not a dynamic math expression, print the math expression:

```
12157         \ifboolexpr{%
12158             (
12159                 bool{mathjax} or
12160                 ( bool{FormatWP} and bool{WPMarkMath} )
12161             ) and
12162             ( not test {
12163                     \ifstrequal {#2}% from \ensuredmath
12164                         {\AltTextOpen\MathImageAltText\AltTextClose}
12165                 }
12166             ) and
12167             ( not bool{LWR@dynamicmath} )
12168         }%
```

For MATHJAX, print the math between \(and \):

```
12169         {% mathjax
12170             \LWR@traceinfo{LWR@subsingledollar: Mathjax}%
12171             {%
12172                 \boolfalse{LWR@HTMLsanitize@tmpb@removebackslash}%
12173                 \textbackslash(%
12174                     {%
```

\ifmmode to avoid error about \ttfamily inside math mode in the case of nested math, ex. equation with tcolorbox with math.

```
12175                    \ifmmode\else\LWR@print@ttfamily\fi%
12176                    \LWR@HTMLsanitizedetokenized{\detokenize{#4}}%
12177                }%
12178                \textbackslash)%
12179            }%
12180        }% mathjax
```

For svg, print the math inside a lateximage, with an <alt> tag of the LaTeX code, and a css style to control the baseline adjustment.

```
12181        {% not mathjax
12182            \LWR@traceinfo{%
12183            LWR@subsingledollar: NOT mathjax, or is ensuremath, or is dynamic%
12184            }%
12185            \LWR@subsingledollarsvg{#1}{#2}{#3}{#4}%
12186        }% not mathjax
12187        \endgroup%
12188    }% not in a lateximage
```

Clear the single-use alt text:

```
12189    \gdef\LWR@ThisAltText{}%
12190    \LWR@traceinfo{LWR@subsingledollar: done}%
12191 }
```

```
12192 \LetLtxMacro\LWR@origdollar$
12193 \LetLtxMacro\LWR@secondorigdollar$% balance for editor syntax highlighting
```

```
12194 \LetLtxMacro\LWR@origopenparen\(
12195 \LetLtxMacro\LWR@origcloseparen\)
12196 \LetLtxMacro\LWR@origopenbracket\[
12197 \LetLtxMacro\LWR@origclosebracket\]
```

$ Redefine the dollar sign to place math inside a lateximage, or use MATHJAX:

$$
```
12198 \begingroup
12199 \catcode`\$=\active%
12200 \protected\gdef${\@ifnextchar$\LWR@doubledollar\LWR@singledollar}%
```

Used by chemformula to escape single-dollar math:

```
12201 \protected\gdef\LWR@newsingledollar{\@ifnextchar$\LWR@doubledollar\LWR@singledollar}%
```

\LWR@doubledollar Redefine the double dollar sign to place math inside a lateximage, or use MATH-JAX:

```
12202 \protected\gdef\LWR@doubledollar$#1$${%
```

If MATHJAX or formatting for a word processor, print the LaTeX expression:

```
12203    \ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
```

For MATHJAX, print the math between \[and \]. If there is a footnote, endnote, or other kind of note ('note' is present), sync the note numbers.

```
12204        {% mathjax: intentional blank line:
12205
12206            \begingroup%
12207            \IfSubStr{\detokenize\expandafter{#1}}{\detokenize{note}}{%
```

The equation is printed to the PDF output inside HTML comment tags. This allows labels and footnotes to be accepted and processed. The math environment is selected here, and \LWR@hidelatexequation will use the original print-mode meaning of math.

```
12208                \LWR@hidelatexequation{math}{#1}%


12209                \InlineClass{hidden}{\LWR@syncnotenumbers}%
12210                \boolfalse{LWR@HTMLsanitize@tmpb@removebackslash}%
12211                \textbackslash[%
12212                {%
12213                    \LWR@print@ttfamily%
12214                    \LWR@HTMLsanitizedetokenized{\detokenize{#1}}%
12215                }%
12216                \textbackslash]
12217                \InlineClass{hidden}{\LWR@syncnotenames}%
12218            }% yes note
12219            {% no note
12220                \boolfalse{LWR@HTMLsanitize@tmpb@removebackslash}%
12221                \textbackslash[%
12222                {%
12223                    \LWR@print@ttfamily%
12224                    \LWR@HTMLsanitizedetokenized{\detokenize{#1}}%
12225                }%
12226                \textbackslash]
12227            }% no note
12228            \endgroup%
12229
12230        }% mathjax
```

For SVG, print the math inside a lateximage, with an <alt> tag of the LATEX code:

```
12231        {% not mathjax
12232            \begin{BlockClass}{displaymath}%
12233            \LWR@newautoidanchor%
12234            \booltrue{LWR@indisplaymathimage}%
12235            \begin{lateximage}%
12236            [% alt text
12237                \textbackslash{[} % extra space
12238                \LWR@HTMLsanitizedetokenized{\detokenize{#1}} % extra space
12239                \textbackslash{]}%
12240            ]%
12241            *% do not add open/closing braces
12242            ?% Do not detokenize the alt tag
12243            (math)% ARIA
```

Support for xfakebold:

```
12244            \LWR@applyxfakebold%


12245            \LWR@origdollar\LWR@origdollar#1\LWR@origdollar\LWR@origdollar%
12246            \end{lateximage}%
```

```
12247        \end{BlockClass}%
12248    }% not mathjax
```

Clear the single-use alt text:

```
12249        \gdef\LWR@ThisAltText{}%
12250 }%
```

\LWR@singledollar {⟨*math expression*⟩}

```
12251 \protected\gdef\LWR@singledollar#1${%
12252    \LWR@traceinfo{LWR@singledollar}%
12253    \ifbool{mathjax}{%
12254        \begingroup%
12255        \boolfalse{LWR@HTMLsanitize@tmpb@removebackslash}%
12256        \LWR@subsingledollar*%
12257        {% alt tag
12258            \textbackslash( %
12259            \LWR@HTMLsanitizedetokenized{\detokenize{#1}} % extra space
12260            \textbackslash)%
12261        }%
12262        {singledollar}% add'l hashing
12263        {#1}% contents
12264        \endgroup%
12265    }{% not mathjax
12266        \LWR@subsingledollar*%
12267        {% alt tag
12268            \textbackslash( %
12269            \LWR@HTMLsanitizedetokenized{\detokenize{#1}} % extra space
12270            \textbackslash)%
12271        }%
12272        {singledollar}% add'l hashing
12273        {\LWR@origensuredmath{#1}}% contents
12274    }% not mathjax
```

Clear the single-use alt text:

```
12275        \gdef\LWR@ThisAltText{}%
12276 }
```

\(Redefine to the above dollar macros.
\[

```
12277 \AtBeginDocument{
12278    \protected\gdef\(#1\){$#1$}
12279    \protected\gdef\[#1\]{$$#1$$}
12280 }
12281
12282 \endgroup% active $
```

```
12283 \AtBeginDocument{
12284 \LetLtxMacro\LWR@openbracketnormal\[
12285 \LetLtxMacro\LWR@closebracketnormal\]
12286 }
```

\@ensuredmath {⟨*expression*⟩}

If MATHJAX, a lateximage is used, since \ensuremath is often used for complex TEX expressions which MATHJAX may not render. If SVG math, a hashed file is used with a simple alt tag, but additional hashing provided by the contents.

```
12287 \LetLtxMacro\LWR@origensuredmath\@ensuredmath
12288
12289 \renewcommand{\@ensuredmath}[1]{%
12290     \ifbool{mathjax}{%
12291         \begingroup%
12292         \boolfalse{LWR@HTMLsanitize@tmpb@removebackslash}%
12293         \LWR@subsingledollar*{\AltTextOpen\MathImageAltText\AltTextClose}%
12294         {%
12295             \protect\LWR@HTMLsanitizedetokenized{\detokenize\expandafter{#1}}%
12296         }%
12297         {%
12298             \relax%
12299             \LWR@origensuredmath{#1}%
12300         }%
12301         \endgroup%
12302     }{% SVG math
```

If already inside a lateximage in math mode, continue as-is.

```
12303         \ifmmode%
12304             \LWR@origensuredmath{#1}%
12305         \else%
```

Create an inline math lateximage with a simple alt tag and additional hashing according to the contents.

```
12306             \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
12307                 {\LWR@origensuredmath{#1}}%
12308                 {%
12309                     \LWR@subsingledollar*%
12310                         {\AltTextOpen\MathImageAltText\AltTextClose}%
12311                         {%
12312                             \protect\LWR@HTMLsanitizedetokenized{%
12313                                 \detokenize\expandafter{#1}%
12314                             }%
12315                         }%
12316                         {\LWR@origensuredmath{#1}}%
12317                 }%
12318         \fi%
12319     }%
```

Clear the single-use alt text:

```
12320     \gdef\LWR@ThisAltText{}%
12321 }
```

Remember then remove the old math and displaymath environments:

```
12322 \let\LWR@orig@math\math
12323 \let\LWR@orig@endmath\endmath
12324
12325 \let\LWR@orig@displaymath\displaymath
12326 \let\LWR@orig@enddisplaymath\enddisplaymath
12327
```

```
12328 \let\math\relax
12329 \let\endmath\relax
12330
12331 \let\displaymath\relax
12332 \let\enddisplaymath\relax
```

math (*env.*) Set math mode then typeset the body of what was between the begin/end. See the environ package for \BODY.

```
12333 \NewEnviron{math}{\expandafter\(\BODY\)}
```

LWR@displaymathnormal (*env.*) Set math mode then typeset the body of what was between the begin/end. See the environ package for \BODY.

```
12334 \NewEnviron{LWR@displaymathnormal}{\expandafter\[\BODY\]\@ignoretrue}
```

Set the default displaymath to the normal version:

```
12335 \LetLtxMacro\displaymath\LWR@displaymathnormal%
12336 \LetLtxMacro\enddisplaymath\endLWR@displaymathnormal%
```

LWR@displaymathother (*env.*) A version of displaymath which can handle complicated objects, but does not supply MathJax or html alt tags.

```
12337 \newenvironment{LWR@displaymathother}
12338 {%
12339     \begin{BlockClass}{displaymath}%
12340     \LWR@newautoidanchor%
12341     \booltrue{LWR@indisplaymathimage}%
12342     \begin{lateximage}[\MathImageAltText]?(math)% [alt](ARIA)
12343     \LWR@origdollar\LWR@origdollar%
12344 }
12345 {%
12346     \LWR@origdollar\LWR@origdollar%
12347     \end{lateximage}%
12348     \end{BlockClass}%
12349 }
```

LWR@equationother (*env.*) A version of displaymath which can handle complicated objects, but does not supply MathJax or html alt tags.

```
12350 \newenvironment{LWR@equationother}
12351 {%
12352     \begin{BlockClass}{displaymathnumbered}%
12353     \LWR@newautoidanchor%
12354     \booltrue{LWR@indisplaymathimage}%
12355     \begin{lateximage}[\MathImageAltText]?(math)% [alt](ARIA)
12356     \LWR@orig@equation%
12357 }
12358 {%
12359     \LWR@orig@endequation%
12360     \end{lateximage}%
12361     \end{BlockClass}%
12362 }
```

85.4 MathJax support

LWR@nextequation (*Ctr*)   Used to add one to compute the next equation number.

```
12363 \newcounter{LWR@nextequation}
```

Determing how to set MathJax section and equation numbers. Adjusts for various kinds of \theequation to determine \theMathJaxsection and \theMathJaxequation.

```
12364 \newcommand\LWR@article@theequation{\@arabic\c@equation}
12365
12366 \newcommand\LWR@book@theequation
12367   {\ifnum \c@chapter>\z@ \thechapter.\fi \@arabic\c@equation}
12368
12369
12370 \newcommand\LWR@chapter@theequation{\thechapter.\arabic{equation}}
12371 \newcommand\LWR@section@thequation{\thesection.\arabic{equation}}
12372 \newcommand\LWR@subsection@thequation{\thesubsection.\arabic{equation}}
12373
12374 \AtBeginDocument{
12375     % default per article class:
12376     \newcommand*{\theMathJaxsubequations}{0}
12377     \newcommand*{\theMathJaxsection}{}
12378     \newcommand*{\theMathJaxequation}{\arabic{equation}}
12379
12380     \ifdefstrequal{\theequation}{\LWR@article@theequation}
12381     {}{
12382     \ifdefstrequal{\theequation}{\LWR@book@theequation}{
12383      \renewcommand*{\theMathJaxsection}{\ifnum \c@chapter>\z@ \thechapter.\fi}
12384     }{
12385     \ifdefstrequal{\theequation}{\LWR@subsection@thequation}{
12386         \renewcommand*{\theMathJaxsection}{\thesubsection{}.}
12387     }{
12388     \ifdefstrequal{\theequation}{\LWR@section@thequation}{
12389         \renewcommand*{\theMathJaxsection}{\thesection{}.}
12390     }{
12391     \ifdefstrequal{\theequation}{\LWR@chapter@theequation}{
12392         \renewcommand*{\theMathJaxsection}{\thechapter{}.}
12393     }{% unknown format
12394         \PackageWarningNoLine{lwarp}
12395         {%
12396           Unknown equation tag format for \protect\theequation.\MessageBreak
12397             Article-style equation numbering will be used%
12398         }
12399     }}}}}
12400 }
```

\LWR@syncmathjax   Sets the MathJax equation format and number for the following equations.

These MathJax commands are printed inside "\(" and "\)" characters. They are printed to HTML output, not interpreted by LaTeX.

```
12401 \newcommand*{\LWR@syncmathjax}{%
```

Tell MathJax that the next equation number is the current LaTeX equation number.

Before each equation, lwarp inserts into the HTML code:

    \seteqnumber{subequations?}{section}{number}

subequations? is 0 usually, 1 if inside amsmath subequations.

section is a string printed as-is, or empty.

number is auto-incremented by MATHJAX between equations.

Place the MATHJAX command inside "\(" and "\)" characters, to be printed to HTML, not interpreted by LATEX.

```
12402        \LWR@stoppars%
12403        \InlineClass{hidden}{
12404            \textbackslash(%
12405            \textbackslash{}seteqnumber%
12406            \{\theMathJaxsubequations\}%
12407            \{\theMathJaxsection\}%
12408            \{\theMathJaxequation\}%
12409            \textbackslash)%
12410        }
12411        \LWR@startpars%
12412 }
```

\LWR@hidelatexequation {⟨*environment*⟩} {⟨*contents*⟩}

Creates the LATEX version of the equation inside an HTML comment.

```
12413 \NewDocumentCommand{\LWR@hidelatexequation}{m +m}{%
```

Stop HTML paragraph handling and open an HTML comment:

```
12414        \LWR@stoppars
12415        \LWR@htmlopencomment
12416
```

Start the LATEX math environment inside the HTML comment:

```
12417        \begingroup
12418        \@nameuse{LWR@orig@#1}
```

While in the math environment, restore various commands to their LATEX meanings.

```
12419        \LWR@restoreorigformatting
12420        \booltrue{LWR@insidemathcomment}
```

Temporarily prevent underfull \hbox warnings.

```
12421        \hbadness=10000\relax%
```

See \LWR@htmlmathlabel in section 85.7.1.

Print the contents of the equation:

```
12422        #2
```

End the LATEX math environment inside the HTML comment:

```
12423        \@nameuse{LWR@orig@end#1}
12424        \endgroup
12425
```

Close the HTML comment and resume HTML paragraph handling:

```
12426      \LWR@htmlclosecomment
12427      \boolfalse{LWR@insidemathcomment}
12428      \LWR@startpars
12429 }
```

\LWR@addmathjax {⟨*environment name*⟩} {⟨*contents*⟩}

Given the name of a math environment and its contents, create a MATHJAX instance. The contents are printed to HTML output, not interpreted by LATEX.

```
12430 \NewDocumentCommand{\LWR@addmathjax}{m +m}{%
```

```
12431      \LWR@orignobreakspace\LWR@orignewline
```

Enclose the MATHJAX environment inside printed "\(" and "\)" characters. Print the environment name and contents, sanitizing for HTML special characters.

```
12432      {%
12433          \LWR@print@ttfamily%
12434          \textbackslash{}begin\{#1\}
```

The `alignat` environment takes a mandatory argument, which must be replicated here.

```
12435          \ifboolexpr{
12436              test {\ifstrequal{#1}{alignat}} or
12437              test {\ifstrequal{#1}{alignat*}} or
12438              test {\ifstrequal{#1}{alignat+}}
12439          }%
12440          {\{\arabic{LWR@maxfields@}\}}%
12441          {}%
```

The environment contents and \end:

```
12442          \LWR@orignewline%
12443          \boolfalse{LWR@HTMLsanitize@tmpb@removebackslash}%
12444          \LWR@HTMLsanitizeexpanded{\detokenize\expandafter{#2}}%
12445          \LWR@orignewline%
12446          \textbackslash{}end\{#1\}
12447      }%
```

```
12448      \LWR@orignewline
12449 }
```

85.5   Equation environment

Remember existing `equation` environment, after redefined by amsmath, if loaded.

```
12450 \AtBeginDocument{
12451      \let\LWR@orig@equation\equation
12452      \let\LWR@orig@endequation\endequation
12453      \csletcs{LWR@orig@equation*}{equation*}
12454      \csletcs{LWR@orig@endequation*}{endequation*}
12455 }
```

\LWR@doequation {⟨*env contents*⟩} {⟨*env name*⟩}

> For svg math output, the contents are typeset using the original equation inside a lateximage, along with an <alt> tag containing a detokenized copy of the LaTeX source for the math.

> For MathJax output, the contents are typeset in an original equation environment placed inside a html comment, with special processing for \labels. The contents are also printed to the html output for processing by the MathJax script.

```
12456 \newcommand*{\LWR@doequation}[2]{%
12457
```

> If mathjax or FormatWP, print the LaTeX expression:

```
12458     \ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
```

> MathJax output:

```
12459     {
```

> Print commands to syncronize MathJax's equation number and format to the current LaTeX chapter/section and equation number:

```
12460         \LWR@syncmathjax%
```

> Print the LaTeX math inside an html comment:

```
12461         \LWR@hidelatexequation{#2}{#1}
12462     }
```

> svg output: Create the lateximage along with an html <alt> tag having an equation number, the LaTeX equation environment commands, and the contents of the environment's \BODY.

```
12463     {% not mathjax
```

> Begin the lateximage with an <alt> tag containing the math source:

```
12464         \ifstrequal{#2}{equation*}{%
12465             \begin{BlockClass}{displaymath}%
12466         }{%
12467             \begin{BlockClass}{displaymathnumbered}%
12468         }%
12469         \LWR@newautoidanchor%
12470         \booltrue{LWR@indisplaymathimage}%
12471         \begin{lateximage}[%
12472             \ifstrequal{#2}{equation*}{%
12473                 \ifdefequal{\LWR@equationtag}{\theequation}{%
12474 %                                no tag was given
12475             }{%
12476                 (\LWR@equationtag) % tag was given
12477             }%
12478         }{%
12479             (\LWR@equationtag) % automatic numbering
12480         }%
12481         \textbackslash{begin\{#2\}} % extra space
12482     \LWR@HTMLsanitizeexpanded{\detokenize\expandafter{#1}} % extra space
```

```
12483          \textbackslash{end\{#2\}}%
12484        ]*?(math)% alt tag, ARIA
```

Support for xfakebold:

```
12485          \LWR@applyxfakebold%
```

Create the actual LaTeX-formatted equation inside the lateximage using the contents of the environment.

```
12486          \@nameuse{LWR@orig@#2}%
12487          #1% contents collected by \collect@body
12488          \@nameuse{LWR@orig@end#2}%
12489          \end{lateximage}%
12490          \end{BlockClass}%
12491      }% not mathjax
```

Clear the single-use alt text:

```
12492      \gdef\LWR@ThisAltText{}%
12493 }
```

After the environment, if MathJax, print the math to the html output for MathJax processing. If a footnote is used, sync the footnote counter before, then unsync after for non-equation environments, as defined next.

```
12494 \newcommand*{\LWR@doendequation}[1]{%
12495      \ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
12496      {%
12497          \IfSubStr{\detokenize\expandafter{\BODY}}{\detokenize{note}}{%
12498              \InlineClass{hidden}{\LWR@syncnotenumbers}%
12499              \LWR@addmathjax{#1}{\BODY}%
12500              \InlineClass{hidden}{\LWR@syncnotenames}%
12501          }{%
12502              \LWR@addmathjax{#1}{\BODY}%
12503          }%
12504      }{}%
12505
```

Clear the single-use alt text:

```
12506      \gdef\LWR@ThisAltText{}%
12507 }
```

The following are used to syncronize footnote marks and related to MathJax if *note* is used inside the MathJax expression. The counter is read from LaTeX then defined into MathJax for use during the following equation. After the equation, the MathJax value is returned to the text from \footnotename. Other notes may be added by appending to \LWR@syncnotenumbers and \LWR@syncnotenames.

\LWR@synconenotenumber {⟨*MathJax variable*⟩} {⟨*mark*⟩}

```
12508 \newcommand*{\LWR@synconenotenumber}[2]{%
12509      \textbackslash(
12510      \textbackslash{}def\textbackslash{}#1\{#2\}
12511      \textbackslash)
12512 }
```

\LWR@syncnotenumbers Assignments to make.

```
12513 \newcommand*{\LWR@syncnotenumbers}{\LWR@synconenotenumber{LWRfootnote}{\thefootnote}}
```

\LWR@synconenotename {⟨*MathJax variable*⟩} {⟨*text*⟩}

```
12514 \newcommand*{\LWR@synconenotename}[2]{%
12515     \textbackslash(
12516     \textbackslash{}def\textbackslash{}#1name\{#2\}
12517     \textbackslash)
12518 }
```

\LWR@syncnotenames Assignments to make.

```
12519 \newcommand*{\LWR@syncnotenames}{\LWR@synconenotename{LWRfootnote}{\footnotename}}
```

Remove existing equation environment:

```
12520 \AtBeginDocument{
12521     \let\equation\relax
12522     \let\endequation\relax
12523     \csletcs{equation*}{relax}
12524     \csletcs{endequation*}{relax}
12525 }
```

equation (*env.*) The new equation environment is created with \NewEnviron (from the environ package), which stores the contents of its environment in a macro called \BODY.

```
12526 \AtBeginDocument{
12527     \NewEnviron{equation}%
12528         {\LWR@doequation{\BODY}{equation}}%
12529         [\LWR@doendequation{equation}]
12530
12531     \LetLtxMacro\LWR@equationnormal\equation
12532     \LetLtxMacro\endLWR@equationnormal\endequation
12533 }% AtBeginDocument
```

equation* (*env.*)

```
12534 \AtBeginDocument{
12535     \NewEnviron{equation*}%
12536         {\LWR@doequation{\BODY}{equation*}}%
12537         [\LWR@doendequation{equation*}]
12538
12539     \csletcs{LWR@equationnormalstar}{equation*}
12540     \csletcs{LWR@endequationnormalstar}{endequation*}
12541 }% AtBeginDocument
```

Remember the "less" version of equation, which uses MATHJAX and alt tags, but does not support complicated contents such as some TikZ expressions.

```
12542 \AtBeginDocument{
12543     \LetLtxMacro\LWR@equationless\equation
12544     \LetLtxMacro\endLWR@equationless\endequation
12545     \csletcs{LWR@equationlessstar}{equation*}
12546     \csletcs{LWR@endequationlessstar}{endequation*}
12547 }
```

85.6 \displaymathnormal and \displaymathother

\displaymathnormal By default, or when selecting \displaymathnormal, MATHJAX math display environments print their contents as text into HTML for MATHJAX to interpret, and
SVG display math environments render their contents as SVG images and use their
contents as the alt tag of HTML output. To do so, the contents are loaded into a
macro for reuse. In some cases, such as complicated TikZ pictures, compilation
will fail.

\displaymathother When selecting \displaymathother, it is assumed that the contents are more com
MATHJAX unsupported plicated than "pure" math. An example is an elaborate TikZ picture, which will
complicated alt tag not render in MATHJAX and will not make sense as an HTML alt tag. In this mode,
MATHJAX is turned off, math display environments become SVG images, even if
MATHJAX is selected, and the HTML alt tags become simple messages. The contents are internally processed as an environment instead of a macro argument, so
complicated objects such as TikZ pictures are more likely to compile successfully.

\displaymathnormal Use when display math environments have simple math which is to sent to MATHsimple math objects JAX or included in HTML alt tags.

```
12548 \newcommand*{\displaymathnormal}{%
12549     \ifbool{LWR@origmathjax}{\booltrue{mathjax}}{\boolfalse{mathjax}}%
12550     \LetLtxMacro\[\LWR@openbracketnormal%
12551     \LetLtxMacro\]\LWR@closebracketnormal%
12552     \LetLtxMacro\displaymath\LWR@displaymathnormal%
12553     \LetLtxMacro\enddisplaymath\endLWR@displaymathnormal%
12554     \LetLtxMacro\equation\LWR@equationnormal%
12555     \LetLtxMacro\endequation\endLWR@equationnormal%
12556     \csletcs{equation*}{LWR@equationnormalstar}%
12557     \csletcs{endequation*}{LWR@endequationnormalstar}%
12558 }
```

\displaymathother Use when display math environments have complicated objects which will not
complicated math objects work with MATHJAX or should not be included in HTML alt tags. Complicated
contents are more likely to compile correctly.

```
12559 \newcommand*{\displaymathother}{%
12560     \boolfalse{mathjax}%
12561     \LetLtxMacro\displaymath\LWR@displaymathother%
12562     \LetLtxMacro\enddisplaymath\endLWR@displaymathother%
12563     \LetLtxMacro\[\LWR@displaymathother%
12564     \LetLtxMacro\]\endLWR@displaymathother%
12565     \LetLtxMacro\equation\LWR@equationother%
12566     \LetLtxMacro\endequation\endLWR@equationother%
12567     \csletcs{equation*}{displaymath}%
12568     \csletcs{endequation*}{enddisplaymath}%
12569 }
```

```
12570 \end{warpHTML}
```

```
12571 \begin{warpprint}
```

Print-mode versions:

```
12572 \newcommand*{\displaymathnormal}{}
12573 \newcommand*{\displaymathother}{}
```

```
12574 \newcommand*{\theMathJaxsubequations}{0}
12575 \newcommand*{\theMathJaxsection}{}
12576 \newcommand*{\theMathJaxequation}{\arabic{equation}}
```

```
12577 \end{warpprint}
```

`12578 \begin{warpHTML}`

85.7 AMS Math environments

85.7.1 Support macros

LWR@amsmultline (*bool*)   True if processing a multline environment.

To compensate for multline-speffic code, LWR@amsmultline is used to add extra horizontal space in \LWR@htmlmathlabel if is used in an amsmath environment which is not a multline environment and not an equation.

```
12579 \newbool{LWR@amsmultline}
12580 \boolfalse{LWR@amsmultline}
```

\LWR@beginhideamsmath   Starts hiding LATEX math inside an HTML comment.

```
12581 \newcommand*{\LWR@beginhideamsmath}{
12582     \LWR@stoppars
12583     \LWR@orignobreakspace\LWR@orignewline
12584     \LWR@htmlopencomment
12585
12586     \begingroup
12587     \LWR@restoreorigformatting
```

Temporarily prevent underfull \hbox warnings.

```
12588     \hbadness=10000\relax%
```

```
12589     \booltrue{LWR@insidemathcomment}
12590 }
```

\LWR@endhideamsmath   Ends hiding LATEX math inside an HTML comment.

```
12591 \newcommand*{\LWR@endhideamsmath}{
12592     \endgroup
12593
12594     \LWR@htmlclosecomment
12595     \boolfalse{LWR@insidemathcomment}
12596     \LWR@orignewline
12597     \LWR@startpars
12598 }
```

85.7.2 Environment patches

The amsmath environments already collect their contents in \@envbody for further processing. eqnarray is not an $\mathcal{A}_{\mathcal{M}}\mathcal{S}$ package, and thus requires special handling.

For svg math: Each envrionment is encapsulated inside a lateximage environ-
ment, along with a special optional argument of \LWR@amsmathbody or \LWR@amsmathbodynumbered
telling lateximage to use as the html <alt> tag the environment's contents which
were automatically captured by the $\mathcal{A}_{\mathcal{M}}\mathcal{S}$ environment.

For MathJax: Each environment is synched with LaTeX's equation numbers, typeset
with LaTeX inside an html comment, then printed to html output for MathJax to
process.

eqnarray (*env.*) This environmnet is not an $\mathcal{A}_{\mathcal{M}}\mathcal{S}$ environment and thus its body is not automat-
ically captured, so the environ package is used to capture the environment into
\BODY.

```
12599 \let\LWR@origeqnarray\eqnarray
12600 \let\LWR@origendeqnarray\endeqnarray
```

To remember whether the starred environment was used, and thus whether to
number the equations:

```
12601 \newbool{LWR@numbereqnarray}
12602 \booltrue{LWR@numbereqnarray}
```

Common code used by eqnarray and Beqnarray (from fancybox):

```
12603 \newcommand{\LWR@eqnarrayfactor}{%
```

If mathjax or FormatWP, print the LaTeX expression:

```
12604    \ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
12605    {%
```

If MathJax, the environment contents (the \BODY) are executed in a html com-
ment to trigger the correct equation number increment (if not starred), then are
included verbatim in the output for MathJax to interpret:

```
12606       \LWR@syncmathjax%
12607       \boolfalse{LWR@amsmultline}%
12608       \boolfalse{LWR@HTMLsanitize@tmpb@removebackslash}%
12609       \ifbool{LWR@numbereqnarray}%
12610       {%
```

If numbering the equations, execute a copy inside an html comment block:

```
12611          \LWR@beginhideamsmath%
12612          \LWR@origeqnarray%
12613          \BODY%
12614          \LWR@origendeqnarray%
12615          \LWR@endhideamsmath%
```

Then print the (sanitized) contents to the output for MathJax to interpret:

```
12616          \LWR@addmathjax{eqnarray}{\BODY}%
12617       }%
12618       {% not LWR@numbereqnarray
```

If not numbering equations, just create the contents for MathJax:

```
12619        \LWR@addmathjax{eqnarray*}{\BODY}%
12620      }% LWR@numbereqnarray
12621    }% mathjax
12622    {% not mathjax
12623      \ifbool{LWR@numbereqnarray}%
12624      {%
```

For numbered svg equations, first create a lateximage with an alt attribute containing sanitized copy of the source code:

```
12625        \begin{BlockClass}{displaymathnumbered}%
12626        \LWR@newautoidanchor%
12627        \booltrue{LWR@indisplaymathimage}%
12628        \begin{lateximage}[%
12629            (\LWR@startingequationtag\textendash\LWR@equationtag)%
12630            \LWR@addmathjax{eqnarray}{\BODY}%
12631        ]%
12632        *?(math)%
```

Support for xfakebold:

```
12633        \LWR@applyxfakebold%
```

Create the image contents using an actual eqnarray:

```
12634        \LWR@origeqnarray%
12635        \BODY%
12636        \LWR@origendeqnarray%
12637        \end{lateximage}%
12638        \end{BlockClass}%
12639      }%
12640      {% not LWR@numbereqnarray
```

If not numbered, do the same, but an extra \nonumber seems to be required:

```
12641        \begin{BlockClass}{displaymath}%
12642        \LWR@newautoidanchor%
12643        \booltrue{LWR@indisplaymathimage}%
12644        \begin{lateximage}[%
12645            \LWR@addmathjax{eqnarray*}{\BODY}%
12646        ]*?(math)%
```

Support for xfakebold:

```
12647        \LWR@applyxfakebold%

12648        \def\@eqncr{\nonumber\@seqncr}%
12649        \csuse{LWR@origeqnarray}%
12650        \BODY%
12651        \nonumber\csuse{LWR@origendeqnarray}%
12652        \end{lateximage}%
12653        \end{BlockClass}%
12654      }% LWR@numbereqnarray
12655    }% not mathjax
```

Default to number equations in the future:

```
12656    \booltrue{LWR@numbereqnarray}%
```

Clear the single-use `alt` text:

```
12657    \gdef\LWR@ThisAltText{}%
12658 }
```

eqnarray itself is made with a blank line before and after to force it to be on its own line:

```
12659 \RenewEnviron{eqnarray}
12660 {%
12661
12662 \LWR@eqnarrayfactor
12663
12664 }
```

The starred version is patched to turn off the numbering:

```
12665 \csgpreto{eqnarray*}{\boolfalse{LWR@numbereqnarray}}
```

```
12666 \end{warpHTML}
```

86 Lateximages

86.1 Description

lateximage (*env.*) A lateximage is a piece of the document which is typeset in LaTeX then included in the HTML output as an image. This is used for math if SVG math is chosen, and also for the picture, tikzpicture, and other environments.

Before typesetting the lateximage a large number of formatting, graphics, and symbols-related macros are temporarily restored to their print-mode meaning by \LWR@restoreorigformatting. (See section 83.)

A lateximage is typeset on its own PDF page inside an HTML comment which starts on the preceeding page and ends on following page, and instructions are written to lateximage.txt for *lwarpmk* to extract the lateximage from the page of the PDF file then generate an accompanying .svg file image file. Meanwhile, instructions to show this image are placed into the HTML file after the comment.

An HTML <span> is created to hold both the HTML comment, which will have the *pdftotext* conversion, and also the link to the final .svg image.

A LaTeX label is used to remember which PDF page has the image. A label is used because footnotes, endnotes, and pagenotes may cause the image to appear at a later time. The label is declared along with the image, and so it correctly remembers where the image finally ended up.

HTML alt tag The HTML alt tag is set to the LaTeX source for SVG math, some chemistry expressions, and perhaps some other expressions which make sense for text copy/paste. In some other cases, the alt tag is set according to the package name.

When creating an SVG math image, its HTML alt tag may be set to the math expression, which may be hashed for image reuse. In the case of \ensuremath or

after \inlinemathother, where the contents require a unique image for each instance of the same expression, the alt tag is set to \MathImageAltText, along with \AltTextOpen and \AltTextClose, and the image is not reused.

This alt expression is visible in the browser if images are not loaded, and appears when the text is copied and pasted. The default is "math image", and it may be changed according to the document's language. This may be set in the preamble, or changed as necessary inside the document, where it will affect the following svg math images.

For many packages, the output is placed inside a lateximage with an html alt tag set to the package name followed by \PackageDiagramAltText. For example:

(-xy- diagram)

This expression is visible in the browser if images are not loaded, and appears when the text is copied and pasted. The default is "diagram", and may it be changed according to the document's language. This may be set in the preamble, or changed as necessary inside the document, where it will affect the following package diagrams.

svg image font size For svg display math and the lateximage environment, the size of the math and text used in the svg image may be adjusted by setting \LateximageFontSizeName to a font size name — *without the backslash*, which defaults to:

\renewcommand{\LateximageFontSizeName}{normalsize}

For inline svg math, font size is instead controlled by \LateximageFontScale, which defaults to:

\newcommand*{\LateximageFontScale}{.75}

86.2 Support counters and macros

12667 \begin{warpHTML}

LWR@lateximagenumber (*Ctr*) Sequence the images.

12668 \newcounter{LWR@lateximagenumber}
12669 \setcounter{LWR@lateximagenumber}{0}

LWR@lateximagedepth (*Ctr*) Do not create \lateximage inside of \lateximage.

12670 \newcounter{LWR@lateximagedepth}
12671 \setcounter{LWR@lateximagedepth}{0}

A few utility macros to write special characters:

12672 \edef\LWR@hashmark{\string#} % for use in \write
12673 \edef\LWR@percent{\@percentchar} % for use in \write

LWR@LIpage (*Ctr*) Used to reference the PDF page number of a lateximage to be written into <project>-images.txt.

12674 \newcounter{LWR@LIpage}

12675 \end{warpHTML}

86.3 Font size

12676 `\begin{warpall}`

\LateximageFontSizeName Declares how large to write text in \lateximages. The .svg file text size should blend well with the surrounding HTML text size.

⚠ no backslash *Do not include the leading backslash in the name.*

12677 `\newcommand*{\LateximageFontSizeName}{normalsize}`

\LateximageFontScale Declares how large to scale inline svg math images. The .svg file text size should blend well with the surrounding HTML text size. The default is 1, but it may be redefined as needed depending on the HTML font.

12678 `\newcommand*{\LateximageFontScale}{1}`

12679 `\end{warpall}`

86.4 Equation numbers

12680 `\begin{warpHTML}`

LWR@startingequation (*Ctr*) For use with lateximage and multi-line numbered equations. Remembers the next equation number so that it may be printed in the alt tag.

```
12681 \newcounter{LWR@startingequation}
12682
12683 \@ifundefined{chapter}
12684 {
12685 \renewcommand{\theLWR@startingequation}{%
12686    \arabic{LWR@startingequation}%
12687 }
12688 }
12689 {% chapter defined
12690 \renewcommand{\theLWR@startingequation}{%
12691    \ifnumcomp{\value{chapter}}{>}{0}{\arabic{chapter}.}{}%
12692    \arabic{LWR@startingequation}%
12693 }
12694 }
```

LWR@isstartingequation (*bool*) True for the first equation tag, false for later tags in the same environment.

12695 `\newbool{LWR@isstartingequation}`

\LWR@startingequationtag Prints the starting equation number or tag.

12696 `\let\LWR@startingequationtag\theLWR@startingequation`

\LWR@equationtag Prints the ending equation number or tag.

This is reset by lateximage, may be temporarily overwritten by \tag calling \LWR@remembertag.

12697 `\newcommand*{\LWR@equationtag}{}`

Only if svg math, patch \tag after packages have loaded, in case someone else modified \tag.

```
12698 \AtBeginDocument{
12699
12700 \ifbool{mathjax}{}{% not mathjax
```

\LWR@remembertag {⟨*tag*⟩}

For use inside the math environments while using svg math. Sets \theLWR@startingequation and \theequation to the given tag.

```
12701 \NewDocumentCommand{\LWR@remembertag}{m}{%
12702     \ifbool{LWR@isstartingequation}%
12703     {%
12704         \global\boolfalse{LWR@isstartingequation}%
12705         \xdef\LWR@startingequationtag{#1}%
12706     }{}%
12707     \xdef\LWR@equationtag{#1}%
12708 }%
12709
12709 }% not mathjax
12710 }% AtBeginDocument
```

86.5 html alt tags

\LWR@amsmathbody {⟨*envname*⟩}   For use inside the optional argument to a lateximage to add the contents of a AMS math environment to the <alt> tag.

```
12711 \newcommand*{\LWR@amsmathbody}[1]
12712 {%
12713     \textbackslash\{begin\}\{#1\} % extra space
12714     \LWR@HTMLsanitizeexpanded{\detokenize\expandafter{\the\@envbody}}%
12715     \textbackslash\{end\}\{#1\}%
12716 }
```

\LWR@amsmathbodynumbered {⟨*envname*⟩}   For use inside the optional argument to a lateximage to add the contents of a AMS math environment to the alt tag, prefixed by the equation numbers.

```
12717 \newcommand*{\LWR@amsmathbodynumbered}[1]
12718 {%
12719     \ifnumcomp{\value{LWR@startingequation}}{=}{\value{equation}}%
12720         {(\LWR@equationtag)}%
12721         {(\LWR@startingequationtag\textendash\LWR@equationtag)} % extra space
12722     \LWR@amsmathbody{#1} % extra space
12723 }
```

86.6 lateximage environment

\LWR@lateximage@oneimageb {⟨*1: alt text*⟩} {⟨*2: filename*⟩} {⟨*3: css style*⟩} {⟨*4: aria role*⟩} Creates the html image tag for the lateximage.

To allow sanitization of the alt tag, if the alt tag is detokenized then it must also
be expanded before being used here.

```
12724 \newcommand{\LWR@leximage@oneimageb}[4]{%
12725     \LWR@subinlineimage{#1}{lateximage}%
12726     {%
12727         \LWR@print@mbox{%
12728             \LWR@ImagesDirectory\OSPathSymbol%
12729             #2%
12730         }%
12731     }{svg}{#3}{#4}%
12732 }
```

\LWR@lateximage@oneimage {⟨*1: alt text*⟩} {⟨*2: filename*⟩} {⟨*3: css style*⟩} {⟨*4: delimit?*⟩} {⟨*5: aria role*⟩}

Creates an image for the lateximage, whose alt text depends on the circum-
stances.

To allow sanitization of the alt tag, if the alt tag is detokenized then it must also
be expanded before being used here.

```
12733 \newcommand{\LWR@lateximage@oneimage}[5]{%
12734     \LWR@traceinfo{LWR@lateximage@oneimage !#1!#2!#3!#4!#5!}%
12735     \ifdefvoid{\LWR@ThisAltText}{%
12736         \IfBooleanTF{#4}{%
12737             \LWR@lateximage@oneimageb{#1}{#2}{#3}{#5}%
12738         }{%
12739             \LWR@lateximage@oneimageb%
12740                 {\AltTextOpen#1\AltTextClose}%
12741                 {#2}{#3}{#5}%
12742         }%
12743     }{%
12744         \LWR@lateximage@oneimageb%
12745             {\AltTextOpen\LWR@ThisAltText\AltTextClose}%
12746             {#2}{#3}{#5}%
12747     }%
12748 }
```

lateximage (*env.*) * [⟨*2: <alt> tag*⟩] * ? [⟨*5: add'l hashing*⟩] [⟨*6: css style*⟩] (⟨*7: aria role*⟩)

Typesets the contents and then renders the result as an svg file. Star #1 causes
the image to be hashed for reuse. Star #3 causes the alt tag to not include
\AltTextOpen and \AltTextClose, for use with math expressions. Question mark
#4 does not detokenize the alt tag (for internal use). This allows the use of macros
inside the alt tag.

The optional <alt> tag is included in the HTML code for use with copy/paste.

image filename hashing If starred, a hashed filename is used.If so, the hash is based on the alt tag and
also the additional hashing argument.

This may be used to provide an expression with a simple alt tag but also enough
additional information to provide a unique hash.

An example is when the expression is a complicated TEX expression, which would
not copy/paste well. A simplified tag may be used, while the complicated expres-
sion is used in the additional hashing argument to ensure a unique image.

Another example is when the expression is simple, but the image depends on options. These options may be decoded into text form and included in the additional hashing argument in order to make the hash unique according to the set of options, even if the simple alt tag is still the same.

*_html.aux (*file*)   A new label is placed into the file *_html.aux:

$$\newlabel{LWRlateximage-<BaseJobname>-<number>}{{<x>}{<y>}}$$

This is used to find the image in the PDF file, according to its name.

*-images.txt (*file*)   A list of images to generate is created in <jobname>-images.txt. Each line has three pipe-delimited fields, containing the PDF page number from <jobname>_html.pdf, where the image is located, a boolean indicating whether the image is hashed, and the filename of the image. The last line has "end" in each field, and is used to detect an incomplete compile.

```
12749 \catcode`\$=\active%
12750
12751 \NewDocumentEnvironment{lateximage}{s o s t? O{} O{} D(){}}%
12752 {%
12753 \LWR@traceinfo{lateximage !#1!#2!#3!#4!#5!#6!}%
12754 \LWR@traceinfo{lateximage: starting on \jobname.pdf page \arabic{page}}%
12755 \LWR@traceinfo{lateximage: entering depth is \arabic{LWR@lateximagedepth}}%
```

Nested lateximages remain one large lateximage:

```
12756 \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
```

If nesting inside an already-existing lateximage, simply record one more level. $\mathcal{A}_{\mathcal{M}}\mathcal{S}$ packages redefine \addtocounter to do nothing if inside a \text, so lower-level TEX macros are used for tracking nested lateximages.

```
12757 {%
12758 %     \addtocounter{LWR@lateximagedepth}{1}%
12759    \global\advance\c@LWR@lateximagedepth 1\relax% Due to AmS \text macro.
12760 }%
```

Otherwise, this is the outer-most lateximage:

```
12761 {% start of outer-most lateximage
```

Remember the next equation number to be allocated, in case it must be printed in a multi-equation environment:

```
12762    \LWR@traceinfo{lateximage: starting outer-most lateximage}%
12763    \setcounter{LWR@startingequation}{\value{equation}}%
12764    \addtocounter{LWR@startingequation}{1}%
12765    \booltrue{LWR@isstartingequation}%
12766    \let\LWR@startingequationtag\theLWR@startingequation%
```

The default equation tag, unless overwritten by \tag:

```
12767    \let\LWR@equationtag\theequation%
```

Starting a new lateximage:

```
12768        \addtocounter{LWR@lateximagenumber}{1}%
12769    \LWR@traceinfo{lateximage: LWR@lateximagenumber is \arabic{LWR@lateximagenumber}}%
```

While inside a lateximage, locally do not use mathjax:

```
12770        \boolfalse{mathjax}%
```

Be sure that are doing a paragraph:

```
12771        \LWR@ensuredoingapar%
```

Inside the lateximage, temporarily prevent underfull \hbox warnings.

```
12772        \hbadness=10000\relax%
```

Next file:

```
12773        \addtocounter{LWR@externalfilecnt}{1}%
12774    \LWR@traceinfo{lateximage: LWR@externalfilecnt is \arabic{LWR@externalfilecnt}}%
```

Figure out what the next page number will be. \setcounterpageref assigns
LWR@LIpage to the page number for the reference LWRlateximage-BaseJobname-XXX:

```
12775        \setcounterpageref{LWR@LIpage}{%
12776            LWRlateximage-\BaseJobname-\arabic{LWR@lateximagenumber}%
12777        }%
12778    \LWR@traceinfo{lateximage: LWR@LIpage is \arabic{LWR@LIpage}}%
```

Create an HTML span which will hold the comment which contains the *pdftotext*
translation of the image's page, and also will hold the link to the .svg file:

```
12779        \LWR@htmltag{span\LWR@indentHTML%
12780            id=\textquotedbl{}%
12781                lateximage-\BaseJobname-\arabic{LWR@lateximagenumber}%
12782                \textquotedbl\LWR@indentHTML
12783            class=\textquotedbl{}lateximagesource\textquotedbl\LWR@orignewline
12784        }%
```

Write instructions to the <ImagesDirectory>.txt file:

```
12785        \LWR@traceinfo{lateximage: about to write to \BaseJobname-images.txt}%
12786    \IfBooleanTF{#1}% starred
12787    {% hash
```

Compute and save the hashed file name for later use:

```
12788        \ifdefvoid{\LWR@ThisAltText}{%
12789            \IfBooleanTF{#3}{% no open/close tags?
12790                \IfBooleanTF{#4}% '?': don't detokenize
12791                    {% don't detokenize
12792                        \edef\LWR@hashedname{%
12793                            \LWR@mdfive{#2-!-#5}%
12794                        }%
12795                    }%
12796                    {% do detokenize
12797                        \edef\LWR@hashedname{%
12798                            \LWR@mdfive{\detokenize\expandafter{#2}-!-#5}%
12799                        }%
```

```
12800                         }%
12801                 }{%
12802                         \IfBooleanTF{#4}% '?': don't detokenize
12803                         {% don't detokenize
12804                             \edef\LWR@hashedname{%
12805                                 \LWR@mdfive{\AltTextOpen#2\AltTextClose-!-#5}%
12806                             }%
12807                         }%
12808                         {% do detokenize
12809                             \edef\LWR@hashedname{%
12810                             \LWR@mdfive{\detokenize\expandafter{\AltTextOpen#2\AltTextClose}-!-#5}%
12811                             }%
12812                         }%
12813                 }%
12814         }{%
12815             \edef\LWR@hashedname{%
12816             \LWR@mdfive{\detokenize\expandafter{\AltTextOpen\LWR@ThisAltText\AltTextClose}-!-#5}%
12817             }%
12818         }%
12819         \LWR@traceinfo{lateximage: hash is \LWR@hashedname}%
```

Write the page, hashing, and hashed name:

```
12820         \immediate\write\LWR@lateximagesfile{%
12821             |\arabic{LWR@LIpage}|true|\LWR@hashedname|%
12822         }%
12823     }% hash
12824     {% no hash
```

No hash, so write the page, no hashing, and the image number:

```
12825         \LWR@traceinfo{lateximage: hash false}%
12826         \immediate\write\LWR@lateximagesfile{%
12827         |\arabic{LWR@LIpage}|false|\LWR@ImagesName\arabic{LWR@externalfilecnt}|%
12828         }%
12829     }% no hash
```

Place an open comment tag. This will hide any traces of the lateximage PDF page which were picked up by *pdftotext*.

```
12830     \LWR@traceinfo{lateximage: about to create open comment}%
12831     \LWR@htmlopencomment%
```

One level deeper. At this outer-most lateximage, it is known that this is not being used inside an $\mathcal{A}\mathcal{M}\mathcal{S}$ \text, since the outer-most level will never be in math mode.

```
12832     \addtocounter{LWR@lateximagedepth}{1}%
```

Start the new PDF page:

```
12833     \LWR@traceinfo{lateximage: about to create a new page}%
12834     \LWR@maybe@orignewpage%
```

If the current page is larger, typeset the image in a "standard" width page and font size:

```
12835     \LWR@traceinfo{lateximage: about to create minipage}%
12836     \setcounter{LWR@mpfootnote@store}{\value{mpfootnote}}
```

```
12837    \ifdimless{\linewidth}{6in}{%
12838        \LWR@print@minipage{\linewidth}%
12839    }{%
12840        \LWR@print@minipage{6in}%
12841    }%
```

```
12842    \ifnumgreater{\value{LWR@minipage@depth}}{0}%
12843        {\setcounter{mpfootnote}{\value{LWR@mpfootnote@store}}}%
12844        {}%
12845    \@nameuse{LWR@print@\LateximageFontSizeName}%
```

Temporarily restore formatting to its PDF definitions: Do not produce HTML tags for \hspace, etc. inside a lateximage.

```
12846    \LWR@traceinfo{lateximage: about to temporarily restore formatting}%
12847    \LWR@restoreorigformatting%
```

If not inside a minipage, use full-page footnotes instead of minipage footnotes. These become HTML footnotes.

```
12848    \ifnumgreater{\value{LWR@minipage@depth}}{0}%
12849        {}%
12850        {%
12851            \def\@mpfn{footnote}%
12852            \def\thempfn{\thefootnote}%
12853            \LetLtxMacro\@footnotetext\LWR@footnotetext%
12854        }%
```

Create the LWRlateximage-jobname-<number> label:

```
12855    \LWR@traceinfo{lateximage: about to create label
12856                  LWRlateximage-\BaseJobname-\arabic{LWR@lateximagenumber}}%
12857 \LWR@orig@label{LWRlateximage-\BaseJobname-\arabic{LWR@lateximagenumber}}%
12858    \LWR@traceinfo{lateximage: finished creating the label}%
```

Adjust the rule color to match HTML:

```
12859    \ifdefvoid{\LWR@ruleHTMLcolor}{}{%
12860        \LWR@print@arrayrulecolor[HTML]{\LWR@ruleHTMLcolor}%
12861    }%
```

Enable print-mode math functions:

```
12862    \LetLtxMacro$\LWR@origdollar%
12863    \catcode`\$=3% math shift
12864    \LetLtxMacro\(\LWR@origopenparen%
12865    \LetLtxMacro\)\LWR@origcloseparen%
```

Only enable print-mode display math if are not already inside display math:

```
12866    \ifbool{LWR@indisplaymathimage}{}{% not in display math
12867        \LetLtxMacro\[\LWR@origopenbracket%
12868        \LetLtxMacro\]\LWR@origclosebracket%
12869        \let\equation\LWR@orig@equation%
12870        \let\endequation\LWR@orig@endequation%
12871        \csletcs{equation*}{LWR@orig@equation*}%
12872        \csletcs{endequation*}{LWR@orig@endequation*}%
12873    }% not in display math
```

For chemformula:

```
12874    \LetLtxMacro\LWR@newsingledollar$%
12875    \LetLtxMacro\LWR@newsingledollar$% syntax highlighting
```

While inside a lateximage, do not use HTML tags for verbatim content, and do not sanitize HTML tags for <, >, &, etc.

```
12876    \boolfalse{LWR@verbtags}%
12877    \boolfalse{LWR@HTMLsanitize@tmpb@enable}%
```

```
12878 }% end of outer-most lateximage
12879 \LWR@traceinfo{lateximage: finished start of environment}%
12880 }% end of \begin{lateximage}
```

\endlateximage When the lateximage environment closes:

```
12881 {% start of \end{lateximage}
12882 \LWR@traceinfo{lateximage: starting end of lateximage}%
```

Nested more than one deep?

```
12883 \LWR@traceinfo{lateximage: internal depth was \arabic{LWR@lateximagedepth}}%
12884 \ifnumcomp{\value{LWR@lateximagedepth}}{>}{1}%
```

If nesting inside an already existing lateximage, simply record one less level. Uses a lower-level TEX macro due to $\mathcal{A}\mathcal{M}\mathcal{S}$ \text change of \addtocounter.

```
12885 {%
12886    \LWR@traceinfo{lateximage: unnesting}%
12887    \global\advance\c@LWR@lateximagedepth -1\relax%
12888 }%
```

If this is the outer-most lateximage:

```
12889 {% end of outer-most lateximage
```

Finish the lateximage minipage and start a new PDF page:

```
12890 \LWR@traceinfo{lateximage: ending outer-most lateximage}%
12891    \endLWR@print@minipage%
12892    \LWR@maybe@orignewpage%
```

Close the HTML comment which encapsulated any traces of the lateximage picked up by *pdftotext*:

```
12893    \LWR@print@vspace*{.5\baselineskip}%
12894    \LWR@htmlclosecomment%
12895    \LWR@traceinfo{lateximage: The page after the image is \arabic{page}}%
```

Create a link to the lateximage, allowing its natural height:

```
12896    \LWR@traceinfo{about to LWR@lateximage@oneimage !#2!}%
```

Generate the HTML link. Due to the need to correctly expand the alt tag, choose which of eight methods to use depending on whether to hash, whether an alt tag

was passed, and whether to detokenize the alt tag. If detokenizing the alt tag, it must be expanded now so it can be sanitized later.

```
12897    \IfBooleanTF{#1}% starred?
12898    {% starred, hash
12899        \IfValueTF{#2}% alt argument
12900        {% alt argument given
12901            \IfBooleanTF{#4}% '?': don't detokenize
12902            {% don't detokenize
12903                \LWR@lateximage@oneimage{#2}%
12904                    {\LWR@hashedname}{#6}{#3}{#7}%
12905            }% don't detokenize
12906            {% detokenize
12907            \expandafter\LWR@lateximage@oneimage\expandafter{\detokenize{#2}}%
12908                    {\LWR@hashedname}{#6}{#3}{#7}%
12909            }% detokenize
12910        }% alt argument given
12911        {% alt argument not given
12912            \IfBooleanTF{#4}% '?': don't detokenize
12913            {% don't detokenize
12914                \LWR@lateximage@oneimage{\ImageAltText}%
12915                    {\LWR@hashedname}{#6}{#3}{#7}%
12916            }% don't detokenize
12917            {% detokenize
12918            \expandafter\LWR@lateximage@oneimage\expandafter{\detokenize\expandafter{\ImageAltText}}%
12919                    {\LWR@hashedname}{#6}{#3}{#7}%
12920            }% detokenize
12921        }% alt argument not given
12922    }% starred, hash
12923    {% not starred, no hash
12924        \IfValueTF{#2}% alt argument
12925        {% alt argument given
12926            \IfBooleanTF{#4}% '?': don't detokenize
12927            {% don't detokenize
12928                \LWR@lateximage@oneimage{#2}%
12929                    {\LWR@ImagesName\theLWR@externalfilecnt}{#6}{#3}{#7}%
12930            }% don't detokenize
12931            {% detokenize
12932            \expandafter\LWR@lateximage@oneimage\expandafter{\detokenize{#2}}%
12933                    {\LWR@ImagesName\theLWR@externalfilecnt}{#6}{#3}{#7}%
12934            }% detokenize
12935        }% alt argument given
12936        {% alt argument not given
12937            \IfBooleanTF{#4}% '?': don't detokenize
12938            {% don't detokenize
12939                \LWR@lateximage@oneimage{\ImageAltText}%
12940                    {\LWR@ImagesName\theLWR@externalfilecnt}{#6}{#3}{#7}%
12941            }% don't detokenize
12942            {% detokenize
12943            \expandafter\LWR@lateximage@oneimage\expandafter{\detokenize\expandafter{\ImageAltText}}%
12944                    {\LWR@ImagesName\theLWR@externalfilecnt}{#6}{#3}{#7}%
12945            }% detokenize
12946        }% alt argument not given
12947    }% not starred, no hash
```

Be sure that are doing a paragraph:

```
12948    \LWR@ensuredoingapar%
```

Close the HTML span which has the *pdftotext* comment and also the link to the
.svg image:

```
12949     \LWR@htmltag{/span}%
12950     \ifbool{HTMLDebugComments}{%
12951         \LWR@htmlcomment{End of lateximage}%
12952     }{}%
```

Undo one lateximage level. This is not inside an $\mathcal{AMS}$ \text, so regular
\addtocounter may be used here.

```
12953     \addtocounter{LWR@lateximagedepth}{-1}%
```

Clear the single-use alt text:

```
12954     \gdef\LWR@ThisAltText{}%
12955 }% end of outer-most lateximage
12956 \LWR@traceinfo{lateximage: exiting depth is \arabic{LWR@lateximagedepth}}%
12957 \LWR@traceinfo{lateximage: done}%
12958 }%
12959 \catcode`\$=3% math shift
12960 \end{warpHTML}
```

for PRINT output: 12961 \begin{warpprint}

lateximage (*env.*) * [⟨<alt> *tag*⟩] * [⟨*add'l hashing*⟩] [⟨*css style*⟩]

Ignored in print mode.

```
12962 \NewDocumentEnvironment{lateximage}{s o s t? o o d()}
12963     {}{}
```

```
12964 \end{warpprint}
```

87   center, flushleft, flushright

for HTML output: 12965 \begin{warpHTML}

center (*env.*) Replace center functionality with css tags. In a <span>, these macros are nullified,
but extra % are used to remove spurious spaces here as well.

```
12966 \newenvironment*{LWR@HTML@center}
12967 {%
12968     \LWR@forcenewpage%
12969     \ifbool{FormatWP}%
12970         {\BlockClass[\LWR@print@mbox{text-align:center}]{center}}%
12971         {\BlockClass{center}}%
12972 }
12973 {\endBlockClass}
12974
12975 \LWR@formattedenv{center}
```

flushright (*env.*)

```
12976 \newenvironment*{LWR@HTML@flushright}
12977 {%
12978     \LWR@forcenewpage%
12979     \ifbool{FormatWP}%
12980         {\BlockClass[\LWR@print@mbox{text-align:right}]{flushright}}%
12981         {\BlockClass{flushright}}%
12982 }
12983 {\endBlockClass}
12984
12985 \LWR@formattedenv{flushright}
```

flushleft (*env.*)

```
12986 \newenvironment*{LWR@HTML@flushleft}
12987 {%
12988     \LWR@forcenewpage%
12989     \ifbool{FormatWP}%
12990         {\BlockClass[\LWR@print@mbox{text-align:left}]{flushleft}}%
12991         {\BlockClass{flushleft}}%
12992 }
12993 {\endBlockClass}
12994
12995 \LWR@formattedenv{flushleft}
```

\centering, \raggedleft, and \raggedright usually have no effect on the HTML output, but they may be used to compare with the next token to identify their use at the start of a float. See \LWR@floatalignment.

\centering

```
12996 \newcommand*{\LWR@HTML@centering}{%
12997     \ifbool{HTMLDebugComments}{%
12998         \LWR@htmlcomment{centering}%
12999     }{}%
13000 }
13001 \LWR@formatted{centering}
```

\raggedleft

```
13002 \newcommand*{\LWR@HTML@raggedleft}{%
13003     \ifbool{HTMLDebugComments}{%
13004         \LWR@htmlcomment{raggedleft}%
13005     }{}%
13006 }
13007 \LWR@formatted{raggedleft}
```

\raggedright

```
13008 \newcommand*{\LWR@HTML@raggedright}{%
13009     \ifbool{HTMLDebugComments}{%
13010         \LWR@htmlcomment{raggedright}%
13011     }{}%
13012 }
13013 \LWR@formatted{raggedright}
```

\leftline {⟨*text*⟩}

13014 \renewcommand{\leftline}[1]{\begin{flushleft}#1\end{flushleft}}

`\centerline {⟨text⟩}`

13015 \renewcommand{\centerline}[1]{\begin{center}#1\end{center}}

`\rightline {⟨text⟩}`

13016 \renewcommand{\rightline}[1]{\begin{flushright}#1\end{flushright}}

13017 \end{warpHTML}

88   Preloaded packages

13018 \begin{warpHTML}

If the given package was loaded before or by lwarp, load the lwarp version as well.

`\LWR@PreloadedPackage {⟨packagename⟩}`

```
13019 \newcommand*{\LWR@PreloadedPackage}[1]{%
13020     \IfPackageLoadedTF{#1}%
13021         {%
13022             \AtBeginDocument{
13023                 \LWR@origRequirePackage{lwarp-#1}%
13024             }
13025         }%
13026         {}%
13027 }
```

Undo nameref if already loaded, such as by memoir:

13028 \LWR@PreloadedPackage{nameref}

If inputtrc was loaded before lwarp, as is usually done, explicitly load the lwarp patches now:

13029 \LWR@PreloadedPackage{inputtrc}

If textcomp was loaded before lwarp, perhaps as part of the font-related packages, explicitly load the lwarp patches now:

13030 \LWR@PreloadedPackage{textcomp}

If xunicode was loaded before lwarp, perhaps as part of the font-related packages, explicitly load the lwarp patches now:

13031 \LWR@PreloadedPackage{xunicode}

If graphics or graphicx were loaded before lwarp, perhaps by xunicode, explicitly load the lwarp patches now:

13032 \LWR@PreloadedPackage{graphics}
13033 \LWR@PreloadedPackage{graphicx}

tagpdf-base may have been preloaded by pdfmanagement-testphase

13034 \LWR@PreloadedPackage{tagpdf-base}

scalefnt may have been preloaded by babel

13035 \LWR@PreloadedPackage{scalefnt}

fontaxes must be preloaded so that lwarp may patch it for HTML.

13036 \LWR@PreloadedPackage{fontaxes}

Various font packages which may be loaded before lwarp:

13037 \LWR@PreloadedPackage{cmbright}
13038 \LWR@PreloadedPackage{fourier}
13039 \LWR@PreloadedPackage{kpfonts}
13040 \LWR@PreloadedPackage{kpfonts-otf}
13041 \LWR@PreloadedPackage{libertinust1math}
13042 \LWR@PreloadedPackage{pxfonts}
13043 \LWR@PreloadedPackage{txfonts}
13044 \LWR@PreloadedPackage{txgreeks}
13045 \LWR@PreloadedPackage{newpxmath}
13046 \LWR@PreloadedPackage{newtxmath}
13047 \LWR@PreloadedPackage{newtxsf}
13048 \LWR@PreloadedPackage{mathalpha}
13049 \LWR@PreloadedPackage{unicode-math}

13050 \LWR@PreloadedPackage{realscripts}

nfssext-cfr may be preloaded by cfm-lm or related font packages.

13051 \LWR@PreloadedPackage{nfssext-cfr}

ulem may be preloaded by ctex, ctexart, and related classes.

13052 \LWR@PreloadedPackage{ulem}

13053 \LWR@PreloadedPackage{xetexko}

geometry is preloaded by lwarp, and perhaps by various classes.

13054 \LWR@PreloadedPackage{geometry}

plext is preloaded by some CJK classes.

13055 \LWR@PreloadedPackage{plext}

stfloats is preloaded by ltj* classes.

13056 \LWR@PreloadedPackage{stfloats}

lltjext is preloaded by ltj* classes.

13057 \LWR@PreloadedPackage{lltjext}

luatexko must be loaded before lwarp.

13058 \LWR@PreloadedPackage{luatexko}

13059 \end{warpHTML}

89 siunitx

siunitx (*Pkg*)

A few HTML unit equivalents are defined here.

siunitx is well supported by lwarp.

Limitations   Some general limitations:

fractions   Due to *pdftotext* limitations, fraction output is replaced by symbol output for `per-mode` and `quotient-mode`.

\cancel is not currently supported for siunitx v3.

Negative values are not automatically colored.

⚠ tabular   Tabular S and s columns are rendered as simple c columns, although key settings will be set. If using scientific notation, `table-format`, `table-align-uncertainty`, `drop-exponent`, etc.. use \tablenum for each cell. This is especially required for
⚠ drop-exponent   drop-exponent, without which the value will be shown incorrectly.

⚠ table-auto-round   table-auto-round is ignored.

Math rendering   Math may be rendered in several ways in the same document:

For math mode with SVG display: The original siunitx code is used while generating the SVG image.

For HTML text mode: lwarp uses siunitx code patched for HTML, and simplified units.

For math expressions while using MATHJAX: A limited emulation is used. Most functions work reasonably well, but many options cannot be emulated. The result usually looks fine, and otherwise is enough to get the meaning across.

Custom units   siunitx allows customized units:

\DeclareSIUnit   {⟨*name*⟩} {⟨*definition*⟩}

\DeclareSIUnit declares a version of the unit for the print version. This is also used when the unit is printed in SVG math or a lateximage. It is also used for HTML if an HTML-specific version is not defined with \HTMLDeclareSIUnit.

> \DeclareSIUnit\myunit{\ensuremath{\text{m}_y}}

\HTMLDeclareSIUnit   {⟨*name*⟩} {⟨*definition*⟩}

⚠ v3 only! Use this after the print unit has been defined. For siunitx v3, \HTMLDeclareSIUnit declares a simplified version of the unit for HTML, for example if the print-mode unit uses TEX boxes or \ensuremath:

    \HTMLDeclareSIUnit\myunit{\text{m}\textsubscript{\textit{y}}}

It is also possible to provide a custom unit for MATHJAX:

    \CustomizeMathJax{\newcommand{\myunit}{\text{m}_y}}

Predefined units   Most units work as-is with HTML. For the following units, lwarp has already set \HTMLDeclareSIUnit: \celsius, \arcminute, \arcsecond, \elementarycharge, \clight, \bohr, \electronmass, \hartree, \planckbar.

⚠ MathJax

Document modifications required for MATHJAX

⚠ \sisetup
- Place \sisetup in the preamble before \begin{document}. Changes made later may be ignored, especially with MATHJAX. The MATHJAX emulation also ignores most macro options.

⚠ complex numbers
- Complex numbers are displayed as entered, ignoring output-complex-root.

custom units
- Custom units may be added with \CustomizeMathJax. For example, from lwarp-common-mathjax-siunitx:

    \CustomizeMathJax{\newcommand{\hartree}{\mathit{E}_{\mathrm{h}}}}
    \CustomizeMathJax{\newcommand{\angstrom}{\mathrm{\unicode{x212B}}}}

⚠ unit spacing
- Units work better using ~ between units instead of using periods.

⚠ \square, \cubic
- To square or cube compound units, enclose the following compound units in braces:

    \cubic{\centi\meter}

  Single units do not require braces.

- For \numlist, the argument is printed as text as-is, so use space between semicolons for improved readability.

⚠ Missing $ inserted
- If using parse-numbers = false, also use \num or \qty. siunitx=siunitx>Missing $ inserted.

Also see MATHJAX option, section 8.7.5.

for HTML output: 13060 \begin{warpHTML}

Options for siunitx:

```
13061 \newrobustcmd{\LWR@siunitx@textcelsius}{\HTMLentity{deg}C}
13062 \newrobustcmd{\LWR@siunitx@textdegree}{\HTMLentity{deg}}
13063 \newrobustcmd{\LWR@siunitx@textprime}{\HTMLunicode{2032}}
13064 \newrobustcmd{\LWR@siunitx@textdblprime}{\HTMLunicode{2033}}
13065 \newrobustcmd{\LWR@siunitx@textplanckbar}{\text{\textit{\HTMLunicode{210F}}}}
13066
13067 \appto\LWR@restoreorigformatting{%
13068 \renewrobustcmd{\LWR@siunitx@textcelsius}{\text{\ensuremath{^\circ}C}}%
13069 \renewrobustcmd{\LWR@siunitx@textdegree}{\text{\ensuremath{^\circ}}}%
13070 \renewrobustcmd{\LWR@siunitx@textprime}{\text{\ensuremath{^\prime}}}%
```

```
13071 \renewrobustcmd{\LWR@siunitx@textdblprime}{\text{\ensuremath{^{\prime\prime}}}}%
13072 \renewrobustcmd{\LWR@siunitx@textplanckbar}{\text{\ensuremath{\hbar}}}%
13073 }
```

```
13074 \end{warpHTML}
```

for PRINT output: The print version of \HTMLDeclareSIUnit.

```
13075 \begin{warpprint}
13076 \NewDocumentCommand{\HTMLDeclareSIUnit}{o +m m}{}
13077 \end{warpprint}
```

90 Graphics print-mode modifications

90.1 General limitations

Per table 9, image filenames may be specified either with or without an extension. If an extension is given it will be used as-is, for either print or HTML output. If

file extensions   no extension is given, a list of possible extensions is tried, which depends on whether print or HTML is being generated. This allows a PDF file for print and a SVG file for HTML, for example. If no extension is given, the automatic search will

⚠ case sensitive   only return lowercase extensions, even if the filename actually has an uppercase extension, and lwarp cannot get around this problem, so image file extensions must be lowercase to be seen by the HTML browser with lwarp. For example, name the image file image.pdf instead of image.PDF, but refer to it in the source as image, without an extension. For images which may be used as-is with either print or HTML, such as JPG or PNG, you may use a capitalized extension if it is specified in the source, such as image.JPG.

\includegraphics file   For \includegraphics with .pdf or .eps files, the user must provide a .pdf or
formats   .eps image file for use in print mode, and also a .svg, .png, or .jpg version of the same image for use in HTML.

    \includegraphics{filename} % print:.pdf/.eps HTML:.svg, etc.

For print output, lwarp will automatically choose the .pdf or .eps format if available, or some other format otherwise. For HTML, one of the other formats is used instead.

If a .pdf or .eps image is referred to with its file extension, the extension will be changed to .svg for HTML:

    \includegraphics{filename.pdf} % uses .svg in html
    \includegraphics{filename.eps} % uses .svg in html

pdftocairo (*Prog*)   To convert a PDF image to SVG, use the utility *pdftocairo*:
PDF to SVG

    Enter ⇒   **pdftocairo -svg filename.pdf**

lwarpmk pdftosvg (*Prog*)   For a large number of images, use *lwarpmk*:

    Enter ⇒   **lwarpmk pdftosvg *.pdf**   **(or a list of filenames)**

lwarpmk epstopdf (*Prog*)   For EPS images converted to PDF using the package epstopdf, use
epstopdf (*Prog*)
epstopdf package

Enter ⇒ **lwarpmk pdftosvg *.PDF**

to convert to svg images.

DVI LATEX   When using DVI *latex*, it is necessary to convert EPS to PDF and then to svg:

Enter ⇒ **lwarpmk epstopdf *.eps**   **(or a list of filenames)**

Enter ⇒ **lwarpmk pdftosvg *.pdf**   **(or a list of filenames)**

PNG and JPG   For PNG or JPGwhile using *pdflatex*, *lualatex*, or *xelatex*, the same file may be used in both print or HTML versions, and may be used with a file extension, but will also be used without the file extension if it is the only file of its base name.

GIF   GIF files may be used for HTML, but another format must also be provided for print output.

file extension priorities   If a file extension is not used, for HTML the file extension priorities are: SVG, GIF, PNG, then JPG.

duplicate files   A complication occurs if a file of the same name exists elsewhere in the TEX tree,
⚠ image not displayed   such as a test image from some LATEX package. TEX looks in the local document directory before considering the directories specified by \graphicspath, but the TEX tree is found as "local", so any file in the tree is found before the directories in \graphicspath. To use such an image, it must be copied to the document's directory to be used for HTML, and furthermore must be in the document's base directory instead of an images subdirectory.

⚠ graphics vs. graphicx   If using the older graphics syntax, use both optional arguments for \includegraphics. A single optional parameter is interpreted as the newer graphicx syntax. Note that
⚠ viewport   viewports are not supported by lwarp — the entire image will be shown.

units   For \includegraphics, avoid px and % units for width and height, or enclose them inside warpHTML environments. For font-proportional image sizes, use ex or em. For fixed-sized images, use cm, mm, in, pt, or pc. Use the keys width=.5\linewidth, or similar for \textwidth or \textheight to give fixed-sized images proportional to a 6 by 9 inch text area. Do not use the scale option, since it is not well supported by HTML browsers.

options   \includegraphics accepts width and height, origin, rotate and scale, plus new class and alt keys. (alt has recently been incorportated into graphicx itself.)

HTML class   With HTML output, \includegraphics accepts an optional class=xyz keyval combination, and if this is given then the HTML output will include that class for the image. The class is ignored for print output.

HTML alt tags   Likewise, the \includegraphics alt key adds an HTML alt tag to an image, and is ignored for print output. If not assigned, each image is given an alt tag according to \ImageAltText.

⚠ scale   Avoid using the \includegraphics scale option. Change:

\includegraphics[scale=<xx>]{ . . . }

to:

\includegraphics[width=<yy>\linewidth]{ . . . }

\rotatebox   \rotatebox accepts the optional origin key.

⚠ browser support \rotatebox, \scalebox, and \reflectbox depend on modern browser support. The css3 standard declares that when an object is transformed the whitespace which they occupied is preserved, unlike LATEX, so expect some ugly results for scaling and rotating.

90.2 Print-mode modifications

for PRINT output: For print output, accept and then discard the new class key:

```
13078 \begin{warpprint}
13079 \define@key{Gin}{class}{}
```

Print-mode additions for the overpic package. See section 467 for the HTML version.

```
13080 \AtBeginDocument{
13081 \IfPackageLoadedTF{overpic}{
13082 \newcommand*{\overpicfontsize}{12}
13083 \newcommand*{\overpicfontskip}{14}
13084 }{}
13085 }
13086 \end{warpprint}
```

91 xcolor boxes

xcolor (Pkg) A few new definitions are provided for enhanced HTML colored boxes, and \fcolorbox is slightly modified. Print-mode version are also provided.

Print-mode versions of new xcolor defintions. These are defined inside warpall because they are also used for HTML while inside a lateximage. They are defined \AtBeginDocument so that the xcolor originals may first be loaded and saved for reuse.

The framed versions are modified to allow a background color of none, in which case only the frame is drawn, allowing the background page color to show.

for HTML & PRINT: 13087 \begin{warpall}

After xparse may have been loaded …

```
13088 \AtBeginDocument{
```

… and *only* if xcolor was loaded:

```
13089 \IfPackageLoadedTF{xcolor}{
13090 \LWR@traceinfo{patching xcolor}
```

The print version:

\colorboxBlock \colorboxBlock is the same as \colorbox:

```
13091 \LetLtxMacro\colorboxBlock\colorbox
```

The original definition is reused by the new versions:

13092 \LetLtxMacro\LWR@orig@print@fcolorbox\fcolorbox

\fcolorbox [⟨*framemodel*⟩] {⟨*framecolor*⟩} [⟨*boxmodel*⟩] {⟨*boxcolor*⟩} {⟨*text*⟩}

In print mode, \fcolorbox is modified to accept a background color of none.

(\fcolorbox is particular about its optional arguments, thus the elaborate combinations of \ifthenelse.)

13093 \newsavebox{\LWR@colorminipagebox}
13094
13095 \NewDocumentCommand{\LWR@print@fcolorbox}{o m o m +m}{%
13096   \LWR@traceinfo{LWR@print@fcolorbox #2 #4}%

Pre-load the contents into an LR box so that they can be used inside a \fcolorbox:

13097   \begin{lrbox}{\LWR@colorminipagebox}%
13098   #5%
13099   \end{lrbox}%

Sort out the various optional arguments and the background color of none. In each case, the LRbox is placed inside a \fcolorbox.

The current color is remembered, then set to the frame, then the current color is used for the contents.

13100   \ifstrequal{#4}{none}%
13101   {% #4 none
13102   \LWR@traceinfo{background is none}%
13103   {% scope the \colorlet
13104   \colorlet{LWR@currentcolor}{.}%
13105   \color{#2}%
13106   \fbox{%
13107   \color{LWR@currentcolor}%
13108   \usebox{\LWR@colorminipagebox}%
13109   }% fbox
13110   }% colorlet
13111   }% #4 none
13112   {% #4 not none
13113   \LWR@traceinfo{background not none}%
13114   \IfValueTF{#1}%
13115   {%
13116   \IfValueTF{#3}%
13117   {\LWR@orig@print@fcolorbox[#1]{#2}[#3]{#4}{\usebox{\LWR@colorminipagebox}}}%
13118   {\LWR@orig@print@fcolorbox[#1]{#2}{#4}{\usebox{\LWR@colorminipagebox}}}%
13119   }%
13120   {% no value #1
13121   \IfValueTF{#3}%
13122   {\LWR@orig@print@fcolorbox{#2}[#3]{#4}{\usebox{\LWR@colorminipagebox}}}%
13123   {\LWR@orig@print@fcolorbox{#2}{#4}{\usebox{\LWR@colorminipagebox}}}%
13124   }% no value #1
13125   }% #4 not none
13126   \LWR@traceinfo{LWR@print@fcolorbox done}%
13127 }

13128 \renewrobustcmd*{\fcolorbox}{\LWR@print@fcolorbox}%

\fcolorboxBlock [⟨*framemodel*⟩] {⟨*framecolor*⟩} [⟨*boxmodel*⟩] {⟨*boxcolor*⟩} {⟨*text*⟩}

In print mode, \fcolorboxBlock is the same as \fcolorbox.

13129 \newcommand*{\LWR@print@fcolorboxBlock}{\LWR@print@fcolorbox}

13130 \newrobustcmd*{\fcolorboxBlock}{\LWR@print@fcolorboxBlock}

fcolorminipage (*env.*) [⟨*1:framemodel*⟩] {⟨*2:framecolor*⟩} [⟨*3:boxmodel*⟩] {⟨*4:boxcolor*⟩} [⟨*5:align*⟩]
                         [⟨*6:height*⟩] [⟨*7:inner-align*⟩] {⟨*8:width*⟩}

In print mode, becomes a \fcolorbox containing a minipage:

13131 \NewDocumentEnvironment{fcolorminipage}{o m o m O{c} O{} o m}
13132 {%
13133   \LWR@traceinfo{*** fcolorminipage: #2 #4 #8}%

Pre-load the contents into an LR box so that they can be used inside a \fcolorbox:

13134   \begin{lrbox}{\LWR@colorminipagebox}%

If inner alignment is not given, use the outer alignment instead:

13135   \IfValueTF{#7}%
13136   {\begin{minipage}[#5][#6][#7]{#8}}%
13137   {\begin{minipage}[#5][#6][#5]{#8}}%
13138 }%
13139 {%
13140   \end{minipage}%
13141   \end{lrbox}%
13142   \LWR@traceinfo{*** starting end fcolorminipage #1 #2 #3 #4 #8}%

Sort out the various optional arguments and the background color of none. In
each case, the LRbox is placed inside a \fcolorbox.

The current color is remembered, then set to the frame, then the current color is
used for the contents.

13143   \ifstrequal{#4}{none}%
13144   {% #4 none
13145   {% scope the \colorlet
13146   \colorlet{LWR@currentcolor}{.}%
13147   \color{#2}%
13148   \fbox{%
13149   \color{LWR@currentcolor}%
13150   \usebox{\LWR@colorminipagebox}%
13151   }% fbox
13152   }% colorlet
13153   }% #4 none
13154   {% #4 not none
13155   \IfValueTF{#1}%
13156   {%
13157   \IfValueTF{#3}%
13158   {\LWR@orig@print@fcolorbox[#1]{#2}[#3]{#4}{\usebox{\LWR@colorminipagebox}}}%
13159   {\LWR@orig@print@fcolorbox[#1]{#2}{#4}{\usebox{\LWR@colorminipagebox}}}%
13160   }%
13161   {% no value #1
13162   \IfValueTF{#3}%

```
13163        {\LWR@orig@print@fcolorbox{#2}[#3]{#4}{\usebox{\LWR@colorminipagebox}}}%
13164          {\LWR@orig@print@fcolorbox{#2}{#4}{\usebox{\LWR@colorminipagebox}}}%
13165          }% no value #1
13166      }% #4 not none
13167      \LWR@traceinfo{*** finished end fcolorminipage}%
13168 }
```

xcolor is known to have been loaded, and provided HTML versions of the following, and the print versions are provide above, so now they may be \LW@formatted.

```
13169 \LWR@formatted{colorbox}
13170 \LWR@formatted{colorboxBlock}
13171 \LWR@formatted{fcolorbox}
13172 \LWR@formatted{fcolorboxBlock}
13173 \LWR@formattedenv{fcolorminipage}
```

```
13174 \LWR@traceinfo{xcolor patches done}
13175 }{}% xcolor loaded
13176 }% AtBeginDocument
```

```
13177 \end{warpall}
```

92   chemmacros environments

\makepolymerdelims and redox reactions must be enclosed in a lateximage during HTML output. These environments are provided here in print mode, and in the chemmacros code in HTML mode, as a high-level semantic syntax which automatically embeds the contents in a lateximage with an appropriate alt tag.

for PRINT output:
```
13178 \begin{warpprint}
```

```
13179 \AtBeginDocument{
13180 \IfPackageLoadedTF{chemmacros}{
```

Env **polymerdelims**

```
13181 \DeclareDocumentEnvironment{polymerdelims}{}
13182      {}{}
```

Env **redoxreaction**   {⟨*space above*⟩} {⟨*space below*⟩}

For print output, extra space is include above and below the image, and a lateximage is not necessary. This extra space must be enforced, even inside a float, so zero-width rules are used.

For the HTML version, see section .

```
13183 \DeclareDocumentEnvironment{redoxreaction}{m m}
13184      {\rule{0pt}{#1}}{\rule[-#2]{0pt}{#2}}
```

```
13185 }{}% chemmacros
13186 }% AtBeginDocument
```

```
13187 \end{warpprint}
```

93 cleveref

loading order cleveref and lwarp-cleveref with its associated macro patches are automatically preloaded at the end of the preamble via \AtEndPreamble and \AfterEndPreamble. This is done because the HTML conversion requires cleveref. The user's document may not require cleveref, thus the user may never explicitly load it, so during HTML output lwarp loads it last. If the user's document preamble uses cleveref options, or functions such as \crefname, then cleveref may be loaded in the user's preamble near the end, and lwarp's additional loading of cleveref will have no effect.

\AtEndPreable forces cleveref to be loaded last, if it has not yet been loaded by the user.

for HTML output:
```
13188 \begin{warpHTML}
13189
13190 \AtEndPreamble{
13191     \RequirePackage{cleveref}
13192 }
13193
13194 \end{warpHTML}
```

94 Preexisting label and reference definitions

Remember and patch some label-related defintions. These will be further encased and patched by other packages later.

\label and \pageref do NOT change their behavior according to print or HTML output, and thus do not use the \LWR@formatted system.

for HTML output: Not using \VerifyCommand for \label because various packages change \label.

```
13195 \begin{warpHTML}
13196
13197 \VerifyCommand[lwarp][latex]{\label}
13198     {3FB4D89D15FDC24A8AA3A2C581E3C48D}
13199
13200 \LetLtxMacro\LWR@orig@label\label
13201 \LetLtxMacro\label\LWR@new@label
13202
13203 \AtBeginDocument{%
13204 \LetLtxMacro\LWR@orig@pageref\pageref%
13205 \LetLtxMacro\pageref\LWR@new@pageref%
13206 }
```

\label Detokenize \@currentnamelabel to avoid bug if math is in the name. Uses lwarp labels instead of \@currentHref.

```
13207 \xpatchcmd{\LWR@orig@label}
13208     {{\@currentlabelname}}
13209     {{\detokenize\expandafter{\@currentlabelname}}}
13210     {}
13211     {
13212         \PackageWarning{lwarp}{%
13213         Could not patch \string\label:\string\@currentlabelname.\MessageBreak
13214         This may cause an error with section names or float captions\MessageBreak
```

```
13215            containing math, for example.\MessageBreak
13216            (Recent updates in the LaTeX kernel may make things work again%
13217        }
13218    }
```

```
13219 \end{warpHTML}
```

95 `picture` environment

picture (*env.*)   The `picture` environment is enclosed inside a `\lateximage`.

`13220 \begin{warpHTML}`

picture (*env.*)

```
13221 \BeforeBeginEnvironment{picture}{\begin{lateximage}[picture]}
13222
13223 \AfterEndEnvironment{picture}{\end{lateximage}}
```

```
13224 \end{warpHTML}
```

96 Minipages and Boxes

A CSS flexbox is used for minipages and parboxes, allowing external and internal vertical positioning.

⚠ inline   A line of text with an inline `minipage` or `\parbox` will have the `minipage` or `\parbox` placed onto its own line, because a paragraph is a block element and cannot be made `inline-block`.

placement   `minipages` and `\parboxes` will be placed side-by-side in HTML unless you place a `\newline` between them.

side-by-side   Side-by-side `minipages` may be separated by `\quad`, `\qquad`, `\enskip`, `\hspace`, `\hfill`, or a `\rule`. When inside a `center` environment, the result is similar in print and HTML. Paragraph tags are suppressed between side-by-side `minipages` and these spacing commands, but not at the start or end of the paragraph.

⚠ minipage in a span   There is limited support for `minipages` inside an HTML <span>. An HTML <div> cannot appear inside a <span>. While in a <span>, `minipages`, and `\parboxes`, and any enclosed lists have limited HTML tags, resulting in an "inline" format, without markup except for HTML breaks. Use `\newline` or `\par` for an HTML break.

⚠ minipage size   When using `minipage`, `\parbox`, and `fminipage`, a virtual 6×9 inch text area is used for `\linewidth`, `\textwidth`, and `\textheight`, both for sizing the minipage, and also for its contents.

if width is `\linewidth`   If a `minipage` or `\parbox` is assigned a width of exactly `\linewidth`, in HTML it is automatically given no HTML width, thus allowed to fill the line as needed, similar to how it appears in print output.

full-width if HTML   A new macro `\minipagefullwidth` requests that, during HTML output, the next

single minipage or \parbox be generated without an HTML width attribute, allowing it to be the full width of the display rather than the declared print-output width. This may be useful where the printed version's width makes no sense in HTML.

⚠ tabular, multicols   Inside a tabular or multicols environment, where the width depends on the browser window, \minipagefullwidth is effectively used by default for every

\UseMinipageWidths   minipage or \parbox inside the environment. \UseMinipageWidths may be used

\IgnoreMinipageWidths   to tell lwarp to honor the specified widths of all following minipages and \parboxes until the end of the local scope, and \IgnoreMinipageWidths may be used to tell lwarp to ignore the specified widths.

⚠ multicol   Inside a multicols, \linewidth is divided by the specified number of columns.

⚠ text alignment   Nested minipages adopt their parent's text alignment in HTML, whereas in regular LATEX PDF output they do not. Use a flushleft or similar environment in the child minipage to force a text alignment.

for HTML output: 13225 \begin{warpHTML}

96.1 Computed lengths

\LWR@minipagewidth (*Len*)   Used to convert the width into printable units.

13226 \newlength{\LWR@minipagewidth}

\LWR@minipageheight (*Len*)   Used to convert the height into printable units.

13227 \newlength{\LWR@minipageheight}

96.2 Virtual page size

LWR@virtualpagedepth (*Ctr*)   Used to only reset the line width at the outermost minipage.

13228 \newcounter{LWR@virtualpagedepth}
13229 \setcounter{LWR@virtualpagedepth}{0}

LWR@setvirtualpage (*env.*)   * [⟨*columns*⟩]

If not nesting a minipage, adjust \linewidth, \textwidth, and \textheight for a virtual 6×9 page, and start on a new PDF page to help prevent page overflows.

If starred, force a new page in the PDF before generating more HTML. This may be done to reduce the chance of page overflow when starting a new minipage.

The optional number of columns defaults to 1.

13230 \NewDocumentEnvironment{LWR@setvirtualpage}{s O{1}}{%
13231   \ifnumequal{\value{LWR@virtualpagedepth}}{0}{%
13232   \IfBooleanT{#1}{\LWR@maybe@orignewpage}%
13233   \setlength{\linewidth}{6in/#2}%
13234   \setlength{\textwidth}{6in}%
13235   \setlength{\textheight}{9in}%
13236   }{}%
13237   \addtocounter{LWR@virtualpagedepth}{1}%

```
13238 }
13239 {\addtocounter{LWR@virtualpagedepth}{-1}}
```

96.3   Footnote handling

Also see section 62 for other forms of footnotes. Minipage footnotes are gathered in section 62.5, and then placed into the document in section 96.4.

96.4   Minipage handling

LWR@minipagefullwidth (*bool*)   Should the next minipage have no HTML width?

```
13240 \newbool{LWR@minipagefullwidth}
13241 \boolfalse{LWR@minipagefullwidth}
```

LWR@forceminipagefullwidth   Should the next minipage have no HTML width? Used to force full width for all
(*bool*)   minipages in an environment such as tabular or multicols, where the actual
width depends on the browser width. Controlled by \useminipagewidths and
\ignoreminipagewidths.

```
13242 \newbool{LWR@forceminipagefullwidth}
13243 \boolfalse{LWR@forceminipagefullwidth}
```

\minipagefullwidth   Requests that the next minipage have no width tag in HTML:

for HTML output: `13244 \newcommand*{\minipagefullwidth}{\global\booltrue{LWR@minipagefullwidth}}`

\UseMinipageWidths   Locally requests that minipage widths be honored.

```
13245 \newcommand*{\UseMinipageWidths}{\boolfalse{LWR@forceminipagefullwidth}}
```

\IgnoreMinipageWidths   Locally requests that minipage widths be ignored.

```
13246 \newcommand*{\IgnoreMinipageWidths}{\booltrue{LWR@forceminipagefullwidth}}
13247 \end{warpHTML}
```

for PRINT output:
```
13248 \begin{warpprint}
13249 \newcommand*{\minipagefullwidth}{}
13250 \newcommand*{\UseMinipageWidths}{}
13251 \newcommand*{\IgnoreMinipageWidths}{}
13252 \end{warpprint}
```

for HTML output: `13253 \begin{warpHTML}`

LWR@minipagethispar (*bool*)   Has a minipage been seen this paragraph? If true, prevents paragraph tags around
horizontal space between minipages.

```
13254 \newbool{LWR@minipagethispar}
13255 \boolfalse{LWR@minipagethispar}
```

LWR@minipage@depth (*Ctr*) Used to track whether to change footnote styles in a lateximage inside an HTML minipage.

```
13256 \newcounter{LWR@minipage@depth}
13257 \setcounter{LWR@minipage@depth}{0}
```

LWR@mpfootnote@store (*Ctr*) Used to maintain minipage footnote number while nesting inside a lateximage.

```
13258 \newcounter{LWR@mpfootnote@store}
```

minipage (*env.*) [⟨*vert position*⟩] [⟨*height*⟩] [⟨*inner vert position*⟩] {⟨*width*⟩}

The vertical positions may be 'c', 't', or 'b'. The inner position may also be 's'.

When using \linewidth, \textwidth, or \textheight, these are scaled proportionally to a 6×9 inch text area.

```
13259 \NewDocumentEnvironment{LWR@HTML@sub@minipage}{m m m m}
13260 {%
13261 \LWR@traceinfo{minipage}%
```

Start an environment, in which width and height is computed based on a virtual page size instead of the extra-large PDF page used during HTML tag generation.

```
13262 \begin{LWR@setvirtualpage}*%
```

Save the requested width now that \linewidth, etc. are adjusted to virtual size.

```
13263 \setlength{\LWR@minipagewidth}{#4}%
13264 \ifnumequal{\value{LWR@virtualpagedepth}}{1}{%
13265     \addtolength{\LWR@minipagewidth}{3em}% room for frames
13266 }{}%
13267 \LWR@traceinfo{computed width is \LWR@printlength{\LWR@minipagewidth}}%
```

Compute height:

```
13268 \setlength{\LWR@minipageheight}{\textheight}% default unless specified
13269 \ifblank{#2}{}{\setlength{\LWR@minipageheight}{#2}}%
```

LaTeX wants to start a paragraph for the virtual minipage, then start a paragraph again for the contents of the minipage, so cancel the paragraph tag handling until the minipage has begun.

```
13270 \ifbool{FormatWP}{\newline}{}%
13271 \LWR@stoppars%
```

If FormatWP, add a text frame:

```
13272 \ifbool{FormatWP}{%
13273
13274 \addtocounter{LWR@thisautoidWP}{1}%
13275 \LWR@htmltag{%
13276     div id=\textquotedbl%
13277         \LWR@print@mbox{autoidWP-\arabic{LWR@thisautoidWP}}%
13278     \textquotedbl\ % space
13279     class=\textquotedbl{}wpminipage\textquotedbl%
13280 }%
```

13281
13282 }{}%

Create the <div> tag with optional alignment style:

13283 \LWR@traceinfo{minipage: creating div class}%
13284 \LWR@htmltag{div class=\textquotedbl{}minipage\textquotedbl\ style=\textquotedbl%
13285 \ifthenelse{\equal{#1}{t}}{\LWR@print@mbox{vertical-align:bottom} ; }{}%
13286 \ifthenelse{\equal{#1}{c}}{\LWR@print@mbox{vertical-align:middle} ; }{}%
13287 \ifthenelse{\equal{#1}{b}}{\LWR@print@mbox{vertical-align:top} ; }{}%
13288 \ifthenelse{\equal{#3}{t}}{\LWR@print@mbox{justify-content:flex-start} ; }{}%
13289 \ifthenelse{\equal{#3}{c}}{\LWR@print@mbox{justify-content:center} ; }{}%
13290 \ifthenelse{\equal{#3}{b}}{\LWR@print@mbox{justify-content:flex-end} ; }{}%
13291 \ifthenelse{\equal{#3}{s}}{\LWR@print@mbox{justify-content:space-between} ; }{}%

Print the width and optional height styles:

13292 \LWR@traceinfo{minipage: about to print the width of \LWR@printlength{\LWR@minipagewidth}}%
13293 \ifbool{LWR@minipagefullwidth}%
13294 {\global\boolfalse{LWR@minipagefullwidth}}%
13295 {%
13296   \ifbool{LWR@forceminipagefullwidth}%
13297   {}%
13298   {%
13299   \ifdimequal{#4}{\linewidth}%
13300   {}%
13301   {width:\LWR@printlength{\LWR@minipagewidth} ; }%
13302   }%
13303 }%
13304 \LWR@traceinfo{minipage: about to print the height}%
13305 \ifblank{#2}{}{height:\LWR@printlength{\LWR@minipageheight} ; }%
13306 \textquotedbl%
13307 }%

Finish with an empty line to start the contents on a new line.

13308
13309 % The preceding empty line is required.

Set the user-accessible line and text width and height values inside the virtual minipage. These do not affect the actual size of the PDF output, but are used by any reference to \linewidth, etc. inside the virtual minipage being created here. \LWR@minipagewidth was the original then padded by 3em, which is restored here. This is done instead of settings back to #4, in case #4 was \linewidth, which was changed to 6in above.

13310 \ifnumequal{\value{LWR@virtualpagedepth}}{1}{%
13311   \addtolength{\LWR@minipagewidth}{-3em}% undo frame padding
13312 }{}%
13313 \setlength{\linewidth}{\LWR@minipagewidth}%

\raggedright cancels hyphenation, which will be done by HTML instead.

13314 \LWR@print@raggedright%

13315   \LWR@newautopagelabel{page}%

Set minipage footnotes:

```
13316 \def\@mpfn{mpfootnote}%
13317 \def\thempfn{\thempfootnote}\c@mpfootnote\z@%
13318 \let\@footnotetext\@mpfootnotetext%
```

Track depth for lateximage footnote type:

```
13319 \addtocounter{LWR@minipage@depth}{1}%
```

Resume paragraph tag handling for the contents of the minipage:

```
13320 \LWR@startpars%
13321 \ifboolexpr{bool{FormatWP} and bool{WPMarkMinipages}}{%
13322
13323 === begin minipage ===
13324
13325 }{}%
13326 \LWR@traceinfo{minipage: finished starting the minipage}%
13327 }% finished \minipage
13328 {% \endminipage
```

Print pending minipage footnotes:

```
13329 \LWR@printpendingmpfootnotes%
```

End the environment with closing tag:

```
13330 \ifboolexpr{bool{FormatWP} and bool{WPMarkMinipages}}{%
13331
13332 === end minipage ===
13333
13334 }{}%
13335 \LWR@stoppars%
13336
13337 \ifbool{FormatWP}{%
13338
13339 \LWR@htmlelementend{div}%
13340
13341 }{}%
```

Wrapup:

```
13342 \addtocounter{LWR@minipage@depth}{-1}%

13343 \LWR@htmldivclassend{minipage}%
13344
13345 \end{LWR@setvirtualpage}%
13346 \LWR@startpars%
13347 \ifbool{FormatWP}{\newline}{}%
```

Prevent paragraph tags around horizontal white space until the start of the next paragraph:

```
13348 \global\booltrue{LWR@minipagethispar}%
13349 \LWR@traceinfo{LWR@minipage: done}%
13350 }
13351
```

```
13352 \NewDocumentEnvironment{LWR@HTML@minipage}{O{t} O{} O{t} m}
13353     {\LWR@HTML@sub@minipage{#1}{#2}{#3}{#4}}
13354     {\endLWR@HTML@sub@minipage}
13355
13356 \LWR@formattedenv{minipage}
```

96.5 \parbox, \mbox, \makebox, \framebox, \fbox, \raisebox

\parbox [⟨*pos*⟩] [⟨*height*⟩] [⟨*inner-pos*⟩] {⟨*width*⟩} {⟨*text*⟩}

A parbox uses the minipage code:

```
13357 \NewDocumentCommand{\LWR@HTML@parbox}{O{t} O{} O{t} m +m}
13358 {
13359 \LWR@traceinfo{parbox of width #4}%
13360 \begin{minipage}[#1][#2][#3]{#4}%
13361 #5
13362 \end{minipage}%
13363 }
13364
13365 \LWR@formatted{parbox}
```

\mbox {⟨*text*⟩}   Nullified for HTML.

```
13366 \newcommand*{\LWR@HTML@mbox}[1]{{#1}}
13367
13368 \LWR@formatted{mbox}
```

\LWR@@makebox@paren {⟨*width*⟩} , {⟨*height*⟩}

Adds to the style in \LWR@temptwo.

```
13369 \NewDocumentCommand{\LWR@@makebox@paren}{m m}{%
13370 \IfValueTF{#2}{%
13371     \setlength{\LWR@tempwidth}{#1\unitlength}%
13372     \setlength{\LWR@tempheight}{#2\unitlength}%
13373     \appto{\LWR@temptwo}{%
13374         \LWR@print@mbox{width:\LWR@printlength{\LWR@tempwidth}} ; % space
13375         \LWR@print@mbox{height:\LWR@printlength{\LWR@tempheight}} ; % space
13376     }%
13377 }{%
13378     \PackageError{lwarp}%
13379         {(width,height) is missing a comma ',' character}%
13380         {\protect\makebox\space and \protect\framebox\space accept
13381             a size in the format (width,height).}%
13382 }%
13383 }
```

\LWR@@makebox@align {⟨*alignment character*⟩}

Adds to the style in \LWR@temptwo.

```
13384 \newcommand*{\LWR@@makebox@align}[1]{%
13385     \def\LWR@align{center}%
```

```
13386     \ifstrequal{#1}{l}{\def\LWR@align{left}}{}%
13387     \ifstrequal{#1}{r}{\def\LWR@align{right}}{}%
13388     \ifstrequal{#1}{s}{\def\LWR@align{justify}}{}%
13389     \appto\LWR@temptwo{%
13390         \LWR@print@mbox{text-align:\LWR@align} ; %
13391     }%
13392 }
```

\makebox (⟨*width,height*⟩) [⟨*width*⟩] [⟨*pos*⟩] {⟨*text*⟩}

```
13393 \NewDocumentCommand{\LWR@HTML@makebox}{>{\SplitArgument{1}{,}}d() o o +m}{%
```

Build the style depending on arguments:

```
13394     \begin{LWR@setvirtualpage}%
13395         \def\LWR@temptwo{}%
13396         \IfValueTF{#1}%
13397         {% (width,height) ..
13398             \LWR@@makebox@paren #1%
13399             \IfValueT{#2}%
13400             {% (width,height) [posn]
13401                 \LWR@@makebox@align{#2}%
13402             }%
13403         }%
13404         {% [width]
13405             \IfValueT{#2}% [width]
13406             {%
13407                 \setlength{\LWR@tempwidth}{#2}%
13408                 \ifdimgreater{\LWR@tempwidth}{0pt}{%
13409                     \appto\LWR@temptwo{%
13410                         width:\LWR@printlength{\LWR@tempwidth} ; % space
13411                     }%
13412                 }{}%
13413             }%
13414         }%
13415         \IfValueT{#3}%
13416         {% [width] [posn]
13417             \LWR@@makebox@align{#3}%
13418         }%
13419         \InlineClass[%
13420             \LWR@print@mbox{display:inline-block} ; %
13421             \LWR@temptwo%
13422         ]%
13423         {makebox}%
13424         {#4}%
13425     \end{LWR@setvirtualpage}%
13426 }
13427 \LWR@formatted{makebox}
```

\framebox (⟨*width,height*⟩) [⟨*width*⟩] [⟨*pos*⟩] {⟨*text*⟩}

```
13428 \NewDocumentCommand{\LWR@HTML@framebox}{d() o o +m}{%
13429     \fbox{\makebox(#1)[#2][#3]{#4}}%
13430 }
13431
13432 \LWR@formatted{framebox}
```

\LWR@forceminwidth {⟨*legth*⟩}

Sets \LWR@atleastonept to be at least 1pt.

```
13433 \newlength{\LWR@atleastonept}
13434
13435 \newcommand*{\LWR@forceminwidth}[1]{%
13436 \setlength{\LWR@atleastonept}{#1}%
13437 \ifthenelse{%
13438     \lengthtest{\LWR@atleastonept>0pt}\AND%
13439     \lengthtest{\LWR@atleastonept<1pt}%
13440 }%
13441    {\setlength{\LWR@atleastonept}{1pt}}%
13442    {}%
13443 }
```

\LWR@fboxstyle   Prints the HTML attributes for a black border and padding.

\LWR@forceminwidth must be used first in order to set the border width.

```
13444 \newcommand*{\LWR@fboxstyle}{%
13445     \LWR@findcurrenttextcolor%
13446   border:\LWR@printlength{\LWR@atleastonept} solid \LWR@origpound\LWR@tempcolor ; %
13447     padding:\LWR@printlength{\fboxsep} ; %
13448     color:\LWR@origpound\LWR@tempcolor%
13449 }
```

\fbox {⟨*text*⟩}

Creates a framed inline span enclosing the text.

Create a new HTML version, but don't use it until after xcolor may have loaded:

```
13450 \newcommand{\LWR@HTML@fbox}[1]{%
13451     \LWR@traceinfo{HTML fbox}%
13452     \LWR@forceminwidth{\fboxrule}%
13453     \LWR@traceinfo{HTML fbox B}%
13454     \InlineClass[%
13455         \LWR@print@mbox{display:inline-block} ; %
13456         \LWR@fboxstyle%
13457     ]{fbox}{#1}%
13458     \LWR@traceinfo{HTML fbox: done}%
13459 }
```

xcolor \lets things to \fbox when it is loaded, and this must remain even for HTML output while in a lateximage, so \fbox is not modified until \AtBeginDocument:

```
13460 \AtBeginDocument{\LWR@formatted{fbox}}
```

\fboxBlock {⟨*text*⟩} Creates a framed HTML <div> of the text.

First, a print-mode version. This is newly defined for print mode, so it is defined inside warpall.

```
13461 \end{warpHTML}
13462
13463 \begin{warpall}
13464 \let\fboxBlock\fbox
13465 \end{warpall}
13466
```

13467 \begin{warpHTML}

Next, an HTML version:

13468 \newcommand{\LWR@HTML@fboxBlock}[1]{%
13469 \LWR@forceminwidth{\fboxrule}%
13470 \LWR@stoppars%
13471 \begin{BlockClass}[\LWR@fboxstyle]{fboxBlock}
13472 #1
13473 \end{BlockClass}
13474 \LWR@startpars%
13475 }
13476
13477 \LWR@formatted{fboxBlock}
13478
13479 \end{warpHTML}

fminipage (*env.*) [⟨*align*⟩] [⟨*height*⟩] [⟨*align*⟩] {⟨*width*⟩}

Creates a framed HTML <div> around its contents.

Print version:

13480 \begin{warpall}
13481
13482 \newsavebox{\LWR@fminipagebox}
13483
13484 \NewDocumentEnvironment{fminipage}{O{t} o O{t} m}
13485 {%

An outer minipage will be used for vertical alignment. An inner minipage will be framed with \fbox.

If the optional inner alignment is not given, use the outer instead:

13486 \IfValueTF{#3}%
13487 {\def\LWR@thisalign{#3}}
13488 {\def\LWR@thisalign{#1}}%

Form the outer minipage depending on whether a height was given. Make the outer minipage larger to compensate for the frame.

13489 \IfValueTF{#2}%
13490 {\minipage[#1][#2+2\fboxsep+2\fboxrule][\LWR@thisalign]{#4+2\fboxsep+2\fboxrule}}%
13491 {\minipage[#1]{#4+2\fboxsep+2\fboxrule}}%

Capture the contents of the environment:

13492 \begin{lrbox}{\LWR@fminipagebox}%

Nest the contents inside an inner minipage of the desired size:

13493 \IfValueTF{#2}%
13494 {\minipage[#1][#2][\LWR@thisalign]{#4}}%
13495 {\minipage[#1]{#4}}%
13496 }
13497 {%

Close the inner minipage and the LR box with the contents:

```
13498 \endminipage%
13499 \end{lrbox}%
```

Create a frame around the contents of the environment:

```
13500 \fbox{\usebox{\LWR@fminipagebox}}%
```

The entire thing is placed inside the outer minipage:

```
13501 \endminipage%
13502 }
13503 \end{warpall}
```

HTML version:

```
13504 \begin{warpHTML}
13505
13506 \NewDocumentEnvironment{LWR@HTML@fminipage}{O{t} o O{t} m}
13507 {%
13508 \LWR@traceinfo{fminipage #1 #2 #3 #4}%
```

Locally change to the virtual page size before processing the requested sizes:

```
13509 \begin{LWR@setvirtualpage}*%
13510 \setlength{\LWR@tempwidth}{#4}%
13511 \IfValueT{#2}{\setlength{\LWR@tempheight}{#2}}%
```

Use a rule of at least one pixel in width:

```
13512 \LWR@forceminwidth{\fboxrule}%
```

```
13513 \LWR@stoppars%
```

```
13514 \begin{BlockClass}[%
13515 \LWR@fboxstyle ; %
13516 \IfValueT{#2}{height:\LWR@printlength{\LWR@tempheight} ; }%
13517 \ifbool{LWR@minipagefullwidth}%
13518 {\global\boolfalse{LWR@minipagefullwidth}}%
13519 {%
13520     \ifbool{LWR@forceminipagefullwidth}%
13521         {}%
13522         {%
13523             \ifdimequal{\LWR@tempwidth}{\linewidth}%
13524                 {}%
13525                 {width:\LWR@printlength{\LWR@tempwidth} ; }%
13526         }%
13527 }%
13528 ]{fminipage}%
13529 }
13530 {%
13531 \end{BlockClass}%
13532 \end{LWR@setvirtualpage}%
```

Prevent paragraph tags around horizontal white space until the start of the next paragraph:

```
13533 \global\booltrue{LWR@minipagethispar}%
13534 \LWR@traceinfo{fminipage done}%
```

```
13535 }
13536
13537 \LWR@formattedenv{fminipage}
```

\raisebox {⟨*raiselen*⟩} [⟨*height*⟩] [⟨*depth*⟩] {⟨*text*⟩}

```
13538 \NewDocumentCommand{\LWR@HTML@raisebox}{m o o m}{%
13539 #4%
13540 }
13541
13542 \LWR@formatted{raisebox}
```

```
13543 \end{warpHTML}
```

97 Direct formatting

⚠ \bfseries, etc. \textbf, etc. are supported, but \bfseries, etc. work only in some situations.

⚠ HTML special chars &, <, and > have special meanings in HTML. If \&, \textless, and \textgreater are used, proper HTML entities will be used, but there may be HTML parsing problems if these special characters occur unescaped in program listings or other verbatim text.

program listings For program listings, the listings package is supported, and its literate option is used to automatically convert &, <, and > to proper HTML entities.

minted sanitizes HTML automatically by its colorizing, which splits the special characters from the rest of the tag.

⚠ verbatim The fancyvrb and fvextra packages automatically sanitize HTML entities, but the core LaTeX verbatim-related environments do not, nor does the verbatim package, so care must be taken to avoid accidentally including valid HTML code inside these environments. It may be sufficient to add a space on either side of &, <, and >.

⚠ gobble fancyvrb does not sanitize HTML when using the gobble option.

For high-level block and inline custom CSS classes, see section 54.10.

for HTML & PRINT: 13544 \begin{warpall}

FixSmallCaps (*bool*) User may set FixSmallCaps to true if small caps are being incorrectly rendered as all caps.

```
13545 \newbool{FixSmallCaps}
13546 \boolfalse{FixSmallCaps}
```

```
13547 \end{warpall}
```

for HTML output: 13548 \begin{warpHTML}

\emph {⟨*text*⟩}

```
13549 \DeclareRobustCommand{\LWR@HTML@emph}[1]{%
13550     {%
```

```
13551          \LWR@HTML@itshape%
13552          \LWR@htmlspan{em}{#1}%
13553     }%
13554 }
13555
13556 \LWR@formatted{emph}
```

\textmd {⟨*text*⟩}

```
13557 \DeclareRobustCommand{\LWR@HTML@textmd}[1]{%
13558     {%
13559          \LWR@HTML@mdseries%
13560          \InlineClass(font-weight:normal){textmd}{#1}%
13561     }%
13562 }
13563
13564 \LWR@formatted{textmd}
```

\textbf {⟨*text*⟩}

```
13565 \DeclareRobustCommand{\LWR@HTML@textbf}[1]{%
13566     {%
13567          \LWR@HTML@bfseries%
13568          \LWR@htmlspan{b}{#1}%
13569     }%
13570 }
13571
13572 \LWR@formatted{textbf}
```

\texteb {⟨*text*⟩}   From nfssext-cfr.

```
13573 \AtBeginDocument{
13574 \IfPackageLoadedTF{nfssext-cfr}{
13575 \providerobustcmd\texteb[1]{}
13576 \DeclareRobustCommand{\LWR@HTML@texteb}[1]{%
13577     {%
13578          \LWR@HTML@ebweight%
13579          \InlineClass{texteb}{#1}%
13580     }%
13581 }
13582 \LWR@formatted{texteb}
13583 }
13584 {% if not loaded
13585     \providerobustcmd\texteb[1]{}
13586 }
13587 }
```

\textlg {⟨*text*⟩}   From nfssext-cfr.

```
13588 \AtBeginDocument{
13589 \IfPackageLoadedTF{nfssext-cfr}{
13590 \providerobustcmd\textlg[1]{}
13591 \DeclareRobustCommand{\LWR@HTML@textlg}[1]{%
13592     {%
13593          \LWR@HTML@lgweight%
13594          \InlineClass{textlg}{#1}%
13595     }%
```

```
13596 }
13597
13598 \LWR@formatted{textlg}
13599 }{% if not loaded
13600     \providerobustcmd{\textlg}[1]{}
13601 }
13602 }
```

\textrm {⟨*text*⟩}

```
13603 \DeclareRobustCommand{\LWR@HTML@textrm}[1]{%
13604     {%
13605         \LWR@HTML@rmfamily%
13606         \InlineClass(font-family:serif){textrm}{#1}%
13607     }%
13608 }
13609
13610 \LWR@formatted{textrm}
```

\textsf {⟨*text*⟩}

```
13611 \DeclareRobustCommand{\LWR@HTML@textsf}[1]{%
13612     {%
13613         \LWR@HTML@sffamily%
13614         \InlineClass(font-family:sans){textsf}{#1}%
13615     }%
13616 }
13617
13618 \LWR@formatted{textsf}
```

\texttt {⟨*text*⟩}

```
13619 \DeclareRobustCommand{\LWR@HTML@texttt}[1]{%
13620     {%
13621         \LWR@HTML@ttfamily%
13622         \LWR@htmlspan{kbd}{#1}%
13623     }%
13624 }
13625
13626 \LWR@formatted{texttt}
```

\textup {⟨*text*⟩}

```
13627 \DeclareRobustCommand{\LWR@HTML@textup}[1]{%
13628     {%
13629         \LWR@HTML@upshape%
13630         \InlineClass(font-style:normal){textup}{#1}%
13631     }%
13632 }
13633
13634 \LWR@formatted{textup}
```

\textit {⟨*text*⟩}

```
13635 \DeclareRobustCommand{\LWR@HTML@textit}[1]{%
13636     {%
```

```
13637          \LWR@HTML@itshape%
13638          \LWR@htmlspan{i}{#1}%
13639      }%
13640 }
13641
13642 \LWR@formatted{textit}
```

\textsc {⟨*text*⟩}

```
13643 \DeclareRobustCommand{\LWR@HTML@textsc}[1]{%
13644      {%
13645          \LWR@HTML@scshape%
13646          \InlineClass{textsc}{#1}%
13647      }%
13648 }
13649
13650 \LWR@formatted{textsc}
```

\textulc {⟨*text*⟩}   From fontaxes.

```
13651 \DeclareRobustCommand{\LWR@HTML@textulc}[1]{%
13652      {%
13653          \LWR@HTML@ulcshape%
13654          \InlineClass{textulc}{#1}%
13655      }%
13656 }
13657
13658 \LWR@formatted{textulc}
```

\textsi {⟨*text*⟩}

```
13659 \AtBeginDocument{
13660 \@ifundefined{textsi}{
13661      \LetLtxMacro\LWR@print@textsi\LWR@print@textsc
13662 }{}
13663
13664 \DeclareRobustCommand{\LWR@HTML@textsi}[1]{%
13665      {%
13666          \LWR@HTML@sishape%
13667          \textsc\textit{#1}}%
13668 %        \InlineClass(
13669 %            font-style: italic;
13670 %            font-variant: small-caps ;
13671 %            font-variant-numeric: oldstyle-nums ;
13672 %        ){textsi}{#1}%
13673      }%
13674 }
13675
13676 \LWR@formatted{textsi}
13677 }
```

\textsl {⟨*text*⟩}

```
13678 \DeclareRobustCommand{\LWR@HTML@textsl}[1]{%
13679      {%
13680          \slshape%
13681          \InlineClass(font-style:oblique){textsl}{#1}%
```

```
13682      }%
13683 }
13684
13685 \LWR@formatted{textsl}
```

\textssc {⟨*text*⟩}

```
13686 \newrobustcmd{\LWR@HTML@textssc}[1]{\textsc{#1}}
13687 \LWR@formatted{textssc}
```

\textnormal {⟨*text*⟩}

```
13688 \DeclareRobustCommand{\LWR@HTML@textnormal}[1]{%
13689      \LWR@HTML@mdseries%
13690      \LWR@HTML@rmfamily%
13691      \LWR@HTML@upshape%
13692      \LWR@HTML@ulcshape%
13693      \InlineClass(%
13694          font-weight: normal;
13695          font-family: serif;
13696          font-style: normal;
13697          font-variant: normal;
13698          font-variant-numeric: normal ;
13699      ){textnormal}{#1}%
13700 }
13701
13702 \LWR@formatted{textnormal}
```

```
13703 \FilenameNullify{%
13704      \LetLtxMacro\emph\@firstofone%
13705      \LetLtxMacro\textmd\@firstofone%
13706      \LetLtxMacro\textbf\@firstofone%
13707      \LetLtxMacro\texteb\@firstofone%
13708      \LetLtxMacro\textlg\@firstofone%
13709      \LetLtxMacro\textrm\@firstofone%
13710      \LetLtxMacro\textsf\@firstofone%
13711      \LetLtxMacro\texttt\@firstofone%
13712      \LetLtxMacro\textup\@firstofone%
13713      \LetLtxMacro\textit\@firstofone%
13714      \LetLtxMacro\textsc\@firstofone%
13715      \LetLtxMacro\textulc\@firstofone%
13716      \LetLtxMacro\textsi\@firstofone%
13717      \LetLtxMacro\textsl\@firstofone%
13718      \LetLtxMacro\textssc\@firstofone%
13719      \LetLtxMacro\textnormal\@firstofone%
13720 }
```

Remembers the current font family, series, and shape. fontaxes support is integrated here.

```
13721 \newcommand*{\LWR@f@family}{rm}
13722 \newcommand*{\LWR@f@series}{md}
13723 \newcommand*{\LWR@f@shape}{up}
13724 \newcommand*{\LWR@f@shapecaps}{ulc}
```

\LWR@textcurrentfont {⟨*text*⟩}

Prints the text with the current font choices. Avoids nesting repeated font selections.

```
13725 \newcounter{LWR@textcurrentfontdepth}
13726 \setcounter{LWR@textcurrentfontdepth}{0}
13727
13728 \newcommand*{\LWR@textcurrentfont}[1]{%
13729     \ifnumcomp{\value{LWR@textcurrentfontdepth}}{>}{0}%
13730         {%
13731             \addtocounter{LWR@textcurrentfontdepth}{1}%
13732             #1%
13733             \addtocounter{LWR@textcurrentfontdepth}{-1}%
13734         }%
13735         {%
13736             \addtocounter{LWR@textcurrentfontdepth}{1}%
13737             \ifboolexpr{%
13738                 test {\ifdefstring{\LWR@f@family}{rm}} and
13739                 test {\ifdefstring{\LWR@f@series}{md}} and
13740                 test {\ifdefstring{\LWR@f@shape}{up}} and
13741                 test {\ifdefstring{\LWR@f@shapecaps}{ulc}}
13742             }%
13743                 {\InlineClass{textnormal}{#1}}%
13744                 {%
13745                     \InlineClass{%
13746                             text\LWR@f@family\LWR@orignobreakspace{}%
13747                             text\LWR@f@series\LWR@orignobreakspace{}%
13748                             text\LWR@f@shape\LWR@orignobreakspace{}%
13749                             text\LWR@f@shapecaps%
13750                     }%
13751                     {#1}%
13752                 }%
13753             \addtocounter{LWR@textcurrentfontdepth}{-1}%
13754         }%
13755 }
```

LWR@blocktextcurrentfont (*env.*) Prints the contents with the current font choices.

```
13756 \newenvironment*{LWR@blocktextcurrentfont}{%
13757 \LWR@stoppars%
13758 \BlockClass{%
13759         text\LWR@f@family\LWR@orignobreakspace{}%
13760         text\LWR@f@series\LWR@orignobreakspace{}%
13761         text\LWR@f@shape\LWR@orignobreakspace{}%
13762         text\LWR@f@shapecaps%
13763     }%
13764 }{\endBlockClass\LWR@startpars}
```

\mdseries

```
13765 \newrobustcmd*{\LWR@HTML@mdseries}{%
13766     \LWR@print@mdseries%
13767     \renewcommand*{\LWR@f@series}{md}%
13768 }
13769 \LWR@formatted{mdseries}
```

\bfseries

```
13770 \newrobustcmd*{\LWR@HTML@bfseries}{%
13771     \LWR@print@bfseries%
13772     \renewcommand*{\LWR@f@series}{bf}%
13773 }
13774 \LWR@formatted{bfseries}
```

\ebweight From nfssext-cfr.

```
13775 \AtBeginDocument{
13776 \IfPackageLoadedTF{nfssext-cfr}{
13777 \providerobustcmd{\ebweight}{}
13778 \newrobustcmd*{\LWR@HTML@ebweight}{%
13779     \LWR@print@ebweight%
13780     \renewcommand*{\LWR@f@series}{eb}%
13781 }
13782 \LWR@formatted{ebweight}
13783 }{}
13784 }
```

\lgweight From nfssext-cfr.

```
13785 \AtBeginDocument{
13786 \IfPackageLoadedTF{nfssext-cfr}{
13787 \providerobustcmd{\lgweight}{}
13788 \newrobustcmd*{\LWR@HTML@lgweight}{%
13789     \LWR@print@lgweight%
13790     \renewcommand*{\LWR@f@series}{lg}%
13791 }
13792 \LWR@formatted{lgweight}
13793 }{}
13794 }
```

\rmfamily

```
13795 \newrobustcmd*{\LWR@HTML@rmfamily}{%
13796     \LWR@print@rmfamily%
13797     \renewcommand*{\LWR@f@family}{rm}%
13798 }
13799 \LWR@formatted{rmfamily}
```

\sffamily

```
13800 \newrobustcmd*{\LWR@HTML@sffamily}{%
13801     \LWR@print@sffamily%
13802     \renewcommand*{\LWR@f@family}{sf}%
13803 }
13804 \LWR@formatted{sffamily}
```

\ttfamily

```
13805 \newrobustcmd*{\LWR@HTML@ttfamily}{%
13806     \LWR@print@ttfamily%
13807     \renewcommand*{\LWR@f@family}{tt}%
13808 }
13809 \LWR@formatted{ttfamily}
```

The following use \AtBeginDocument due to the LaTeX core \reinstall@nfss@defs,
which redefines these \AtBeginDocument. See **texdoc source2e**.

\upshape

```
13810 \newrobustcmd*{\LWR@HTML@upshape}{%
13811     \LWR@print@upshape%
13812     \renewcommand*{\LWR@f@shape}{up}%
13813 }
13814 \AtBeginDocument{\LWR@formatted{upshape}}
```

\itshape

```
13815 \newrobustcmd*{\LWR@HTML@itshape}{%
13816     \LWR@print@itshape%
13817     \renewcommand*{\LWR@f@shape}{it}%
13818 }
13819 \AtBeginDocument{\LWR@formatted{itshape}}
```

\scshape Note: \LWR@print@scshape is not used here since some fonts, such as erewhon,
          copy/paste as all-caps.

```
13820 \newrobustcmd*{\LWR@HTML@scshape}{%
13821     \ifbool{FixSmallCaps}{}{%
13822         \LWR@print@scshape%
13823     }%
13824     \renewcommand*{\LWR@f@shapecaps}{sc}%
13825 }
13826 \AtBeginDocument{\LWR@formatted{scshape}}
```

\ulcshape From fontaxes.

```
13827 \@ifundefined{ulcshape}{
13828     \LetLtxMacro\ulcshape\upshape
13829 }{}
13830 \newrobustcmd*{\LWR@HTML@ulcshape}{%
13831     \LWR@print@ulcshape%
13832     \renewcommand*{\LWR@f@shapecaps}{ulc}%
13833 }
13834 \AtBeginDocument{\LWR@formatted{ulcshape}}
```

\sishape

```
13835 \@ifundefined{sishape}{
13836     \LetLtxMacro\sishape\scshape
13837 }{}
13838 \newrobustcmd*{\LWR@HTML@sishape}{%
13839     \ifbool{FixSmallCaps}{}{%
13840         \LWR@print@sishape%
13841     }%
13842     \renewcommand*{\LWR@f@shape}{it}
13843     \renewcommand*{\LWR@f@shapecaps}{sc}%
13844 }
13845 \AtBeginDocument{\LWR@formatted{sishape}}
```

\slshape

```
13846 \newrobustcmd*{\LWR@HTML@slshape}{%
13847     \LWR@print@slshape%
13848     \renewcommand*{\LWR@f@shape}{sl}%
13849 }
13850 \AtBeginDocument{\LWR@formatted{slshape}}
```

\sscshape

```
13851 \newrobustcmd{\LWR@HTML@sscshape}{\LWR@HTML@scshape}
13852 \AtBeginDocument{\LWR@formatted{sscshape}}
```

\normalfont

```
13853 \newrobustcmd*{\LWR@HTML@normalfont}{\rmfamily\mdseries\upshape\ulcshape}
13854 \LWR@formatted{normalfont}
```

```
13855 \FilenameNullify{%
13856     \LetLtxMacro\rmfamily\@empty%
13857     \LetLtxMacro\sffamily\@empty%
13858     \LetLtxMacro\ttfamily\@empty%
13859     \LetLtxMacro\bfseries\@empty%
13860     \LetLtxMacro\ebweight\@empty%
13861     \LetLtxMacro\lgweight\@empty%
13862     \LetLtxMacro\mdseries\@empty%
13863     \LetLtxMacro\upshape\@empty%
13864     \LetLtxMacro\slshape\@empty%
13865     \LetLtxMacro\sishape\@empty%
13866     \LetLtxMacro\scshape\@empty%
13867     \LetLtxMacro\itshape\@empty%
13868     \LetLtxMacro\ulcshape\@empty%
13869     \LetLtxMacro\sscshape\@empty%
13870     \LetLtxMacro\normalfont\@empty%
13871 }
```

\sp {⟨text⟩}

For siunitx-v2. Must work in math mode.

```
13872 \renewcommand{\sp}[1]{\text{<sup>#1</sup>}}{}}
```

\sb {⟨text⟩}

For siunitx-v2. Must work in math mode.

```
13873 \renewcommand{\sb}[1]{\text{<sub>#1</sub>}}{}}
```

\textsuperscript {⟨text⟩}

```
13874 \newrobustcmd{\LWR@HTML@textsuperscript}[1]{\LWR@htmlspan{sup}{#1}}
13875 \LWR@formatted{textsuperscript}
```

\@textsuperscript {⟨text⟩}

```
13876 \newcommand{\LWR@HTML@@textsuperscript}[1]{\LWR@htmlspan{sup}{#1}}
13877 \LWR@formatted{@textsuperscript}
```

\textsubscript {⟨*text*⟩}

```
13878    \newrobustcmd{\LWR@HTML@textsubscript}[1]{\LWR@htmlspan{sub}{#1}}
13879    \LWR@formatted{textsubscript}
```

\@textsubscript {⟨*text*⟩}

```
13880    \newcommand{\LWR@HTML@@textsubscript}[1]{\LWR@htmlspan{sub}{#1}}
13881    \LWR@formatted{@textsubscript}
```

\up {⟨*text*⟩} Prints superscript.

This is \let at the beginning of the document in case some other package has changed the definition.

```
13882 \AtBeginDocument{\let\up\textsuperscript}
```

\fup {⟨*text*⟩} Prints superscript.

Supports fmtcount package.

This is \let at the beginning of the document in case some other package has changed the definition.

```
13883 \AtBeginDocument{\let\fup\textsuperscript}
```

\underline {⟨*text*⟩}

```
13884 \renewcommand{\underline}[1]{%
13885    \InlineClass%
13886       (text-decoration:underline; text-decoration-skip: auto)%
13887       {underline}{#1}%
13888 }
```

\LWR@overline {⟨*text*⟩}

```
13889 \newcommand{\LWR@overline}[1]{%
13890    \InlineClass%
13891       (text-decoration:overline; text-decoration-skip: auto)%
13892       {overline}{#1}%
13893 }
```

\LWR@currenttextcolor The color to use for text and \rule, defaulting to black:

```
13894 \newcommand*{\LWR@currenttextcolor}{black}
```

\LWR@tempcolor The color converted to HTML colorspace.
\LWR@tempcolortwo
```
13895 \newcommand*{\LWR@tempcolor}{}
13896 \newcommand*{\LWR@tempcolortwo}{}
```

\LWR@findcurrenttextcolor Sets \LWR@tempcolor to the current color.

```
13897 \newcommand*{\LWR@findcurrenttextcolor}{%
13898     \renewcommand{\LWR@tempcolor}{000000}%
13899 }
```

\LWR@textcurrentcolor {⟨*text*⟩} Like \textcolor but uses the current \color instead.

```
13900 \NewDocumentCommand{\LWR@textcurrentcolor}{m}{%
13901     \renewcommand*{\LWR@currenttextcolor}{black}%
13902     #1%
13903 }
```

```
13904 \end{warpHTML}
```

for PRINT output:
```
13905 \begin{warpprint}
```

\LWR@textcurrentfont {⟨*text*⟩}

Prints the text with the current font choices.

```
13906 \newcommand*{\LWR@textcurrentfont}[1]{#1}
```

LWR@blocktextcurrentfont (*env.*) Prints the contents with the current font choices.

```
13907 \newenvironment*{LWR@blocktextcurrentfont}{}{}
```

\FilenameNullify {⟨*macros to nullify*⟩}

```
13908 \newcommand*{\FilenameNullify}[1]{}
```

```
13909 \end{warpprint}
```

98 Skips, spaces, font sizes

for HTML output:
```
13910 \begin{warpHTML}
```

LWR@HTMLsanitize@nobreakspace (*bool*)
Used to disable the nbsp entity inside verbatims <pre> sections, but not inside inline verbatims where spacing must be preserved by <nbsp>.

(fvextra used ~ which showed as <nbsp>.)

```
13911 \newbool{LWR@HTMLsanitize@nobreakspace}
13912 \booltrue{LWR@HTMLsanitize@nobreakspace}
```

\, and \thinspace may be redefined by other packages, so are redefined \AtBeginDocument here.

Direct-formatting space commands become HTML entities:

```
13913 \AtBeginDocument{%
13914 %
13915 \renewrobustcmd*{\,}{\HTMLUnicode{202f}}% HTML thin non-breakable space, not using LWR@formatted
13916 %
```

```
13917 \newrobustcmd*{\LWR@HTML@thinspace}{\HTMLunicode{202f}}% HTML thin non-breakable space
13918 \LWR@formatted{thinspace}
13919 %
13920 \newrobustcmd*{\LWR@HTML@negthinspace}{\HTMLunicode{202f}}   % HTML thin non-breakable space
13921 \LWR@formatted{negthinspace}
```

Cannot use \LWR@formatted for ~ or \nobreakspace.

```
13922 \renewrobustcmd*{~}{%
13923     \ifbool{LWR@HTMLsanitize@nobreakspace}%
13924         {\leavevmode\nobreak\HTMLentity{nbsp}}%
13925         {\LWR@orignobreakspace}%
13926 }
13927
13928 \LetLtxMacro\nobreakspace~
```

\?-\nobreakspace seems to be necessary for packages such as ctexbook, where this is used at the end of the document.

```
13929 \expandafter\LetLtxMacro\csname ?-\string\nobreakspace\endcsname~
```

```
13930 \newrobustcmd*{\LWR@HTML@textellipsis}{\HTMLunicode{2026}}
13931 \LWR@formatted{textellipsis}
13932 %
13933 \newrobustcmd*{\LWR@HTML@vdots}{\HTMLunicode{22EE}}
13934 \LWR@formatted{vdots}
13935 %
13936 }% AtBeginDocument
```

Direct-formatting font sizes are remembered for future use:

```
13937 \newcommand*{\LWR@font@size}{normalsize}
13938
13939 \newrobustcmd*{\LWR@HTML@normalsize}{\renewcommand*{\LWR@font@size}{normalsize}}
13940 \LWR@formatted{normalsize}
13941
13942 \newrobustcmd*{\LWR@HTML@small}{\renewcommand*{\LWR@font@size}{small}}
13943 \LWR@formatted{small}
13944
13945 \newrobustcmd*{\LWR@HTML@footnotesize}{\renewcommand*{\LWR@font@size}{footnotesize}}
13946 \LWR@formatted{footnotesize}
13947
13948 \newrobustcmd*{\LWR@HTML@scriptsize}{\renewcommand*{\LWR@font@size}{scriptsize}}
13949 \LWR@formatted{scriptsize}
13950
13951 \newrobustcmd*{\LWR@HTML@tiny}{\renewcommand*{\LWR@font@size}{tiny}}
13952 \LWR@formatted{tiny}
13953
13954 \newrobustcmd*{\LWR@HTML@large}{\renewcommand*{\LWR@font@size}{large}}
13955 \LWR@formatted{large}
13956
13957 \newrobustcmd*{\LWR@HTML@Large}{\renewcommand*{\LWR@font@size}{Large}}
13958 \LWR@formatted{Large}
13959
13960 \newrobustcmd*{\LWR@HTML@LARGE}{\renewcommand*{\LWR@font@size}{LARGE}}
13961 \LWR@formatted{LARGE}
13962
13963 \newrobustcmd*{\LWR@HTML@huge}{\renewcommand*{\LWR@font@size}{huge}}
13964 \LWR@formatted{huge}
```

```
13965
13966 \newrobustcmd*{\LWR@HTML@Huge}{\renewcommand*{\LWR@font@size}{Huge}}
13967 \LWR@formatted{Huge}


13968 \DeclareDocumentCommand{\onecolumn}{}{}
13969
13970 \DeclareDocumentCommand{\twocolumn}{O{}}{
13971
13972 #1
13973
13974 }
```

\hfill

```
13975 \newcommand*{\LWR@HTML@hfill}{\qquad}
13976 \LWR@formatted{hfill}
```

\hrulefill

```
13977 \newcommand*{\LWR@HTML@hrulefill}{%
13978     \ifbool{LWR@doingapar}%
13979         {\rule{1in}{1pt}}%
13980         {%
13981             \LWR@findcurrenttextcolor%
13982             \ifdefstring{\LWR@tempcolor}{000000}%
13983         {%
13984             \begin{BlockClass}{hrule}%
13985             \end{BlockClass}%
13986         }%
13987         {%
13988             \begin{BlockClass}[%
13989               border-top: 1px solid \LWR@origpound\LWR@tempcolor % space
13990             ]{hrule}%
13991             \end{BlockClass}%
13992         }%
13993       }%
13994 }%
13995 \LWR@formatted{hrulefill}
```

\dotfill

```
13996 \newcommand*{\LWR@HTML@dotfill}{\dots}
13997 \LWR@formatted{dotfill}
```

\newpage Not \LWR@foramtted since cannot be used inside a lateximage anyhow.

```
13998 \renewcommand*{\newpage}{
13999
14000 }
```

\newline Uses the HTML <br /> element.

```
14001 \newrobustcmd*{\LWR@newlinebr}{\unskip\LWR@htmltag{br /}\LWR@orignewline}%
14002 \LetLtxMacro\newline\LWR@newlinebr
```

\\ Redefined to \LWR@endofline or \LWR@tabularendofline.

\LWR@endofline * [⟨*len*⟩]

>\\ is assigned to \LWR@endofline at \LWR@LwarpStart.

>Inside tabular, \\ is temporarily changed to \LWR@tabularendofline.

```
14003 \LetLtxMacro\LWR@origendofline\\
14004 \NewDocumentCommand{\LWR@endofline}{s O{0pt}}
14005 {%
14006 \newline%

14007 \setlength{\LWR@templengthone}{#2}%
14008 \ifdimgreater{\LWR@templengthone}{0pt}{\newline}{}%
14009 }
```

\LWR@minipagestartpars Minipages are often placed side-by-side inside figures, with a bit of horizontal space to separate them. Since HTML does not allow a <div> to be inside a p, paragraphs must be turned off during the generation of the minipage, then turned on after the minipage is complete. When this occurs between side-by-side minipages, lwarp correctly suppresses the paragraph tags between the minipages, unless some other text is between the minipages. Such text forms its own paragraph, resulting in text after a minipage to be on its own line. Since people often place

\hspace small horizontal space between minipages, it is desirable to maintain this space if

\enskip possible. lwarp tries to do this by remembering that a minipage has been seen, in

\quad which case paragraph tags are suppressed around \hspace, \enskip, \quad, and

\qquad \qquad until the end of the paragraph, when the closing p tag is created.

>When a minipage is seen, the boolean LWR@minipagethispar is set, telling the following horizontal whitespace commands to try to suppress their surrounding paragraph tags. LWR@minipagethispar is cleared at the next end of paragraph, when the HTML paragraph closing tag is generated.

>Placed just before \hspace, \quad, or \qquad's HTML output.

```
14010 \newcommand*{\LWR@minipagestartpars}{%
14011     \ifbool{LWR@minipagethispar}{\LWR@startpars}{}%
14012 }
```

\LWR@minipagestoppars Placed just after \hspace, \quad, or \qquad's HTML output.

```
14013 \newcommand*{\LWR@minipagestoppars}{%
14014     \ifbool{LWR@minipagethispar}{\LWR@stoppars}{}%
14015 }
```

\quad Handles special minipage & horizontal space interactions. Uses 2003 EM SPACE to pass validation.

```
14016 \newrobustcmd*{\LWR@HTML@quad}{%
14017     \LWR@minipagestoppars%
14018     \HTMLunicode{2003}%
14019     \LWR@minipagestartpars%
14020 }
14021 \LWR@formatted{quad}
```

\qquad Handles special minipage & horizontal space interactions.

```
14022 \newrobustcmd*{\LWR@HTML@qquad}{\quad\quad}
14023 \LWR@formatted{qquad}
```

\enskip Handles special minipage & horizontal space interactions.

```
14024 \newrobustcmd*{\LWR@HTML@enskip}{%
14025     \LWR@minipagestoppars%
14026     \HTMLunicode{2002}%
14027     \LWR@minipagestartpars%
14028 }
14029 \LWR@formatted{enskip}
```

\LWR@tempwidth (*Len*) Used to compute span width, height, raise for \hspace and \rule:

\LWR@tempheight (*Len*)
```
14030 \newlength{\LWR@tempwidth}
```
\LWR@tempraise (*Len*)
```
14031 \newlength{\LWR@tempheight}
14032 \newlength{\LWR@tempraise}
```

\hspace * {⟨*length*⟩} * {⟨*length*⟩}

Handles special minipage & horizontal space interactions.

Prints a span of a given width. Ignores the optional star.

\hspace{\fill} is converted to \hspace{2em}, equal to \qquad.

```
14033 \NewDocumentCommand{\LWR@HTML@hspace}{s m}{%
14034 \setlength{\LWR@tempwidth}{#2}%
```

If \fill, change to \qquad:

```
14035 \ifnum\gluestretchorder\LWR@tempwidth>0%
14036 \setlength{\LWR@tempwidth}{2em}%
14037 \fi%
```

Only if the width is greater than zero:

```
14038 \ifdimcomp{\LWR@tempwidth}{>}{0pt}{%
```

If had a minipage this paragraph, try to inline the white space without generating paragraph tags:

```
14039     \LWR@minipagestoppars%
```

Support the HTML thin wrappable space:

```
14040     \ifdimcomp{\LWR@tempwidth}{=}{.16667em}%
14041     {%
14042         \HTMLunicode{2009}% thin breakable space
14043     }%
```

Print the span with the converted width. Not rounded.

```
14044     {%
14045         \LWR@htmltagc{%
14046         span style=\textquotedbl{}width:\LWR@printlength{\LWR@tempwidth}; % extra space
14047             display:inline-block\textquotedbl%
14048         }%
```

If formatting for a word processor, approximate with a number of \quads, in case a span of a given width is not supported:

```
14049        \ifbool{FormatWP}{%
14050            \setlength{\LWR@templengthone}{\LWR@tempwidth}%
14051            \whiledo{\lengthtest{\LWR@templengthone>1em}}{%
14052                \quad%
14053                \addtolength{\LWR@templengthone}{-1em}%
14054            }%
14055        }%
```

If NOT formatting for a word processor, include an empty comment to avoid an empty span:

```
14056        {\LWR@htmlcomment{}}%
```

Close the span:

```
14057            \LWR@htmltagc{/span}%
14058        }%
```

If had a minipage this paragraph, try to inline the white space without generating paragraph tags:

```
14059    \LWR@minipagestartpars%
14060 }{}% width greater than 0
14061 }%
14062 \LWR@formatted{hspace}
```

\LWR@vspace * {⟨*length*⟩} Nullified vspace.

```
14063 \NewDocumentCommand{\LWR@HTML@vspace}{s m}{}
14064
14065 \LWR@formatted{vspace}
```

\linebreak [⟨*num*⟩]   Inserts an HTML br tag.

```
14066 \renewcommand*{\linebreak}[1][]{\newline}
```

\nolinebreak [⟨*num*⟩]

```
14067 \renewcommand*{\nolinebreak}[1][]{}
```

\pagebreak [⟨*num*⟩]   Starts a new paragraph.

```
14068 \renewcommand*{\pagebreak}[1][]{
14069
14070 }
```

\nopagebreak [⟨*num*⟩]

```
14071 \renewcommand*{\nopagebreak}[1][]{}
```

\enlargethispage * {⟨*len*⟩}

```
14072 \RenewDocumentCommand{\enlargethispage}{s m}{}
```

\clearpage
\cleardoublepage

```
14073 \renewcommand*{\clearpage}{}
14074 \renewcommand*{\cleardoublepage}{}
```

\rule [⟨*raise*⟩] {⟨*width*⟩} {⟨*height*⟩}

Handles special minipage & horizontal space interactions.

Creates a span of a given width and height. Ignores the optional star.

\fill is zero-width, so \hspace{\fill} is ignored.

```
14075 \newcommand*{\LWR@HTML@rule}[3][]{%
```

The width is copied into a temporary LATEX length, from which comparisons and conversions may be made:

```
14076 \setlength{\LWR@tempwidth}{#2}%
```

If it's zero-width then skip the entire rule:

```
14077 \ifthenelse{\lengthtest{\LWR@tempwidth=0pt}}%
14078 {}% zero- width
14079 {% non-zero width
```

If it's non-zero width, set a minimal thickness so that it more reliably shows in the browser:

```
14080     \ifthenelse{%
14081         \lengthtest{\LWR@tempwidth>0pt}\AND%
14082         \lengthtest{\LWR@tempwidth<1pt}%
14083     }%
14084         {\setlength{\LWR@tempwidth}{1pt}}%
14085         {}%
```

Likewise with height:

```
14086     \setlength{\LWR@tempheight}{#3}%
14087     \ifthenelse{%
14088         \lengthtest{\LWR@tempheight>0pt}\AND%
14089         \lengthtest{\LWR@tempheight<1pt}%
14090     }%
14091         {\setlength{\LWR@tempheight}{1pt}}%
14092         {}%
```

If had a minipage this paragraph, try to inline the rule without generating paragraph tags:

```
14093     \LWR@minipagestoppars%
```

Print the span with the converted width and height. The width and height are NOT rounded, since a height of less than 1pt is quite common in LATEX code.

```
14094     \LWR@findcurrenttextcolor%
14095     \LWR@htmltagc{%
14096     span\LWR@indentHTML%
14097     style=\textquotedbl%
```

The HTML background color is used to draw the filled rule according to the LATEX foreground color set by \textcolor.

```
14098        \ifbool{FormatWP}{}{background:\LWR@currenttextcolor ; }%
```

The width and height are printed, converted to PT:

```
14099        width:\LWR@printlength{\LWR@tempwidth} ; %
14100        height:\LWR@printlength{\LWR@tempheight} ; %
```

The raise height is converted to a CSS transform. The *2 raise multiplier is to approximately match HTML output's X height. Conversion to a LATEX length allows a typical LATEX expression to be used as an argument for the raise, whereas printing the raise argument directly to HTML output without conversion to a LATEX length limits the allowable syntax. To do: A superior method would compute a ratio of LATEX ex height, then print that to HTML with an ex unit.

```
14101        \ifblank{#1}%
14102        {}%
14103        {%
14104            \setlength{\LWR@tempraise}{0pt-#1}%
14105            \setlength{\LWR@tempraise}{\LWR@tempraise*2}%
14106            \LWR@indentHTML%
14107            -ms-transform: translate(0pt,\LWR@printlength{\LWR@tempraise}); %
14108            \LWR@indentHTML%
14109          -webkit-transform: translate(0pt,\LWR@printlength{\LWR@tempraise}); %
14110            \LWR@indentHTML%
14111            transform: translate(0pt,\LWR@printlength{\LWR@tempraise}); %
14112            \LWR@indentHTML%
14113        }%
```

Display inline-block to place the span inline with the text:

```
14114        display:inline-block;\textquotedbl\LWR@orignewline%
14115        }%
```

If formatting for a word processor, approximate with a number of underscores, in case a span of a given width is not supported:

```
14116        \ifbool{FormatWP}{%
14117            \setlength{\LWR@templengthone}{\LWR@tempwidth}%
14118            \whiledo{\lengthtest{\LWR@templengthone>1em}}{%
14119                \_{}%
14120                \addtolength{\LWR@templengthone}{-1em}%
14121            }%
14122        }%
```

If NOT formatting for a word processor, add a comment to avoid an empty <span>:

```
14123            {\LWR@htmlcomment{}}%
```

Close the span:

```
14124        \LWR@htmltagc{/span}%
```

If had a minipage this paragraph, try to inline the white space without generating paragraph tags:

```
14125     \LWR@minipagestartpars%
14126 }% non-zero width
14127 }
14128
14129 \LWR@formatted{rule}
```

```
14130 \end{warpHTML}
```

99 \phantomsection

`14131 \begin{warpHTML}`

\LWR@phantomsection Emulate the hyperref \phantomsection command, often used to insert the bibliography into the table of contents. Ignores \ForceHTMLTOC.

```
14132 \newrobustcmd*{\LWR@phantomsection}{%
14133     \begingroup%
14134     \boolfalse{LWR@forcinghtmltoc}%
14135     \section*{}%
14136     \endgroup%
14137 }
```

```
14138 \end{warpHTML}
```

100   \LaTeX and other logos

Logos for HTML and print modes:

Some of these logos may be redefined in a later package, so after loading other packages, and at the beginning of the document, their definitions are finally set by \LWR@formatted.

For CSS conversions, see:
http://edward.oconnor.cx/2007/08/tex-poshlet
http://nitens.org/taraborelli/texlogo
and the spacing described in the metafont package documentation.

```
14139 \begin{warpall}
14140 \newbool{LWR@warnXe}
14141 \boolfalse{LWR@warnXe}
14142
14143 \newrobustcmd*{\Xe}
14144     {%
14145         X\hspace{-.1667em}\raisebox{-.5ex}{E}%
14146         \global\booltrue{LWR@warnXe}%
14147     }
14148
14149 \AtBeginDocument{
14150     \IfPackageLoadedTF{graphics}{
14151         \IfPackageLoadedTF{metalogo}{}{
```

```
14152              \renewrobustcmd*{\Xe}
14153                  {X\hspace{-.1667em}\raisebox{-.5ex}{\reflectbox{E}}}
14154          }
14155      }{}
14156 }
14157
14158 \AtEndDocument{
14159      \ifbool{LWR@warnXe}{
14160          \PackageNoteNoLine{lwarp}{Load graphicx or graphics
14161              for improved XeTeX logo}
14162      }{}
14163 }
14164
14165 \providerobustcmd*{\XeTeX}{\mbox{\Xe\hspace{-.125em}\TeX}}
14166 \providerobustcmd*{\XeLaTeX}{\mbox{\Xe\hspace{-.125em}\LaTeX}}
14167 \providerobustcmd*{\AmS}{%
14168      \leavevmode\hbox{$\mathcal A\kern-.2em\lower.376ex%
14169      \hbox{$\mathcal M$}\kern-.2em\mathcal S$}%
14170 }
14171 \newrobustcmd*{\LyX}{\textsf{LyX}}
14172 \providerobustcmd*{\LuaTeX}{\mbox{Lua\TeX}}
14173 \providerobustcmd*{\LuaLaTeX}{\mbox{Lua\LaTeX}}
14174 \providerobustcmd*{\BibTeX}{\mbox{B\textsc{ib}\TeX}}
14175 \providerobustcmd*{\MakeIndex}{\mbox{\textit{MakeIndex}}}
14176 \providerobustcmd*{\ConTeXt}{\mbox{Con\TeX{}t}}
14177 \providerobustcmd*{\MiKTeX}{\mbox{MiK\TeX}}
14178 \end{warpall}
```

`14179 \begin{warpHTML}`

The print-mode versions of the following may be changed by metalogo, so their print formatting is recorded \AtBeginDocument.

\TeX TEX

latexlogo is a css class used to properly typeset the E and A in LATEX and friends.

latexlogofont is a css class used to select the font for the rest of the logo in LATEX, LuaTEX, ConTEXt, etc.

```
14180 \newrobustcmd*{\LWR@HTML@TeX}
14181 {%
14182      \InlineClass{latexlogofont}%
14183      {%
14184          \InlineClass{latexlogo}%
14185          {%
14186              T%
14187              \InlineClass{latexlogosub}{e}%
14188              X%
14189          }%
14190      }%
14191 }
14192 \AtBeginDocument{\LWR@formatted{TeX}}% may have been patched by metalogo
```

\LaTeX LATEX, LATEX 2ε
\LaTeXe
```
14193 \newrobustcmd*{\LWR@HTML@LaTeX}
14194 {%
```

```
14195        \InlineClass{latexlogofont}%
14196        {%
14197            \InlineClass{latexlogo}%
14198            {%
14199                L%
14200                \InlineClass{latexlogosup}{a}%
14201                T%
14202                \InlineClass{latexlogosub}{e}%
14203                X%
14204            }%
14205        }%
14206 }
14207
14208 \AtBeginDocument{\LWR@formatted{LaTeX}}% may have been patched by metalogo
14209
14210
14211 \newrobustcmd*{\LWR@HTML@LaTeXe}
14212 {%
14213        \LaTeX%
14214        \InlineClass{latexlogofont}{%
14215            \InlineClass{latexlogotwoe}{%
14216                2%
14217                \InlineClass{latexlogotwoesub}{\HTMLunicode{03B5}}%
14218            }%
14219        }%
14220 }
14221 \AtBeginDocument{\LWR@formatted{LaTeXe}}% may have been patched by metalogo
```

\LuaTeX LuaTeX, LuaLaTeX

\LuaLaTeX

```
14222 \newrobustcmd*{\LWR@HTML@LuaTeX}{\InlineClass{latexlogofont}{Lua}\TeX}
14223 \AtBeginDocument{\LWR@formatted{LuaTeX}}% may have been patched by metalogo
14224
14225 \newrobustcmd*{\LWR@HTML@LuaLaTeX}{\InlineClass{latexlogofont}{Lua}\LaTeX}
14226 \AtBeginDocument{\LWR@formatted{LuaLaTeX}}% may have been patched by metalogo
```

\XeTeX XƎTEX, XƎLATEX

\XeLaTeX

xetexlogo is a css class which aligns the backwards E in XƎTEX and spaces TEX appropriately.

xelatexlogo is a css class which aligns the backwards E in XƎLATEX and spaces LATEX appropriately.

```
14227 \newrobustcmd*{\LWR@HTML@Xe}
14228        {%
14229            X%
14230            \InlineClass{xelatexlogosub}{\HTMLunicode{18e}}%
14231        }
14232 \AtBeginDocument{\LWR@formatted{Xe}}% may have been patched by metalogo
14233
14234 \newrobustcmd*{\LWR@HTML@XeTeX}{\InlineClass{xelatexlogo}{\Xe}\TeX}
14235 \AtBeginDocument{\LWR@formatted{XeTeX}}% may have been patched by metalogo
14236
14237 \newrobustcmd*{\LWR@HTML@XeLaTeX}{\InlineClass{xelatexlogo}{\Xe}\LaTeX}
14238 \AtBeginDocument{\LWR@formatted{XeLaTeX}}% may have been patched by metalogo
```

`\ConTeXt` ConTEXt

```
14239 \newrobustcmd*{\LWR@HTML@ConTeXt}{%
14240     \InlineClass{latexlogofont}{Con}\TeX{}%
14241     \InlineClass{latexlogofont}{t}%
14242 }
14243 \LWR@formatted{ConTeXt}
```

`\BibTeX` BIBTEX, *MakeIndex*
`\MakeIndex`

```
14244 \newrobustcmd*{\LWR@HTML@BibTeX}
14245     {\InlineClass{latexlogofont}{B\textsc{ib}}\TeX}
14246 \LWR@formatted{BibTeX}
14247
14248 \newrobustcmd*{\LWR@HTML@MakeIndex}
14249     {\InlineClass{latexlogofont}{\textit{MakeIndex}}}
14250 \LWR@formatted{MakeIndex}
```

`\AmS` 𝒜ℳ𝒮

amslogo is a css class used for the 𝒜ℳ𝒮 logo.

```
14251 \AtBeginDocument{%
14252 \newrobustcmd*{\LWR@HTML@AmS}
14253 {%
14254     \InlineClass{amslogo}{%
14255         \textit{%
14256             A%
14257             \InlineClass{latexlogosub}{M}%
14258             S%
14259         }%
14260     }%
14261 }%
14262 \LWR@formatted{AmS}
14263 }
```

`\MiKTeX` MiKTEX

```
14264 \newrobustcmd*{\LWR@HTML@MiKTeX}{\InlineClass{latexlogofont}{MiK}\TeX}
14265 \LWR@formatted{MiKTeX}
```

`\LyX` LyX

lyxlogo is a css class used for the LyX logo.

```
14266 \newrobustcmd*{\LWR@HTML@LyX}{\InlineClass{lyxlogo}{LyX}}
14267 \LWR@formatted{LyX}
```

```
14268 \end{warpHTML}
```

101   Starting and stopping lwarp

`14269 \begin{warpHTML}`

`\LWR@LwarpStart` Automatically sets up the HTML-related actions for the start and end of the docu-
`\LWR@LwarpEnd` ment.

```
14270 \AfterEndPreamble{\LWR@LwarpStart}
14271 \AtEndDocument{\LWR@LwarpEnd}
14272 \DeclareHookRule{enddocument}{lwarp}{after}{legacy}
```

```
14273 \end{warpHTML}
```

102 Loading array

array is required for lwarp's column parsing. It and its patches are now loaded.

for HTML output:
```
14274 \begin{warpHTML}
14275 \RequirePackage{array}
```

The following are compared with the tabular preamble > to add CSS classes to
adjust tabular cells. Defined here now that \arraybackslash is defined after array
is loaded.

```
14276 \edef\LWR@detect@centeringarraybackslash{\centering\arraybackslash}
14277 \edef\LWR@detect@raggedrightarraybackslash{\raggedright\arraybackslash}
14278 \edef\LWR@detect@raggedleftarraybackslash{\raggedleft\arraybackslash}
14279 \def\LWR@detect@itshape{\itshape}
14280 \def\LWR@detect@bfseries{\bfseries}
14281 \def\LWR@detect@bfit{\bfseries\itshape}
14282 \end{warpHTML}
```

103 Loading everyshi patches

everyshi is emulated by the LaTeX core, so its patches are loaded here. \AtBeginDocument
is used in case an older verison of LaTeX is used.

for HTML output:
```
14283 \begin{warpHTML}
14284 \AtBeginDocument{
14285     \IfPackageLoadedTF{everyshi}{
14286         \RequirePackage{lwarp-everyshi}
14287     }{}
14288 }
14289 \end{warpHTML}
```

104 Loading textcomp patches

textcomp has now been integrated into the LaTeX core, so its patches are loaded
now.

for HTML output:
```
14290 \begin{warpHTML}
14291 \RequirePackage{lwarp-textcomp}
14292 \end{warpHTML}
```

105   Loading amsmath, amsthm patches, centernot

amsmath, amsthm, and centernot may have been preloaded, such as by newtx, so their patches are loaded now.

14293 \begin{warpHTML}
14294 \IfPackageLoadedTF{amsthm}{
14295   \RequirePackage{lwarp-amsthm}
14296 }{}

14297 \IfPackageLoadedTF{amsmath}{
14298   \RequirePackage{lwarp-amsmath}
14299 }{}

amsthm may load centernot, so centernot must be checked second.

14300 \IfPackageLoadedTF{centernot}{
14301   \RequirePackage{lwarp-centernot}
14302 }{}
14303 \end{warpHTML}

106   Loading KOMA-SCRIPT class patches

Load patches to koma-script.

14304 \begin{warpHTML}

14305 \IfClassLoadedTF{scrbook}{\RequirePackage{lwarp-patch-komascript}}{}
14306 \IfClassLoadedTF{scrartcl}{\RequirePackage{lwarp-patch-komascript}}{}
14307 \IfClassLoadedTF{scrreprt}{\RequirePackage{lwarp-patch-komascript}}{}

14308 \end{warpHTML}

107   Loading MEMOIR class patches

Load patches to memoir.

14309 \begin{warpprint}
14310 \IfClassLoadedTF{memoir}{\LWR@origRequirePackage{xcolor}}{}
14311 \end{warpprint}

14312 \begin{warpHTML}
14313 \IfClassLoadedTF{memoir}{\RequirePackage{lwarp-patch-memoir}}{}
14314 \end{warpHTML}

108   ut* class patches

Load patches to uj* and ut* classes, as well as ltj* classes.

14315 \begin{warpHTML}

14316 `\newcommand*{\LWR@patchujtclasses}{`

uj/t does not use `\partname`

14317   `\def\@partnameformat{}`

14318   `\def\@partcntformat##1{%`
14319   `\prepartname%`
14320   `\csname the##1\endcsname%`
14321   `\postpartname%`
14322   `\quad%`
14323   `}`
14324   `\@ifundefined{chapter}{}{`
14325   `\def\@chapcntformat##1{%`
14326   `\prechaptername%`
14327   `\csname the##1\endcsname%`
14328   `\postchaptername%`
14329   `\quad%`
14330   `}`
14331   `}`
14332   `\renewcommand*{\LWR@printchaptername}{}`

Use decimal points instead of centered dots:

14333   `\renewcommand{\thepart}{\@Roman\c@part}`
14334   `\@ifundefined{chapter}{`
14335   `\renewcommand{\thesection}{\@arabic\c@section}`
14336   `}{`
14337   `\renewcommand{\thechapter}{\@arabic\c@chapter}`
14338   `\renewcommand{\thesection}{\thechapter.\@arabic\c@section}`
14339   `}`
14340   `\renewcommand{\thesubsection}{\thesection.\@arabic\c@subsection}`
14341   `\renewcommand{\thesubsubsection}{%`
14342 `\thesubsection.\@arabic\c@subsubsection}`
14343   `\renewcommand{\theparagraph}{%`
14344 `\thesubsubsection.\@arabic\c@paragraph}`
14345   `\renewcommand{\thesubparagraph}{%`
14346 `\theparagraph.\@arabic\c@subparagraph}`
14347   `\@ifundefined{chapter}{`
14348   `\renewcommand{\thefigure}{\@arabic\c@figure}`
14349   `\renewcommand{\thetable}{\@arabic\c@table}`
14350   `}{`
14351   `\renewcommand{\thefigure}{%`
14352 `\ifnum\c@chapter>\z@\thechapter.\fi\@arabic\c@figure}`
14353   `\renewcommand{\thetable}{%`
14354 `\ifnum\c@chapter>\z@\thechapter.\fi\@arabic\c@table}`
14355   `}`
14356 `}`
14357
14358 `\IfClassLoadedTF{ujarticle}{\LWR@patchujtclasses}{}`
14359 `\IfClassLoadedTF{ujbook}{\LWR@patchujtclasses}{}`
14360 `\IfClassLoadedTF{ujreport}{\LWR@patchujtclasses}{}`
14361 `\IfClassLoadedTF{utarticle}{\LWR@patchujtclasses}{}`
14362 `\IfClassLoadedTF{utbook}{\LWR@patchujtclasses}{}`
14363 `\IfClassLoadedTF{utreport}{\LWR@patchujtclasses}{}`
14364 `\IfClassLoadedTF{ltjarticle}{\LWR@patchujtclasses}{}`
14365 `\IfClassLoadedTF{ltjbook}{\LWR@patchujtclasses}{}`
14366 `\IfClassLoadedTF{ltjreport}{\LWR@patchujtclasses}{}`
14367 `\IfClassLoadedTF{ltjsarticle}{\LWR@patchujtclasses}{}`

```
14368 \IfClassLoadedTF{ltjsbook}{\LWR@patchujtclasses}{}
14369 \IfClassLoadedTF{ltjsreport}{\LWR@patchujtclasses}{}
14370 \IfClassLoadedTF{ltjskiyou}{\LWR@patchujtclasses}{}
14371 \IfClassLoadedTF{ltjspf}{\LWR@patchujtclasses}{}
14372 \IfClassLoadedTF{ltjtarticle}{\LWR@patchujtclasses}{}
14373 \IfClassLoadedTF{ltjtbook}{\LWR@patchujtclasses}{}
14374 \IfClassLoadedTF{ltjtreport}{\LWR@patchujtclasses}{}
```

```
14375 \end{warpHTML}
```

109   CTEX patches

Patches for ctex and related classes, which are loaded before lwarp.

All CTEX classes and the ctex package seem to load ctexpatch, so its presence is used to decide whether to have lwarp patch CTEX.

for HTML output: `14376 \begin{warpHTML}`

\AtBeginDocument in case the user set FileSectionNames in the preamble.

```
14377 \AtBeginDocument{
14378     \IfPackageLoadedTF{ctexpatch}{%
14379         \def\@partcntformat#1{%
14380             \LWR@isolate{\CTEX@partname}~%
14381             \CTEX@part@aftername%
14382         }%
14383
14384         \def\@partnameformat{}
14385
14386         \def\@chapcntformat#1{%
14387             \LWR@isolate{\CTEX@chaptername}~%
14388             \CTEX@chapter@aftername%
14389         }%
14390
14391         \renewcommand*{\LWR@printchaptername}{}
14392     }{}
14393 }
```

```
14394 \end{warpHTML}
```

110   kotexutf patches

Patch for kotexutf, which is loaded before lwarp.

kotexutf's \@setref was conflicting with lwarp's cross references.

for HTML output: `14395 \begin{warpHTML}`

If kotexutf's version of \@setref is detected, it is reverted to the original.

```
14396 \AtBeginDocument{
14397 \IfPackageLoadedTF{kotexutf}{%
```

```
14398        \def\LWR@kotexutf@setref#1#2#3{%
14399            \@setref@dhucs@orig{#1}{#2}{#3}%
14400          \ifx#1\relax\else
14401            \bgroup
14402            \dhucs@make@cjkchar@null
14403            \edef\@temp{\expandafter#2#1}\global\josatoks\expandafter{\@temp}%
14404            \egroup
14405          \fi%
14406        }%
14407
14408        \ifdefequal{\@setref}{\LWR@kotexutf@setref}{
14409            \let\@setref\@setref@dhucs@orig
14410        }{}
14411 }{}
14412 }
```

```
14413 \end{warpHTML}
```

111   babel and polyglossia warnings

lwarp prints a message instructing the user how to avoid the following error.

(These are not \PackageWarnings because there may not be a problem.)

lwarp uses cleveref, which has some limitations when using polyglossia, possibly resulting in the error

> ! Undefined control sequence. . . . __hook begindocument

To test compatibility, add

> \usepackage{cleveref}

near the end of the preamble (as the last package to be loaded), and try to compile the print version. It may be necessary to set

> \setdefaultlanguage{english}

or some other language supported by cleveref, then select other languages using \setotherlanguages.

Once the print version works with cleveref and polyglossia, the HTML version should work as well using lwarp.

for HTML output:
```
14414 \begin{warpHTML}
14415 \AtBeginDocument{
14416
14417 \IfPackageLoadedTF{polyglossia}{
14418     \PackageNoteNoLine{lwarp}
14419     {%
14420         Polyglossia has been loaded.  Lwarp also uses cleveref.\MessageBreak
14421         See the cleveref documentation regarding\MessageBreak
14422         polyglossia support.  Some languages are not supported.\MessageBreak
14423         --- \MessageBreak
14424         If the error\MessageBreak
14425         \space\space Undefined control sequence ...
14426         \protect\__hook begindocument\MessageBreak
14427         occurs here, use the polyglossia macro:\MessageBreak
```

```
14428          \space\space\protect\setmainlanguage\protect{...\protect}
14429     }
14430 }{
14431     \IfPackageLoadedTF{babel}{
14432        \PackageNoteNoLine{lwarp}
14433        {%
14434           Babel has been loaded.  Lwarp also uses cleveref.\MessageBreak
14435           See the cleveref documentation regarding\MessageBreak
14436           babel support.  Some languages are not supported%
14437        }
14438     }{}
14439 }
14440
14441 }
14442 \end{warpHTML}
```

112 MATHJAX warnings

\LWR@mathjaxwarn {⟨*packagename*⟩} {⟨*More text.*⟩}

      Issue a warning that MATHJAX is emulated. To be done \AtBeginDocument.

```
14443 \newcommand*{\LWR@mathjaxwarn}[2]{%
14444     \IfPackageLoadedTF{lwarp-#1}{%
14445        \ifblank{#2}{%
14446           \PackageWarningNoLine{lwarp}
14447              {%
14448                Lwarp provides emulation for MathJax when used\MessageBreak
14449                   with the #1 package%
14450              }
14451        }{%
14452           \PackageWarningNoLine{lwarp}
14453              {%
14454                Lwarp provides emulation for MathJax when used\MessageBreak
14455                   with the #1 package.\MessageBreak
14456                   #2%
14457              }
14458        }%
14459     }{}%
14460 }
14461
14462 % \begin{macro}{\LWR@nomathjaxwarn} \marg{packagename} \marg{More text.}
14463 %
14464 % Issue a warning that \MathJax\ is not supported.
14465 % To be done \cs{AtBeginDocument}.
14466 %
14467 % \changes{v0.894}{2020/12/22}{Warn if using packages not supported by \MathJax.}
14468 % \changes{v0.895}{2021/01/08}{Improved \MathJax\ warning.}
14469 %    \begin{macrocode}
14470 \newcommand*{\LWR@nomathjaxwarn}[2]{%
14471     \IfPackageLoadedTF{lwarp-#1}{%
14472        \ifblank{#2}{%
14473           \PackageWarningNoLine{lwarp}
14474              {%
14475                Lwarp does not provide MathJax support for #1.\MessageBreak
14476                   Use SVG math by removing the Lwarp mathjax option%
14477              }
```

```
14478          }{%
14479              \PackageWarningNoLine{lwarp}
14480                  {%
14481                      Lwarp does not provide MathJax support for #1.\MessageBreak
14482                          #2%
14483                  }
14484          }%
14485      }{}%
14486 }
```

\LWR@forceSVGmessage {⟨*packagename*⟩}

```
14487 \newcommand*{\LWR@forceSVGmessage}[1]{%
14488      SVG math output may be enabled for select math\MessageBreak
14489      expressions to preserve #1 visual\MessageBreak
14490      features for those particular expressions.\MessageBreak
14491     Before the chosen inline math, use \protect\inlinemathother\MessageBreak
14492      to begin using SVG math, and \protect\inlinemathnormal\MessageBreak
14493      afterward to resume using MathJax math.\MessageBreak
14494      Before display math, use \protect\displaymathother\MessageBreak
14495      to begin using SVG math, and use \protect\displaymathnormal\MessageBreak
14496      after to resume using MathJax for the following math.\MessageBreak
14497      Or, use SVG math for all expressions by removing\MessageBreak
14498      the mathjax option for the lwarp package%
14499 }
```

If MATHJAX is being used, issue a warning for certain packages.

```
14500 \AtBeginDocument{
14501      \ifbool{mathjax}{
14502          \LWR@nomathjaxwarn{aligned-overset}{}
14503          \LWR@nomathjaxwarn{amscdx}{\LWR@forceSVGmessage{amscdx}}
14504          \LWR@mathjaxwarn{arydshln}
14505              {In a math array, do not use the optional argument\MessageBreak
14506              for \protect\cdashline.\space\space
14507              Furthermore, \protect\cline\space is not\MessageBreak
14508              supported by MathJax}
14509          \LWR@nomathjaxwarn{autoaligne}{}
14510          \LWR@mathjaxwarn{autonum}
14511              {MathJax does not support equation+.\MessageBreak
14512              You may use the warpprint and warpHTML\MessageBreak
14513              environments to isolate the package load\MessageBreak
14514              and the equation+ environments}
14515          \LWR@mathjaxwarn{bigdelim}
14516              {Delimiters appear only of the first line}
14517          \LWR@nomathjaxwarn{boldtensors}{}
14518          \LWR@mathjaxwarn{booktabs}
14519              {\protect\cmidrule\space is not displayed}
14520          \LWR@mathjaxwarn{breqn}
14521              {Each environment becomes an SVG image}
14522          \LWR@mathjaxwarn{colortbl}
14523              {Colors are ignored in MathJax.\MessageBreak
14524              (Text mode tabular does support colortbl.)\MessageBreak
14525              \LWR@forceSVGmessage{colortbl}}
14526          \LWR@mathjaxwarn{delarray}{\LWR@forceSVGmessage{delarray}}
14527          \LWR@nomathjaxwarn{gauss}{\LWR@forceSVGmessage{gauss}}
14528          \LWR@mathjaxwarn{hhline}
14529              {A simple \protect\hline\space is used}
14530          \LWR@mathjaxwarn{isomath}
```

```
14531        {Some of the symbol font macros such as \protect\mathsfbfit\MessageBreak
14532            do not use a sans font because MathJax does not yet\MessageBreak
14533              have sans Greek.  Tensors may look like vectors%
14534            }
14535        \LWR@nomathjaxwarn{jkmath}{\LWR@forceSVGmessage{jkmath}}
14536        \LWR@mathjaxwarn{libertinust1math}
14537        {Some of the symbol font macros such as \protect\mathsfbfit\MessageBreak
14538            do not use a sans font because MathJax does not yet\MessageBreak
14539              have sans Greek.  Tensors may look like vectors%
14540            }
14541        \LWR@mathjaxwarn{mathtools}
14542          {See the Lwarp manual regarding the disallowspaces\MessageBreak
14543          and showonlyrefs options, the alignat environment,\MessageBreak
14544          \protect\DeclarePairedDelimiter\space and related,\MessageBreak
14545          and \cs{MakeAboxedCommand}%
14546          }
14547        \LWR@mathjaxwarn{mathspec}
14548          {Double quotes are removed, even inside \protect\text}
14549        \LWR@mathjaxwarn{mismath}
14550        {MathJax does not support \cs{enumber}, \cs{inumber},\MessageBreak
14551    \protect\jnumber, \protect\pinumber, \protect\MathUp, \protect\MathIt,\MessageBreak
14552          \protect\MathNumbers, or \protect\MathNormal.\MessageBreak
14553    \protect\itpi\space is made available as a clone of \protect\pi.\MessageBreak
14554          Tensors are not sans serif%
14555          }
14556        \LWR@mathjaxwarn{multirow}
14557          {Multirow works as expected in text mode, but\MessageBreak
14558          limited emulation is provided for MathJax math.\MessageBreak
14559        \protect\multirow\space ignores all arguments except\MessageBreak
14560          the text}
14561        \LWR@mathjaxwarn{nicematrix}
14562          {Keys/values are ignored in MathJax.\MessageBreak
14563          \protect\Cdots, etc. do not span multiple cells.\MessageBreak
14564          AutoNiceMatrix, etc. are not supported for MathJax.\MessageBreak
14565    \protect\CodeBefore, \protect\Body, and \protect\CodeAfter\MessageBreak
14566          \space\space also are not supported for MathJax.\MessageBreak
14567          \LWR@forceSVGmessage{nicematrix}%
14568          }
14569        \LWR@nomathjaxwarn{pb-diagram}{\LWR@forceSVGmessage{pb-diagram}}
14570 %      \LWR@mathjaxwarn{physics}
14571 % %          {The third-party extension is not used.\MessageBreak
14572 %          {The MathJax v3 extension is used.\MessageBreak
14573 %          See the Lwarp manual for details}
14574        \LWR@mathjaxwarn{siunitx}
14575    {Place \protect\sisetup\space before \protect\begin{document}.\MessageBreak
14576          Many optional arguments are ignored}
14577        \LWR@nomathjaxwarn{tensind}{}
14578        \LWR@mathjaxwarn{unicode-math}
14579          {Do not use embedded Unicode characters.\MessageBreak
14580          (Not all characters are encoded correctly.)\MessageBreak
14581          Some symbol fonts are not supported by MathJax,\MessageBreak
14582          and are only approximated.\MessageBreak
14583    Greek macros such as \protect\alpha\space respond to the math-style\MessageBreak
14584          option.  Latin symbols does not, per MathJax\MessageBreak
14585        limitations, unless placed inside \protect\symbit\space or similar}
14586        \LWR@nomathjaxwarn{unitsdef}{}
14587        \LWR@mathjaxwarn{witharrows}
14588          {Arrows can only point to the next line.\MessageBreak
14589          Text is only placed on a single line}
14590        \LWR@nomathjaxwarn{xy}
```

```
14591        {In text, xy works as-is.  SVG images will be generated.\MessageBreak
14592            \LWR@forceSVGmessage{xy}}
14593    }{}
14594 }
```

113 Temporary patches

These are temporary fixes for issues which probably soon will be fixed by others, outside of lwarp.

File 2 **`lwarp-2in1.sty`**

§ 114   Package **2in1**

2in1 (*Pkg*)   2in1 is ignored.

for HTML output: 1 `\LWR@ProvidesPackageDrop{2in1}`

File 3   **`lwarp-2up.sty`**

§ 115   Package **2up**

2up (*Pkg*)   2up is ignored.

for HTML output: 1 `\LWR@ProvidesPackageDrop{2up}[2010/05/15]`

2 `\def\source#1#2#3{}`
3 `\def\target#1#2#3{}`
4 `\def\targetlayout#1{}`
5 `\newdimen\pageseplength`
6 `\newdimen\pagesepwidth`
7 `\newdimen\pagesepoffset`
8 `\def\twoupemptypage{}`
9 `\def\twoupclearpage{}`
10 `\def\twoupeject{}`
11 `\def\twouparticle{}`
12 `\def\twoupplain{}`
13 `\def\twouplegaltarget{}`
14 `\def\twouplandscape{}`
15 `\def\TwoupWrites{}`

File 4   **`lwarp-a4.sty`**

§ 116   Package **a4**

a4 (*Pkg*)   a4 is ignored.

for HTML output: 1 `\LWR@ProvidesPackageDrop{a4}[2004/04/15]`

2 `\newcommand*{\WideMargins}{}`

File 5   **`lwarp-a4wide.sty`**

§ 117   Package **a4wide**

a4wide (*Pkg*)   a4wide is ignored.

for HTML output: 1 `\LWR@ProvidesPackageDrop{a4wide}[1994/08/30]`

File 6 **`lwarp-a5comb.sty`**

§ 118   Package **a5comb**

a5comb (*Pkg*)   a5comb is ignored.

for HTML output: 1 `\LWR@ProvidesPackageDrop{a5comb}`

File 7 **`lwarp-abstract.sty`**

§ 119   Package **abstract**

(*Emulates or patches code by* PETER WILSON.)

abstract (*Pkg*)   abstract is supported and patched by lwarp.

⚠ missing TOC   If using the number option with file splits, be sure to place the table of contents before the abstract. The number option causes a section break which may cause a file split, which would put a table of contents out of the home page if it is after the abstract.

for HTML output: memoir provides an abstract environment even though it is not an article or report class. Meanwhile, lwarp loads book to emulate memoir, but book does not have an abstract environment, so when the abstract package is loaded for emulation there is no pre-existing abstract to redefine, which would cause an error. Thus, a null abstract is provide here:

1 `\ProvideDocumentEnvironment{abstract}{}{}{}`

Accept all options for lwarp-abstract:

2 `\LWR@ProvidesPackagePass{abstract}[2009/06/08]`

```
3 \AtBeginDocument{
4 \BeforeBeginEnvironment{abstract}{
5 \LWR@forcenewpage
6 \BlockClass{abstract}
7 }
8 \AfterEndEnvironment{abstract}{\endBlockClass}
9 }
10
11 \renewcommand{\@bsrunintitle}{%
12 \hspace*{\abstitleskip}%
13 {\abstractnamefont%
14 \InlineClass{abstractrunintitle}{\abstractname}%
15 \@bslabeldelim}%
16 }
17
18 \IfClassLoadedTF{memoir}
19 {
20   \renewenvironment{abstract}{%
```

```
20 % %      \titlepage
21 %     \null\vfil
22 %     \@beginparpenalty\@lowpenalty
23   \setup@bstract
24     \if@bsrunin
25     \else
26 %       \if@bsstyle
27 %         \abstitlestyle{\BlockClassSingle{abstracttitle}{\abstractname}}
28 %       \else
29         \ifnumber@bs
30           \num@bs
31         \else
32         \begin{\absnamepos}%
33   \abstractnamefont \BlockClassSingle{abstracttitle}{\abstractname}
34 %           \@endparpenalty\@M
35           \end\absnamepos%
36       \vspace{\abstitleskip}%
37         \fi
38 %       \fi
39 %       \vspace{\abstitleskip}%
40     \fi
41     \put@bsintoc%
42     \begin{@bstr@ctlist}\if@bsrunin\@bsrunintitle\fi\abstracttextfont}%
43     {\par\end{@bstr@ctlist}%\vfil\null%\endtitlepage
44     }
45 }{% not memoir
46 \if@titlepage
47   \renewenvironment{abstract}{%
48 %      \titlepage
49     \null\vfil
50     \@beginparpenalty\@lowpenalty
51     \if@bsrunin
52     \else
53       \if@bsstyle
54         \abstitlestyle{\BlockClassSingle{abstracttitle}{\abstractname}}
55       \else
56         \ifnumber@bs
57           \num@bs
58         \else
59         \begin{\absnamepos}%
60   \abstractnamefont \BlockClassSingle{abstracttitle}{\abstractname}
61           \@endparpenalty\@M
62           \end\absnamepos%
63 %%        \vspace{\abstitleskip}%
64         \fi
65       \fi
66       \vspace{\abstitleskip}%
67     \fi
68     \put@bsintoc%
69     \begin{@bstr@ctlist}\if@bsrunin\@bsrunintitle\fi\abstracttextfont}%
70     {\par\end{@bstr@ctlist}\vfil\null%\endtitlepage
71     }
72 \else
73   \renewenvironment{abstract}{%
74     \if@bsrunin
75     \else
76       \if@bsstyle
77         \abstitlestyle{\BlockClassSingle{abstracttitle}{\abstractname}}
78       \else
79         \ifnumber@bs
```

```
80          \num@bs
81        \else
82 \begin{\absnamepos}%
83 \abstractnamefont\BlockClassSingle{abstracttitle}{\abstractname}%
84 \end\absnamepos%
85 %%        \vspace{\abstitleskip}%
86        \fi
87      \fi
88      \vspace{\abstitleskip}%
89    \fi
90    \put@bsintoc%
91    \begin{@bstr@ctlist}\if@bsrunin\@bsrunintitle\fi\abstracttextfont}%
92    {\par\end{@bstr@ctlist}}
93 \fi
94 }% not memoir
```

File 8 `lwarp-academicons.sty`

§ 120   Package **academicons**

(*Emulates or patches code by* Diogo A. B. Fernandes.)

academicons (*Pkg*) academicons is patched for use by lwarp.

If \aiicon is used, the name of the icon is used in the alt tag. Otherwise, for each of the individual icon macros, a generic alt tag is used.

for HTML output: 1 \LWR@ProvidesPackagePass{academicons}[2018/06/27]

```
2 \LetLtxMacro\LWR@orig@symbol\symbol
3
4 \let\LWR@academicons@orig@AI\AI
5
6 \newcommand*{\LWR@academicons@symbol}[1]{%
7    \begin{lateximage}*[academicon][academicons#1]%
8    \begingroup%
9    \LWR@academicons@orig@AI%
10   \LWR@orig@symbol{#1}%
11   \endgroup%
12   \end{lateximage}%
13 }
14
15 \renewcommand*{\AI}{%
16   \LetLtxMacro\symbol\LWR@academicons@symbol%
17 }
18
19 \renewcommand*{\aiicon}[1]
20 {%
21   \begin{lateximage}*[#1 icon]?[academicons#1]%
22   \AI\csname aiicon@#1\endcsname%
23   \end{lateximage}%
24 }
```

File 9   **`lwarp-accents.sty`**

§ 121   Package   # accents

(*Emulates or patches code by* JAVIER BEZOS.)

accents (*Pkg*)   accents is used as-is for SVG math, and is emulated for MATHJAX.

for HTML output: 1 `\LWR@ProvidesPackagePass{accents}[2006/05/12]`

For MATHJAX:

```
2 \begin{warpMathJax}
3 \LWR@infoprocessingmathjax{accents}
4
5 \CustomizeMathJax{\newcommand{\ring}[1]{\mathring{#1}}}
6 \CustomizeMathJax{\newcommand{\accentset}[2]{\overset{#1{}}{#2}}}
```

As of this writing, MATHJAX v3 does not yet support groups for macros, so for \underaccent, the originals are remembered here, then they are temporarily redefined and used inside \underaccent, then restored to their originals. \LARGE gives a reasonable size, and \raise is used to adjust vertically without introducing extra line space.

```
7 \CustomizeMathJax{\let\LWRgrave\grave}
8 \CustomizeMathJax{\let\LWRacute\acute}
9 \CustomizeMathJax{\let\LWRcheck\check}
10 \CustomizeMathJax{\let\LWRbreve\breve}
11 \CustomizeMathJax{\let\LWRbar\bar}
12 \CustomizeMathJax{\let\LWRhat\hat}
13 \CustomizeMathJax{\let\LWRdot\dot}
14 \CustomizeMathJax{\let\LWRtilde\tilde}
15 \CustomizeMathJax{\let\LWRddot\ddot}
16 \CustomizeMathJax{\let\LWRvec\vec}
17 \CustomizeMathJax{\let\LWRwidetilde\widetilde}
18
19 \CustomizeMathJax{\newcommand{\underaccent}[2]{%
20     {%
21     \renewcommand{\grave}[1]{{\LARGE\LWRgrave{##1}}}%
22     \renewcommand{\acute}[1]{{\LARGE\LWRacute{##1}}}%
23     \renewcommand{\check}[1]{{\LARGE\LWRcheck{##1}}}%
24     \renewcommand{\breve}[1]{{\LARGE\LWRbreve{##1}}}%
25     \renewcommand{\bar}[1]{{\LARGE\LWRbar{##1}}}%
26     \renewcommand{\hat}[1]{{\LARGE\LWRhat{##1}}}%
27     \renewcommand{\dot}[1]{{\LARGE\LWRdot{##1}}}%
28     \renewcommand{\tilde}[1]{{\LARGE\LWRtilde{##1}}}%
29     \renewcommand{\ddot}[1]{{\LARGE\LWRddot{##1}}}%
30     \renewcommand{\vec}[1]{{\LARGE\LWRvec{##1}}}%
31     \renewcommand{\widetilde}[1]{{\LARGE\LWRwidetilde{\hphantom{#2}}}}%
32     \underset{\raise 2pt {#1{}}}{#2}%
33     \let\grave\LWRgrave%
34     \let\acute\LWRacute%
35     \let\check\LWRcheck%
36     \let\breve\LWRbreve%
37     \let\bar\LWRbar%
```

```
38      \let\hat\LWRhat%
39      \let\dot\LWRdot%
40      \let\tilde\LWRtilde%
41      \let\ddot\LWRddot%
42      \let\vec\LWRvec%
43      \let\widetilde\LWRwidetilde%
44      }%
45 }}
46
47 \CustomizeMathJax{\newcommand{\undertilde}[1]{%
48      \underset{\raise 3pt {\widetilde{\hphantom{#1}}}}{#1}%
49 }}
50 \end{warpMathJax}
```

File 10   `lwarp-accessibility.sty`

§ 122   Package   # accessibility

accessibility (*Pkg*)   accessibility is emulated.

for HTML output: Discard all options for lwarp-accessibility:

```
1 \LWR@ProvidesPackageDrop{accessibility}[2019/10/14]
```

```
2 \newcommand{\alt}[1]{\ThisAltText{#1}}
3 \newcommand{\newhref}[3]{\ThisAltText{#2}\LWR@href{#1}{#3}}%
4 \providecommand{\thead}[1]{\textbf{#1}}
```

For MATHJAX:

```
5 \begin{warpMathJax}
6 \CustomizeMathJax{\newcommand{\alt}[1]{}}
7 \CustomizeMathJax{\newcommand{\thead}[1]{\text{\textbf{#1}}}}
8 \end{warpMathJax}
```

File 11   `lwarp-accsupp.sty`

§ 123   Package   # accsupp

accsupp (*Pkg*)   accsupp is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{accsupp}[2018/03/28]`

```
2 \newcommand*{\BeginAccSupp}[1]{}
3 \newcommand*{\EndAccSupp}[1]{}
```

For MATHJAX:

```
4 \begin{warpMathJax}
5 \CustomizeMathJax{\newcommand{\BeginAccSupp}[1]{}}
6 \CustomizeMathJax{\newcommand{\EndAccSupp}[1]{}}
7 \end{warpMathJax}
```

File 12 **`lwarp-acro.sty`**

§ 124   Package **acro**

(*Emulates or patches code by* CLEMENS NIEDERBERGER.)

acro (*Pkg*)   acro is patched for use by lwarp.

⚠   formats   Define acronymn formats using \textbf instead of \bfseries etc.

for HTML output: 1 \LWR@ProvidesPackagePass{acro}[2019/10/12]

\DeclareAcronym is used in the preamble, where lwarp has not yet made the dollar
active, so temporarily enable lwarp math catcode just for this definition:

```
2 \ExplSyntaxOn
3 \NewDocumentCommand \LWR@DeclareAcronym {mm}
4 {
5     \acro_declare_acronym:nn {#1} {#2}
6     \catcode`\$=3% lwarp
7 }
8 \ExplSyntaxOff
9
10 \RenewDocumentCommand{\DeclareAcronym}{}{
11     \catcode`\$=\active% lwarp
12     \LWR@DeclareAcronym
13 }
```

Replace dot fill with simple dots:

```
14 \ExplSyntaxOn
15 \cs_new_protected:Npn \LWR@HTML@acro_dot_fill: {\dots\space}
16 \LWR@formatted{acro_dot_fill:}
17 \ExplSyntaxOff
```

Modified to activate the current font:

```
18 \ExplSyntaxOn
19 \IfPackageAtLeastTF{acro}{2020/04/29}%
20 {}% v3 or later
21 {% before v3
22 \IfPackageAtLeastTF{acro}{2019/09/23}%
23 {% v2.10 or later
24 \cs_gset_protected:Npn \__acro_typeset:nn #1#2
25   {
26     \mode_if_horizontal:F { \leavevmode }
27     \group_begin:
28       \use:x
29         {
30           \bool_if:cTF {l__acro_custom_#1_format_bool}
31             { \exp_not:v {l__acro_custom_#1_format_tl} }
32             { \exp_not:v {l__acro_#1_format_tl} }
33             {\exp_not:N\LWR@textcurrentfont{#2}}%   lwarp
34         }
35     \group_end:
```

```
36    }
37
38 \cs_gset_protected:Npn \__acro_ending_format:nn #1#2
39    {
40      \bool_if:NTF \l__acro_include_endings_format_bool
41        {
42          \str_case:nn {#1}
43            {
44              {long}
45              {
46                \bool_if:NTF \l__acro_custom_long_format_bool
47                  { \l__acro_custom_long_format_tl }
48                  {
49                    \bool_if:NTF \l__acro_first_instance_bool
50                      { \l__acro_first_long_format_tl }
51                      { \l__acro_long_format_tl }
52                  }
53              }
54              {short}
55              {
56                \bool_if:NTF \l__acro_custom_short_format_bool
57                  { \l__acro_custom_short_format_tl }
58                  { \l__acro_short_format_tl }
59              }
60              {alt}
61              {
62                \bool_if:NTF \l__acro_custom_alt_format_bool
63                  { \l__acro_custom_alt_format_tl }
64                  { \l__acro_alt_format_tl }
65              }
66            }
67        }
68        { \use:n }
69        {\exp_not:N\LWR@textcurrentfont{#2}}% lwarp
70    }
71 }% v2.10 or later
72 {% before v2.10
73 \cs_gset_protected:Npn \acro_write_short:nn #1#2
74    {
75      \mode_if_horizontal:F { \leavevmode }
76      \group_begin:
77        \bool_if:NTF \l__acro_custom_format_bool
78          { \l__acro_custom_format_tl }
79          { \l__acro_short_format_tl }
80        {\LWR@textcurrentfont{#2}}% lwarp
81      \group_end:
82    }
83
84 \cs_gset_protected:Npn \acro_write_alt:nn #1#2
85    {
86      \mode_if_horizontal:F { \leavevmode }
87      \group_begin:
88        \bool_if:NTF \l__acro_custom_format_bool
89          { \l__acro_custom_format_tl }
90          { \l__acro_alt_format_tl }
91        {\LWR@textcurrentfont{#2}}% lwarp
92      \group_end:
93    }
94
95 \cs_gset_protected:Npn \acro_write_long:nn #1#2
```

```
96  {
97    \mode_if_horizontal:F { \leavevmode }
98    \group_begin:
99      \bool_if:NTF \l__acro_custom_long_format_bool
100       { \l__acro_custom_long_format_tl }
101       { \use:n }
102     {
103       \use:x
104         {
105           \exp_not:n {#1}
106           {
107             \bool_if:NTF \l__acro_first_upper_bool
108               { \exp_not:N \__acro_first_upper_case:n { \exp_not:n {
109                   \LWR@textcurrentfont{#2}% lwarp
110               } } }
111               { \exp_not:n {\LWR@textcurrentfont{#2}} }% lwarp
112         }
113       }
114     }
115   \group_end:
116  }
117 }% before v2.10
118 }% before v3
119 \ExplSyntaxOff
```

File 13 **`lwarp-acronym.sty`**

§ 125   Package # acronym

(*Emulates or patches code by* Tobias Oetiker.)

acronym (*Pkg*)   acronym is patched for use by lwarp.

⚠ multiply-defined labels   \acresetall does not work with cleveref, causing multiply-defined labels. lwarp
patches acronym for HTML, but not for print mode.

for HTML output: 1 \LWR@ProvidesPackagePass{acronym}[2020/04/17]

Simplifies for HTML. Unable to use \VerifyCommand here due to \csname being
used.

```
2 \expandafter\def\csname AC@\AC@prefix{}@acro\endcsname#1[#2]#3{%
3   \ifAC@nolist%
4   \else%
5   \ifnum%
6     \ifAC@printonlyused 1%
7     \else\ifAC@printonlyreused 1%
8     \else 0\fi\fi%
9   =1\relax%
10    \ifnum%
11      \ifAC@printonlyused%
12      \expandafter\ifx\csname acused@#1@once\endcsname\AC@used 1 \else 0 \fi%
13      \else\ifAC@printonlyreused%
14      \expandafter\ifx\csname acused@#1@twice\endcsname\AC@used 1 \else 0 \fi%
15      \else 0 \fi\fi%
16    =1\relax%
17      \item[\protect\AC@hypertarget{#1}{%
```

```
18          \AC@hyperref[acro:#1]{\aclabelfont{#2}\hfill}%
19       }]\AC@hyperref[acro:#1]{#3}%
20         \ifAC@withpage%
21           \expandafter\ifx\csname r@acro:#1\endcsname\relax%
22             \PackageInfo{acronym}{%
23               Acronym #1 used in text but not spelled out in
24               full in text}%
25           \else%
26 %            \nobreak\leaders\hbox{%
27 %              $\m@th\mkern\@dotsep mu\hbox{.}\mkern\@dotsep mu$%
28 %            }\hfill%
29 %            \nobreak\hb@xt@\@pnumwidth{%
30 %            \hfil\normalfont\normalcolor
31             \qquad --- %      lwarp
32             \AC@pageref{acro:#1}%
33 %            }%
34         \fi%
35       \fi\\%
36     \fi%
37 \else%
38 \item[\protect\AC@hypertarget{#1}{\AC@hyperref[acro:#1]{\aclabelfont{#2}\hfill}}]%
39     \AC@hyperref[acro:#1]{#3}%
40 \fi%
41 \fi%
42 \begingroup
43   \def\acroextra##1{}%
44   \@bsphack
45     \ifAC@printonlyreused%
46       \protected@write\@auxout{}{%
47         \string\newacro{#1}[%
48           \expandafter\ifx\csname acused@#1@twice\endcsname\AC@used%
49             \string\AC@hyperlink{#1}{#2}%
50           \else%
51             {#2}%
52           \fi%
53         ]{#3}%
54       }%
55     \else%
56       \protected@write\@auxout{}{%
57         \string\newacro{#1}[\string\AC@hyperlink{#1}{#2}]{#3}%
58       }%
59     \fi%
60   \@esphack
61 \endgroup
62 \ignorespaces}
```

Uses `\textit` instead of `\itshape`:

```
63 \renewcommand{\acfia}[1]{%
64 {\textit{\AC@acl{#1}}} (\ifAC@starred\acs*{#1}\else\acs{#1}\fi)}
```

Removes the mbox to allow math inside:

```
65 \VerifyCommand[lwarp][acronym]{\AC@acs}{E2119484F7CD2A5D4B064390C6BB806F}
66
67 \renewcommand*\AC@acs[1]{%
68 %    \mbox{
69 \expandafter\AC@get\csname fn@#1\endcsname\@firstoftwo{#1}}}
70 %  }
```

Fix for acronym labels in the captions of floats.

```
71 \renewcommand{\@starttoc}[1]{%
72     \LWR@htmlelementclass{nav}{#1}
73     \LetLtxMacro\@verridelabel\@gobble
74     \LWR@orig@starttoc{#1}
75     \LWR@htmlelementclassend{nav}{#1}
76 }
```

Modified for cleveref and lwarp:

```
77 \VerifyCommand[lwarp][acronym]{\AC@und@newl@bel}{661CF70DCB3E1AA8871B26E785BE7C86}
78
79 \renewcommand*\AC@und@newl@bel[3]{%
80     \@ifundefined{#1@#3}%
81     {%
82         \global\expandafter\let\csname#2@#3\endcsname\@nnil
83         \global\expandafter\let\csname#2@#3@lwarp\endcsname\@nnil% lwarp
84         \global\expandafter\let\csname#2@#3@cref\endcsname\@nnil% lwarp
85     }%
86     {%
87         \global\expandafter\let\csname#1@#3\endcsname\relax
88         \global\expandafter\let\csname#1@#3@lwarp\endcsname\relax% lwarp
89         \global\expandafter\let\csname#1@#3@cref\endcsname\relax% lwarp
90     }%
91 }%
```

Improve paragraph handling:

```
92 \BeforeBeginEnvironment{acronym}{\LWR@stoppars}
93 \AfterEndEnvironment{acronym}{\LWR@startpars}
```

Create hyperlinks, even though hyperref is only emulated:

```
94 \AtBeginDocument{
95     \LetLtxMacro\AC@hyperlink\hyperlink
96     \LetLtxMacro\AC@hyperref\hyperref
97     \newcommand*\AC@raisedhypertarget[2]{%
98 %       \Hy@raisedlink{%
99 %         \hypertarget{#1}{}%
100 %        }%
101       #2}%
102     \LetLtxMacro\AC@hypertarget\AC@raisedhypertarget
103     \def\AC@phantomsection{%
104 %      \Hy@GlobalStepCount\Hy@linkcounter
105 %      \edef\@currentHref{section*.\the \Hy@linkcounter}%
106 %      \Hy@raisedlink{%
107 %        \hyper@anchorstart{\@currentHref}\hyper@anchorend
108 %      }%
109 %      \phantomsection%
110     }%
111 }
112
113 \appto\LWR@restoreorigformatting{%
114     \LetLtxMacro\AC@hyperlink\@secondoftwo%
115     \LetLtxMacro\AC@hyperref\LWR@nullify@hyperref%
116 }
```

File 14   `lwarp-adjmulticol.sty`

§ 126   Package   # adjmulticol

(*Emulates or patches code by* BORIS VEYTSMAN.)

adjmulticol (*Pkg*)   adjmulticol is emulated.

Emulation similar to multicols is used, with adjusted margins. If the number of columns is specified as 1, it is set so, but if two or greater are used, lwarp allows a variable number of columns up to three.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{adjmulticol}[2012/01/20]
```

```
2 \RequirePackage{multicol}
```

adjmulticols   * {⟨*numcols*⟩} {⟨*left margi*⟩} {⟨*right margin*⟩}

```
3 \NewDocumentEnvironment{adjmulticols}{s m m m}
4 {%
```

Compute the margins, and limit to positive only:

```
5 \setlength{\LWR@templengthone}{#3}%
6 \ifdimcomp{\LWR@templengthone}{<}{0pt}{\setlength{\LWR@templengthone}{0pt}}{}%
7 \setlength{\LWR@templengthtwo}{#4}
8 \ifdimcomp{\LWR@templengthtwo}{<}{0pt}{\setlength{\LWR@templengthtwo}{0pt}}{}%
```

If one column is specified, use a <div> of class singlecolumn, else use multicols:

```
9 \newcommand*{\LWR@mcolstype}{multicols}%
10 \ifnumcomp{#2}{=}{1}{\renewcommand*{\LWR@mcolstype}{singlecolumn}}{}%
```

Help avoid page overflow:

```
11 \LWR@forcenewpage%
```

Create the <div> with the given margin and class:

```
12 \BlockClass[%
13     \LWR@print@mbox{margin-left:\LWR@printlength{\LWR@templengthone}} ; %
14     \LWR@print@mbox{margin-right:\LWR@printlength{\LWR@templengthtwo}}%
15 ]{\LWR@mcolstype}%
16 }
17 {\endBlockClass}
```

File 15   `lwarp-addlines.sty`

§ 127   Package   # addlines

(*Emulates or patches code by* WILL ROBERTSON.)

addlines (*Pkg*) addlines is ignored.

1 \LWR@ProvidesPackageDrop{addlines}[2018/12/05]

2 \newcommand\addlines{\@ifstar\addlines@a\addlines@a}
3 \newcommand\addlines@a[1][1]{}
4 \let\addline\addlines
5 \newcommand\removelines{\@ifstar\removelines@a\removelines@a}
6 \newcommand\removelines@a[1][1]{}
7 \let\removeline\removelines
8 \newcommand\squeezepage[1][0]{}

File 16 `lwarp-afterpage.sty`

§ 128   Package # afterpage

(*Emulates or patches code by* DAVID CARLISLE.)

afterpage (*Pkg*) afterpage is emulated.

Discard all options for lwarp-afterpage:

1 \LWR@ProvidesPackageDrop{afterpage}[2014/10/28]

2 \newcommand{\afterpage}[1]{#1}

File 17 `lwarp-algorithm2e.sty`

§ 129   Package # algorithm2e

(*Emulates or patches code by* CHRISTOPHE FIORIO.)

algorithm2e (*Pkg*) algorithm2e is patched for use by lwarp.

For print output, captions are placed according to package options, but for HTML output captions are placed where used. Therefore, to have captions appear at the top of the algorithms for both print and HTML, place each captions at the top of each algorithm.

1 \LWR@ProvidesPackagePass{algorithm2e}[2017/07/18]

For the list-of entries:

2 \renewcommand{\l@algocf}[2]{\hypertocfloat{1}{algocf}{loa}{#1}{#2}}

Select the lwarp float style according to the algorithm2e style:

3 \newcommand*{\LWR@floatstyle@algocf}{ruled}
4
5 \ifdefstring{\algocf@style}{boxed}{%
6 \renewcommand*{\LWR@floatstyle@algocf}{boxed}
7 }{}
8

```
9 \ifdefstring{\algocf@style}{boxruled}{%
10 \renewcommand*{\LWR@floatstyle@algocf}{boxruled}
11 }{}
12
13 \ifdefstring{\algocf@style}{plain}{%
14 \renewcommand*{\LWR@floatstyle@algocf}{plain}
15 }{}
```

Paragraph handling to allow line numbers under certain conditions:

```
16 \renewcommand{\algocf@everypar}{%
17     \ifbool{LWR@algocf@dopars}{%
18         \ifbool{LWR@doingstartpars}{%
19             \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
20                 {}%
21                 {%
```

algorthm2e uses \everypar, so the open paragraph tag is generated here instead of \LWR@openparagraph:

```
22                     \LWR@htmltagc{\LWR@tagregularparagraph}\LWR@orignewline%

23                     \algocf@everyparnl\algocf@everyparhanging%
24                 }%
25         }{}%
26     }{}%
27 }
```

lwarp caption handling:

```
28 \renewcommand{\algocf@makecaption}[2]{%
29     \LWR@HTML@caption@begin{algocf}%
30     \LWR@isolate{\algocf@captiontext{#1}{#2}}%
31     \LWR@HTML@caption@end%
32 }
```

Print any caption where it is declared:

```
33 \renewcommand{\algocf@makecaption@plain}[2]{%
34     \LWR@HTML@caption@begin{algocf}%
35     \LWR@isolate{\algocf@captiontext{#1}{#2}}%
36     \LWR@HTML@caption@end%
37 }
38
39 \renewcommand{\algocf@makecaption@boxed}[2]{%
40     \LWR@HTML@caption@begin{algocf}%
41     \LWR@isolate{\algocf@captiontext{#1}{#2}}%
42     \LWR@HTML@caption@end%
43 }
44
45 \renewcommand{\algocf@makecaption@ruled}[2]{%
46     \LWR@HTML@caption@begin{algocf}%
47     \LWR@isolate{\algocf@captiontext{#1}{#2}}%
48     \LWR@HTML@caption@end%
49 }
```

Turn off line numbering while making the caption:

```
50 \long\def\algocf@latexcaption#1[#2]#3{% original definition of caption
51 \boolfalse{LWR@algocf@dopars}%        lwarp
52   \par%
53   \addcontentsline{\csname ext@#1\endcsname}{#1}%
54 {\protect\numberline{\csname the#1\endcsname}{\ignorespaces \LWR@isolate{#2}}}%
55   \begingroup%
56   \@parboxrestore%
57   \if@minipage%
58     \@setminipage%
59   \fi%
60   \normalsize%
61   \@makecaption{\csname fnum@#1\endcsname}{\ignorespaces #3}\par%
62   \endgroup%
63 \booltrue{LWR@algocf@dopars}%        lwarp
64 }
```

Line numbers are printed in a <span> of class alg2elinenumber:

```
65 \renewcommand{\algocf@printnl}[1]{%
66     \InlineClass{alg2elinenumber}{\NlSty{#1}}~%
67 }%
```

While initializing an algorithm environment, locally declare the style of a regular figure to be the same as the algorithm style, in case the figure option was used.

```
68 \preto\@algocf@init{%
69   \edef\LWR@floatstyle@figure{\LWR@floatstyle@algocf}%
70 }
```

For lwarp, the algorithm is not assembled inside a box, since lateximages would not work, so the captions are printed where declared.

```
71 \renewcommand{\@algocf@start}{%
72   \let\@mathsemicolon=\;\def\;{\ifmmode\@mathsemicolon\else\@endalgoln\fi}%
73 %   \raggedright%
74     \AlFnt{}%
75     \booltrue{LWR@algocf@dopars}% lwarp
76 }
77
78 \renewcommand{\@algocf@finish}{%
79     \boolfalse{LWR@algocf@dopars}% lwarp
80     \lineskip\normallineskip\setlength{\skiptotal}{\@defaultskiptotal}%
81     \let\;=\@mathsemicolon%
82     \let\]=\@emathdisplay%
83 }
```

Use an HTML break:

```
84 \renewcommand{\BlankLine}{%
85 \LWR@stoppars%
86 \LWR@htmltagc{br /}%
87 \LWR@startpars%
88 }
```

Simplified for HTML. The paragraph handling must be preserved.

```
89 \renewcommand{\SetKwInOut}[2]{%
90   \algocf@newcommand{#1}[1]{%
91     \ifthenelse{\boolean{algocf@hanginginout}}%
```

```
 92        {\relax}%
 93        {\algocf@seteveryparhanging{\relax}}%
 94     \ifthenelse{\boolean{algocf@inoutnumbered}}%
 95        {\relax}%
 96        {\algocf@seteveryparnl{\relax}}%
 97     {%
 98            \KwSty{#2\algocf@typo:}%
 99        ~##1\par%
100     }%
101     \algocf@linesnumbered% reset the numbering of the lines
102     \ifthenelse{\boolean{algocf@hanginginout}}%
103        {\relax}%
104        {\algocf@reseteveryparhanging}%
105   }%
106 }%
107
108 \renewcommand{\ResetInOut}[1]{}%
```

Each of the following creates a <div> of a given class, and turns off line numbering while creating the <div> tags:

```
109 \renewcommand{\algocf@Vline}[1]{%
110     \boolfalse{LWR@algocf@dopars}%
111     \begin{BlockClass}{alg2evline}
112     \booltrue{LWR@algocf@dopars}%
113     #1
114     \boolfalse{LWR@algocf@dopars}%
115     \end{BlockClass}
116     \booltrue{LWR@algocf@dopars}%
117 }
```

```
118 \renewcommand{\algocf@Vsline}[1]{%
119     \boolfalse{LWR@algocf@dopars}%
120     \begin{BlockClass}{alg2evsline}
121     \booltrue{LWR@algocf@dopars}%
122     #1
123     \boolfalse{LWR@algocf@dopars}%
124     \end{BlockClass}
125     \booltrue{LWR@algocf@dopars}%
126 }
```

```
127 \renewcommand{\algocf@Noline}[1]{%
128     \boolfalse{LWR@algocf@dopars}%
129     \begin{BlockClass}{alg2enoline}
130     \booltrue{LWR@algocf@dopars}%
131     #1
132     \boolfalse{LWR@algocf@dopars}%
133     \end{BlockClass}
134     \booltrue{LWR@algocf@dopars}%
135 }
```

The [H] environment is converted to a regular float, which in HTML is placed where declared. Reusing the regular float allows the [H] version to reuse the ruled and boxed options.

```
136 \LetLtxMacro\algocf@Here\algocf
137 \LetLtxMacro\endalgocf@Here\endalgocf
```

File 18 **`lwarp-algorithmicx.sty`**

§ 130   Package   # algorithmicx

(*Emulates or patches code by* Szász János.)

algorithmicx (*Pkg*)   algorithmicx is supported with minor adjustments.

for HTML output: 1 \LWR@ProvidesPackagePass{algorithmicx}[2005/04/27]

Inside the algorithmic environment, level indenting is converted to a <span> of the required length, and comments are placed inside a <span> which is floated right.

⚠ package conflicts   If using \newfloat, trivfloat, and/or algorithmicx together, see section 646.1.

```
2 \AtBeginEnvironment{algorithmic}{%
3 %
4 \let\origALG@doentity\ALG@doentity%
5 %
6 \renewcommand*{\ALG@doentity}{%
7 \origALG@doentity%
8 \LWR@htmltagc{%
9     span style=\textquotedbl{}%
10        width:\LWR@printlength{\ALG@thistlm}; display:inline-block;%
11    \textquotedbl%
12 }%
13 \ifbool{FormatWP}{%
14 \setlength{\LWR@templengthone}{\the\ALG@thistlm}%
15 \whiledo{\lengthtest{\LWR@templengthone>1em}}{%
16 \quad%
17 \addtolength{\LWR@templengthone}{-1em}%
18 }%
19 }{}%
20 \LWR@htmltagc{/span}%
21 }%
22
23 \let\LWR@origComment\Comment%
24
25 \renewcommand{\Comment}[1]{%
26     \InlineClass{floatright}{\LWR@origComment{#1}}%
27 }%
28 }
29
30 \renewcommand\algorithmiccomment[1]{%
31 \hfill\HTMLunicode{25B7} #1% white right triangle
32 }%
```

File 19 **`lwarp-alltt.sty`**

§ 131   Package   # alltt

(*Emulates or patches code by* Johannes Braams.)

alltt (*Pkg*)   alltt is patched for use by lwarp.

for HTML output: 1 `\LWR@ProvidesPackagePass{alltt}[1997/06/16]`

```
2 \AfterEndPreamble{
3 \LWR@traceinfo{Patching alltt.}
4
5 \AtBeginEnvironment{alltt}{%
6     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
7         {}%
8         {%
9             \LWR@forcenewpage
```

Vertical spacing changes if inside a list.

```
10             \LWR@atbeginverbatim{alltt}%
11         }%
12 }
13
14 \AfterEndEnvironment{alltt}{%
15     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
16         {}%
17         {%
```

Vertical spacing changes if inside a list.

```
18             \LWR@afterendverbatim%
19         }%
20 }
21
22 }
```

File 20   **`lwarp-amscdx.sty`**

§ 132   Package   # amscdx

(*Emulates or patches code by* MARTIN VERMEER.)

amscdx (*Pkg*)   amscdx is used as-is for SVG math.

⚠ MATHJAX   For MATHJAX, a warning notes that the CD environment must be enclosed between
`\displaymathother` and `\displaymathnormal`.

for HTML output: 1 `\LWR@ProvidesPackagePass{amscdx}[2019/07/02]`

```
2 \begin{warpMathJax}
3 \CustomizeMathJax{%
4     \renewenvironment{CD}
5     {\text{(Use \unicode{x005C}displaymathother before the CD enviroment.) \quad}}
6     {\quad \text{(Use \unicode{x005C}displaymathnormal after the CD enviroment.)}}
7 }
8
9 \CustomizeMathJax{\newcommand{\CDfattrue}{}}
10 \CustomizeMathJax{\newcommand{\CDfatfalse}{}}
11 \CustomizeMathJax{\newcommand{\CDashtrue}{}}
12 \CustomizeMathJax{\newcommand{\CDashfalse}{}}
```

```
13 \CustomizeMathJax{\newcommand{\CDlor}[1]{}}
14 \end{warpMathJax}
```

File 21   **`lwarp-amsmath.sty`**

§ 133   Package   **amsmath**

(*Emulates or patches code by* AMERICAN MATHEMATICAL SOCIETY, LATEX3 PROJECT.)

amsmath (*Pkg*)   amsmath is patched for use by lwarp.

for HTML output: `1 \LWR@ProvidesPackagePass{amsmath}[2017/09/02]`

\dotso   An HTML text-mode version.

```
2 \newcommand*{\LWR@HTML@dotso}{\textellipsis\ }
3 \LWR@formatted{dotso}
```

Patches to allow \eqref inside a caption:

```
4 \def\maketag@@@#1{\text{#1}}
5 \def\tagform@#1{\maketag@@@{(\ignorespaces#1\unskip)}}
```

Patches for $\mathcal{A}_{\mathcal{M}}\mathcal{S}$ math \tag macro to remember the first tag:

```
6 \ifbool{mathjax}{}{% not mathjax
7
8 \VerifyCommand[lwarp][amsmath]{\make@df@tag@@}{A5AA7B9CD20DC2C73B1D19D582C44A8E}
9 \VerifyCommand[lwarp][amsmath]{\make@df@tag@@@}{670399C01F88B0E9B0874E9B129FA404}
10
11 \LetLtxMacro\LWR@origmake@df@tag@@\make@df@tag@@
12 \LetLtxMacro\LWR@origmake@df@tag@@@\make@df@tag@@@
13
14 \renewcommand*{\make@df@tag@@}[1]{%
15     \LWR@remembertag{#1}%
16     \LWR@origmake@df@tag@@{#1}%
17 }
18
19 \renewcommand*{\make@df@tag@@@}[1]{%
20     \LWR@remembertag{#1}%
21     \LWR@origmake@df@tag@@@{#1}%
22 }
23
24 }% not mathjax
```

For nesting $\mathcal{A}_{\mathcal{M}}\mathcal{S}$ environments:

```
25 \newcounter{LWR@amsmathdepth}
26 \setcounter{LWR@amsmathdepth}{0}
```

The following $\mathcal{A}_{\mathcal{M}}\mathcal{S}$ environments are patched in-place:

LWR@maxfields@ (*Ctr*)   A copy of maxfields@ as it was passed. This is used to generate the mandatory argument for alignat and alignat* when using MATHJAX.

```
27 \newcounter{LWR@maxfields@}
28
29 \VerifyCommand[lwarp][amsmath]{\start@align}{D39AF6A45F9E97A21F17EADB4D21D218}
30
31 \xpatchcmd{\start@align}
32     {\maxfields@#3\relax}
33     {%
34         \maxfields@#3\relax%
35         \setcounter{LWR@maxfields@}{#3}%
36     }
37     {}
38     {\LWR@patcherror{amsmath}{start@align}}
```

\LWR@amsmathenv@@before   * {⟨*environment name*⟩}

* if the environment was starred.

Embeds the environment inside a lateximage.

```
39 \NewDocumentCommand{\LWR@amsmathenv@@before}{s m}{%
40     \IfBooleanTF{#1}{
41         \begin{BlockClass}{displaymath}
42     }{
43         \begin{BlockClass}{displaymathnumbered}
44     }
45     \LWR@newautoidanchor%
46     \booltrue{LWR@indisplaymathimage}%
47     \begin{lateximage}[\LWR@amsmathbodynumbered{#2}]*?%
48     \LWR@applyxfakebold%
49 }
```

\LWR@amsmathenv@before   * {⟨*environment name*⟩}

* if the environment was starred.

Embeds the environment with MATHJAX or a lateximage.

```
50 \NewDocumentCommand{\LWR@amsmathenv@before}{s m}{%
51     \ifnumequal{\value{LWR@amsmathdepth}}{0}{%
52         \LWR@stoppars%
53     \ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
54     {
55         \LWR@syncmathjax
56         \boolfalse{LWR@amsmultline}
57         \ifstrequal{#2}{multline}{\booltrue{LWR@amsmultline}}{}
58         \ifstrequal{#2}{multline*}{\booltrue{LWR@amsmultline}}{}
```

⚠ autonum's "+" environments are not supported by MATHJAX.

```
59             \LWR@beginhideamsmath
60         }
61         {
62             \IfBooleanTF{#1}{
63                 \LWR@amsmathenv@@before*{#2}
64             }{
65                 \LWR@amsmathenv@@before{#2}
66             }
67         }
68     }{}
69     \addtocounter{LWR@amsmathdepth}{1}
70 }
```

\LWR@amsmathenv@@after

Embeds the environment inside a lateximage.

```
71 \newcommand*{\LWR@amsmathenv@@after}{%
72     \end{lateximage}\end{BlockClass}\LWR@startpars%
73 }
```

\LWR@amsmathenv@after   * {⟨*environment name*⟩}

* if the environment was starred. Ignored here, only used for a consistent syntax.

Embeds the environment with MATHJAX or a lateximage.

```
74 \NewDocumentCommand{\LWR@amsmathenv@after}{s m}{%
75     \ifnumequal{\value{LWR@amsmathdepth}}{1}{%
76         \ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
77         {
78             \LWR@endhideamsmath
79             \boolfalse{LWR@amsmultline}
80             \LWR@addmathjax{#2}{\the\@envbody}%
81         }
82         {\LWR@amsmathenv@@after}
```

Clear the single-use alt text:

```
83         \gdef\LWR@ThisAltText{}%
84     }{}
85     \addtocounter{LWR@amsmathdepth}{-1}
86 }
```

multline (*env.*)

```
87 \BeforeBeginEnvironment{multline}{\LWR@amsmathenv@before{multline}}
88
89 \AfterEndEnvironment{multline}{\LWR@amsmathenv@after{multline}}
```

multline* (*env.*)

```
90 \BeforeBeginEnvironment{multline*}{\LWR@amsmathenv@before*{multline*}}
91
92 \AfterEndEnvironment{multline*}{\LWR@amsmathenv@after*{multline*}}
93
```

gather (*env.*)

```
94 \BeforeBeginEnvironment{gather}{\LWR@amsmathenv@before{gather}}
95
96 \AfterEndEnvironment{gather}{\LWR@amsmathenv@after{gather}}
```

gather* (*env.*)

```
97 \BeforeBeginEnvironment{gather*}{\LWR@amsmathenv@before*{gather*}}
98
99 \AfterEndEnvironment{gather*}{\LWR@amsmathenv@after*{gather*}}
```

align (*env.*)

```
100 \BeforeBeginEnvironment{align}{\LWR@amsmathenv@before{align}}
101
102 \AfterEndEnvironment{align}{\LWR@amsmathenv@after{align}}
```

align⋆ (*env.*)

```
103 \BeforeBeginEnvironment{align*}{\LWR@amsmathenv@before*{align*}}
104
105 \AfterEndEnvironment{align*}{\LWR@amsmathenv@after*{align*}}
```

flalign (*env.*)

```
106 \BeforeBeginEnvironment{flalign}{\LWR@amsmathenv@before{flalign}}
107
108 \AfterEndEnvironment{flalign}{\LWR@amsmathenv@after{flalign}}
```

flalign⋆ (*env.*)

```
109 \BeforeBeginEnvironment{flalign*}{\LWR@amsmathenv@before*{flalign*}}
110
111 \AfterEndEnvironment{flalign*}{\LWR@amsmathenv@after*{flalign*}}
```

alignat (*env.*)

```
112 \BeforeBeginEnvironment{alignat}{\LWR@amsmathenv@before{alignat}}
113
114 \AfterEndEnvironment{alignat}{\LWR@amsmathenv@after{alignat}}
```

alignat⋆ (*env.*)

```
115 \BeforeBeginEnvironment{alignat*}{\LWR@amsmathenv@before*{alignat*}}
116
117 \AfterEndEnvironment{alignat*}{\LWR@amsmathenv@after*{alignat*}}
```

```
118 \AtBeginEnvironment{subequations}{
119     \renewcommand*{\theMathJaxsubequations}{1}
120     \renewcommand*{\theMathJaxsection}{\theparentequation}
121     \renewcommand*{\theMathJaxequation}{\arabic{equation}}
122 }
```

For MathJax:

```
123 \begin{warpMathJax}
124 \CustomizeMathJax{\newcommand{\intertext}[1]{\text{#1}\notag \\}}
125 \CustomizeMathJax{\let\Hat\hat}
126 \CustomizeMathJax{\let\Check\check}
127 \CustomizeMathJax{\let\Tilde\tilde}
128 \CustomizeMathJax{\let\Acute\acute}
129 \CustomizeMathJax{\let\Grave\grave}
130 \CustomizeMathJax{\let\Dot\dot}
131 \CustomizeMathJax{\let\Ddot\ddot}
132 \CustomizeMathJax{\let\Breve\breve}
133 \CustomizeMathJax{\let\Bar\bar}
134 \CustomizeMathJax{\let\Vec\vec}
135 \end{warpMathJax}
```

File 22 **`lwarp-amsthm.sty`**

§ 134   Package   **amsthm**

(*Emulates or patches code by* Publications Technical Group — American Mathematical Society.)

The original source code is located in `amsclass.dtx`, and printed in `amsclass.pdf`.

amsthm (*Pkg*)   amsthm is patched for use by lwarp.

Table 19: amsthm package — css styling of theorems and proofs

Theorem: `<div>` of class amsthmbody<theoremstyle>

Theorem Name: `<span>` of class amsthmname<theoremtyle>

Theorem Number: `<span>` of class amsthmnumber<theoremstyle>

Theorem Note: `<span>` of class amsthmnote<theoremstyle>

Proof: `<div>` of class amsthmproof

Proof Name: `<span>` of class amsthmproofname

where `<theoremstyle>` is plain, definition, etc.

---

for HTML output: amsthm must be loaded before mdframed:

```
1 \IfPackageLoadedTF{mdframed}{
2    \PackageError{lwarp}
3    {%
4        Package mdframed must be loaded after package amsthm.\MessageBreak
5        Enter 'H' for solutions%
6    }
7    {%
8        Move ``\protect\usepackage{amsthm}'' before
9        ``\protect\usepackage{mdframed}''.\MessageBreak
10       Package amsthm may be loaded by something else,\MessageBreak
11       which must also be moved before mdframed.%
12   }
13 }
14 {\relax}
```

Necessary for \text, used by \openbox, etc., below:

```
15 \RequirePackage{amsmath}
```

```
16 \LWR@ProvidesPackagePass{amsthm}[2017/10/31]
```

Storage for the style being used for new theorems:

```
17 \newcommand{\LWR@newtheoremstyle}{plain}
```

Patched to remember the style being used for new theorems:

```
18 \renewcommand{\theoremstyle}[1]{%
19   \@ifundefined{th@#1}{%
20     \PackageWarning{amsthm}{Unknown theoremstyle `#1'}%
21     \thm@style{plain}%
22     \renewcommand{\LWR@newtheoremstyle}{plain}% lwarp
23   }{%
24     \thm@style{#1}%
25     \renewcommand{\LWR@newtheoremstyle}{#1}% lwarp
26   }%
27 }
```

Patched to remember the style for this theorem type:

```
28 \VerifyCommand[lwarp][amsthm]{\@xnthm}{21F7FB3FB6FB0C1A0F2EECD66EE87A60}
29
30 \def\@xnthm#1#2{%
31   \csedef{LWR@thmstyle#2}{\LWR@newtheoremstyle}% lwarp
32   \let\@tempa\relax
33   \@xp\@ifdefinable\csname #2\endcsname{%
34     \global\@xp\let\csname end#2\endcsname\@endtheorem
35     \ifx *#1% unnumbered, need to get one more mandatory arg
36       \edef\@tempa##1{%
37         \gdef\@xp\@nx\csname#2\endcsname{%
38           \@nx\@thm{\@xp\@nx\csname th@\the\thm@style\endcsname}%
39             {}{##1}}}%
40     \else % numbered theorem, need to check for optional arg
41       \def\@tempa{\@oparg{\@ynthm{#2}}[]}%
42     \fi
43     \AtBeginEnvironment{#2}{%                              lwarp
44         \edef\LWR@thisthmstyle{\@nameuse{LWR@thmstyle#2}}%  lwarp
45     }%                                                    lwarp
46   }%
47   \@tempa%
48 }
```

Patched to enclose with css:

```
49 \newcommand{\LWR@haveamsthmname}{
50     \renewcommand{\thmname}[1]{%
51         \InlineClass{amsthmname\LWR@thisthmstyle}{##1}%
52     }
53 }
54
55 \newcommand{\LWR@haveamsthmnumber}{
56     \renewcommand{\thmnumber}[1]{%
57         \InlineClass{amsthmnumber\LWR@thisthmstyle}{##1}%
58     }
59 }
60
61 \newcommand{\LWR@haveamsthmnote}{
62     \renewcommand{\thmnote}[1]{%
63         \InlineClass{amsthmnote\LWR@thisthmstyle}{##1}%
64     }
65 }
66
67 \LWR@haveamsthmname
68 \LWR@haveamsthmnumber
69 \LWR@haveamsthmnote
```

Patched for CSS. Not using \VerifyCommand because the existing defintion depends on other packages. The following is from amsthm's own definition.

```
70 \def\@begintheorem#1#2[#3]{%

71     \GetTitleString{#3}%                                        lwarp
72     \let\@currentlabelname\GetTitleStringResult%                lwarp
73     \item[%

74     \LWR@newautopagelabel{page}\LWR@orignewline%

75 %    \deferred@thm@head{
76 %        \the\thm@headfont \thm@indent
77     \@ifempty{#1}{\let\thmname\@gobble}{\LWR@haveamsthmname}%        lwarp
78     \@ifempty{#2}{\let\thmnumber\@gobble}{\LWR@haveamsthmnumber}%    lwarp
79     \@ifempty{#3}{\let\thmnote\@gobble}{\LWR@haveamsthmnote}%        lwarp
80     \thm@swap\swappedhead\thmhead{#1}{#2}{#3}%
81     \the\thm@headpunct % space
82     \thmheadnl % possibly a newline.
83     \hskip\thm@headsep
84 %    }%
85     ]%
86   \ignorespaces}
```

Patched for CSS:

```
87 \VerifyCommand[lwarp][amsthm]{\@thm}{2624BDB5B96C45756978B3D393430088}
88
89 \def\@thm#1#2#3{%
90   \ifhmode\unskip\unskip\par\fi
91   \normalfont
92   \LWR@forcenewpage%                                       lwarp

93   \LWR@printpendingfootnotes%                              lwarp

94   \BlockClass{amsthmbody\LWR@thisthmstyle}%                lwarp
95   \trivlist
96   \let\thmheadnl\relax
97   \let\thm@swap\@gobble
98   \thm@notefont{\fontseries\mddefault\upshape}%
99   \thm@headpunct{.}% add period after heading
100  \thm@headsep 5\p@ plus\p@ minus\p@\relax
101  \thm@space@setup
102  #1% style overrides
103  \@topsep \thm@preskip            % used by thm head
104  \@topsepadd \thm@postskip        % used by \@endparenv
105  \def\@tempa{#2}\ifx\@empty\@tempa
106    \def\@tempa{\@oparg{\@begintheorem{#3}{}}[]}%
107  \else
108    \refstepcounter{#2}%
109    \def\@tempa{\@oparg{\@begintheorem{#3}{\csname the#2\endcsname}}[]}%
110  \fi
111  \@tempa%
112 }
```

cleveref patches \@thm to do \cref@thmoptarg if an optional argument is given. lwarp then patches \cref@thmoptarg \AtBeginDocument.

```
113 \AtBeginDocument{%
114 %
115 \VerifyCommand[lwarp][amsthm]{\cref@thmoptarg}{64B912D4D903D245FD05837C5838C9EC}
116 %
117 \def\cref@thmoptarg[#1]#2#3#4{%
118     \ifhmode\unskip\unskip\par\fi%
119     \normalfont%
120     \LWR@forcenewpage%                              lwarp

121     \LWR@printpendingfootnotes%                     lwarp

122     \BlockClass{amsthmbody\LWR@thisthmstyle}%   lwarp
123     \trivlist%
124     \let\thmheadnl\relax%
125     \let\thm@swap\@gobble%
126     \thm@notefont{\fontseries\mddefault\upshape}%
127     \thm@headpunct{.}% add period after heading
128     \thm@headsep 5\p@ plus\p@ minus\p@\relax%
129     \thm@space@setup%
130     #2% style overrides
131     \@topsep \thm@preskip                    % used by thm head
132     \@topsepadd \thm@postskip                % used by \@endparenv
133     \def\@tempa{#3}\ifx\@empty\@tempa%
134         \def\@tempa{\@oparg{\@begintheorem{#4}{}}[]}%
135     \else%
136         \refstepcounter[#1]{#3}%  <<< cleveref modification
137         \def\@tempa{\@oparg{\@begintheorem{#4}{\csname the#3\endcsname}}[]}%
138     \fi%
139     \@tempa
140 }%
141 }% AtBeginDocument
142
143 \def\@endtheorem{%
144     \endtrivlist%

145   \LWR@printpendingfootnotes%                        lwarp

146     \endBlockClass%
147     \@endpefalse%
148 }
```

Proof QED symbol:

```
149 \AtBeginDocument{
150 \@ifundefined{LWR@orig@openbox}{
151 \LetLtxMacro\LWR@orig@openbox\openbox
152 \LetLtxMacro\LWR@orig@blacksquare\blacksquare
153 \LetLtxMacro\LWR@orig@Box\Box
154
155 \def\openbox{\text{\HTMLunicode{25A1}}}% UTF-8 white box
156 \def\blacksquare{\text{\HTMLunicode{220E}}}% UTF-8 end-of-proof
157 \def\Box{\text{\HTMLunicode{25A1}}}% UTF-8 white box
158
159 \appto\LWR@restoreorigformatting{%
160     \LetLtxMacro\openbox\LWR@orig@openbox%
161     \LetLtxMacro\blacksquare\LWR@orig@blacksquare%
162     \LetLtxMacro\Box\LWR@orig@Box%
163 }% appto
```

```
164 }{}% @ifundefined
165 }% AtBeginDocument
```

Patched to add a <span>:

```
166 \DeclareRobustCommand{\qed}{%
167   \ifmmode \mathqed
168   \else
169 %     \leavevmode\unskip\penalty9999 \hbox{}\nobreak\hfill
170 %     \quad\hbox{\qedsymbol}%
171         \InlineClass{theoremendmark}{\qedsymbol}%   lwarp
172   \fi
173 }
```

Patched for css:

```
174 \renewenvironment{proof}[1][\proofname]{\par
175   \LWR@forcenewpage% lwarp


176   \LWR@printpendingfootnotes%                    lwarp


177     \BlockClass{amsthmproof}% lwarp
178     \LWR@newautopagelabel{page}%
179   \pushQED{\qed}%
180   \normalfont \topsep6\p@\@plus6\p@\relax
181   \trivlist
182   \item[
183       \InlineClass{amsthmproofname}{#1\@addpunct{.}}]\ignorespaces% changes
184 }{%
185   \popQED\endtrivlist%


186   \LWR@printpendingfootnotes%                lwarp


187   \endBlockClass% lwarp
188   \@endpefalse
189 }
```

File 23 **`lwarp-anonchap.sty`**

§ 135   Package   **anonchap**

(*Emulates or patches code by* PETER WILSON.)

anonchap (*Pkg*)   anonchap is emulated.

tocloft (*Pkg*)   If using tocloft with tocbibind, anonchap, fncychap, or other packages which
⚠ tocloft & other packages   change chapter title formatting, load tocloft with its titles option, which tells
tocloft to use standard LaTeX commands to create the titles, allowing other pack-
ages to work with it.

The code is shared by tocbibind.

for HTML output: 1 \LWR@ProvidesPackageDrop{anonchap}[2009/08/03]

```
2 \newcommand{\simplechapter}[1][\@empty]{%
3    \def\@chapcntformat##1{%
4        #1~\csname the##1\endcsname\simplechapterdelim\quad%
5    }%
6 }
7
8 \newcommand{\restorechapter}{%
9 \let\@chapcntformat\@seccntformat%
10 }
```

File 24 **`lwarp-anysize.sty`**

§ 136   Package **anysize**

(*Emulates or patches code by* MICHAEL SALZENBERG, THOMAS ESSER.)

anysize (*Pkg*)   anysize is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{anysize}[1994/08/13]

```
2 \def\papersize#1#2{}
3 \def\marginsize#1#2#3#4{}
```

File 25 **`lwarp-appendix.sty`**

§ 137   Package **appendix**

(*Emulates or patches code by* PETER WILSON.)

appendix (*Pkg*)   appendix is patched for use by lwarp.

⚠ incorrect TOC link   During HTML conversion, the option toc without the option page results in a TOC link to whichever section was before the appendices environment. It is recommended to use both toc and also page at the same time.

for HTML output: 1 \LWR@ProvidesPackagePass{appendix}[2009/09/02]

```
2 \renewcommand*{\@chap@pppage}{%
3 \part*{\appendixpagename}
4 \if@dotoc@pp
5 \addappheadtotoc
6 \fi
7 }
8
9 \renewcommand*{\@sec@pppage}{%
10 \part*{\appendixpagename}
11 \if@dotoc@pp
12 \addappheadtotoc
13 \fi
14 }
```

File 26 **`lwarp-apxproof.sty`**

§ 138   Package **apxproof**

(*Emulates or patches code by* PIERRE SENELLART.)

apxproof (*Pkg*)   apxproof is patched for use by lwarp.

for HTML output: 1 `\LWR@ProvidesPackagePass{apxproof}[2022/10/14]`

2 `\VerifyCommand[lwarp][apxproof]{\FVB@axp@VerbatimOut}{ADA4853FD25696EB39CD005CF44C7B5C}`
3
4 `\xpatchcmd{\FVB@axp@VerbatimOut}`
5 `    {\FV@Scan}`
6 `    {\boolfalse{LWR@HTMLsanitize@tmpb@enable}\FV@Scan}`
7 `    {}`
8 `    {\LWR@patcherror{apxproof}{FVB@axp@VerbatimOut}}`

File 27   **`lwarp-ar.sty`**

§ 139   Package **ar**

(*Emulates or patches code by* AGOSTINO DE MARCO.)

ar (*Pkg*)   ar is patched for use by lwarp.

for HTML output: 1 `\LWR@ProvidesPackagePass{ar}[2012/01/23]`

Measure and print the width of the supplied glyph.

2 `\newlength{\LWR@ar@width}`
3
4 `\newcommand*{\LWR@ar@printwidth}[1]{%`
5 `    \setlength{\LWR@ar@width}{\widthof{#1}}%`
6 `    width:%`
7 `    \LWR@convertto{em}{\the\LWR@ar@width}em%`
8 `}`

The HTML version of \AR:

9 `\newrobustcmd*{\LWR@HTML@AR}{%`

Start a hashed lateximage, additionally hashed by the font series, with a width depending on the given glyph:

10 `    \begin{lateximage}*[AR][\LWR@f@series][\LWR@ar@printwidth{\LWR@print@AR}]%`

For text mode, set the font series according to the HTML font series:

11 `    \ifmmode\else\csuse{LWR@orig\LWR@f@series series}\fi%`

Print the original glyph using the newly set font series:

```
12      \LWR@print@AR%
```

Done.

```
13      \end{lateximage}%
14 }
```

Combine the print and HTML versions:

```
15 \LWR@formatted{AR}
```

```
16 \newrobustcmd*{\LWR@HTML@ARb}{%
17      \begin{lateximage}*[AR][b][\LWR@ar@printwidth{\LWR@print@ARb}]%
18      \LWR@print@ARb%
19      \end{lateximage}%
20 }
21 \LWR@formatted{ARb}
```

```
22 \newrobustcmd*{\LWR@HTML@ARss}{%
23   \begin{lateximage}*[ARss][\LWR@f@series][\LWR@ar@printwidth{\LWR@print@ARss}]%
24   \ifmmode\else\csuse{LWR@orig\LWR@f@series series}\fi%
25   \LWR@print@ARss%
26   \end{lateximage}%
27 }
28 \LWR@formatted{ARss}
```

```
29 \newrobustcmd*{\LWR@HTML@ARssb}{%
30      \begin{lateximage}*[AR][ssb][\LWR@ar@printwidth{\LWR@print@ARssb}]%
31      \LWR@print@ARssb%
32      \end{lateximage}%
33 }
34 \LWR@formatted{ARssb}
```

```
35 \newrobustcmd*{\LWR@HTML@ARtt}{%
36      \begin{lateximage}*[AR][tt][\LWR@ar@printwidth{\LWR@print@ARtt}]%
37      \LWR@print@ARtt%
38      \end{lateximage}%
39 }
40 \LWR@formatted{ARtt}
```

For MATHJAX:

```
41 \begin{warpMathJax}
42 \CustomizeMathJax{\newcommand{\AR}{\mathit{A\!\!R}}}
43 \CustomizeMathJax{\newcommand{\ARb}{\boldsymbol{A\!\!R}}}
44 \end{warpMathJax}
```

File 28   **`lwarp-arabicfront.sty`**

§ 140   Package   # arabicfront

arabicfront (*Pkg*)   arabicfront is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{arabicfront}[2006/09/03]`

File 29 **`lwarp-array.sty`**

§ 141 Package **array**

array (*Pkg*) array is used as-is for print output, and emulated for HTML.

plarray and plextarray do not affect \firsthline or \lasthline, and so are not affected by the following.

for HTML output: If array is not yet loaded, remove the default nullfied macros:

```
1 \IfPackageLoadedTF{array}{}{%
2     \let\firsthline\relax
3     \let\lasthline\relax
4 }
5
6 \LWR@ProvidesPackagePass{array}[2018/12/30]
```

Provide simplified column types for HTML:

```
7 \HTMLnewcolumntype{w}[2]{#1}
8 \HTMLnewcolumntype{W}[2]{#1}
```

More HTML versions:

```
 9 \newcommand*{\LWR@HTML@firsthline}{\LWR@HTMLhline}%
10 \LWR@expandableformatted{firsthline}
11
12 \newcommand*{\LWR@HTML@lasthline}{\LWR@HTMLhline}%
13 \LWR@expandableformatted{lasthline}
```

```
14 \let\tabularnewline\\
15 \providecommand*{\LWR@HTML@tabularnewline}{\LWR@tabularendofline}
16 \LWR@formatted{tabularnewline}
```

For MATHJAX:

```
17 \CustomizeMathJax{
18     \newcommand{\multicolumn}[3]{#3}% only uses one cell
19 }
```

File 30 **`lwarp-arydshln.sty`**

§ 142 Package **arydshln**

(*Emulates or patches code by* HIROSHI NAKASHIMA.)

arydshln (*Pkg*) arydshln heavily patches tabular code, so the actual package is not used. arydshln is emulated for HTML tabular, and reverts to solid rules for SVG math array and tabular in a lateximage.

css is not able to display a double-dashed border, so a single-dashed rule is displayed as a single-dashed border, and a double-dashed rule is displayed as a thicker single-dashed border.

For MATHJAX, limited emulation is provided for math mode.

array is required to allow \newcolumn below.

```
1 \RequirePackage{array}
```

```
2 \LWR@ProvidesPackageDrop{arydshln}[2018/09/26]
```

Ignored, but included for source compatibility:

```
3 \newdimen\dashlinedash \dashlinedash4pt %
4 \newdimen\dashlinegap \dashlinegap4pt %
5 \let\hdashlinewidth\dashlinedash
6 \let\hdashlinegap\dashlinegap
7
8 \def\ADLnullwide{}
9 \def\ADLsomewide{}
10 \def\ADLnullwidehline{}
11 \def\ADLsomewidehline{}
12
13 \def\ADLactivate{}
14 \def\ADLinactivate{}
15 \newcommand*{\ADLdrawingmode}[1]{}
16 \newcommand*{\ADLnoshorthanded}{}
17 \newcommand*{\dashgapcolor}[2][]{}
18 \newcommand*{\nodashgapcolor}{}
```

In a lateximage, revert to solid vertical rules:

```
19 \appto\LWR@restoreorigformatting{%
20 \newcolumntype{:}{|}%
21 \newcolumntype{;}[1]{|}%
22 \LetLtxMacro\hdashline\hline%
23 }
```

Some of these macros are already defined as temporary placeholders in the lwarp core, so they must be redefined here.

The emulated defaults also work for an emulated print mode inside a lateximage:

```
24 \def\hdashline{
25 %        \adl@hdashline\adl@ihdashline
26      \adl@hdashline\adl@inactivehdl
27 }
28 \def\adl@hdashline#1{\noalign{\ifnum0=`}\fi
29 %        \ifadl@zwhrule \vskip-\arrayrulewidth
30 %        \else
31 %            \adl@hline\adl@connect\arrayrulewidth
32              \hrule \@height \arrayrulewidth% lwarp
33 %        \fi
34        \@ifnextchar[%]
35                  {#1}%
36                  {#1[%
37 %                    \dashlinedash/\dashlinegap
38                    1pt/1pt
```

```
39                        ]}}}
40 % \def\adl@ihdashline[#1/#2]{\ifnum0=`{\fi}%
41 %        \multispan{\adl@columns}\unskip \adl@hcline\z@[#1/#2]%
42 %        \noalign{\ifnum0=`}\fi
43 %        \futurelet\@tempa\adl@xhline}
44 \def\adl@inactivehdl[#1/#2]{
45 %        \ifadl@zwhrule \vskip-\arrayrulewidth \fi
46        \hrule\@height\arrayrulewidth
47        \futurelet\@tempa\adl@xhline}
48 \def\adl@xhline{\ifx\@tempa\hline \adl@ixhline\fi
49        \ifx\@tempa\hdashline \adl@ixhline\fi
50        \ifnum0=`{\fi}}
51 \def\adl@ixhline{\vskip\doublerulesep \adl@hline\relax\doublerulesep}
52 \def\adl@hline#1#2{%
53 % \@tempcnta#2
54 %        \global\advance\adl@totalheight\@tempcnta
55 %        \xdef\adl@rowsL{\adl@rowsL
56 %                (#1/\number\@tempcnta);}%
57 %        \xdef\adl@rowsR{\adl@rowsR
58 %                (#1/\number\@tempcnta);}
59 }
60
61 \def\cdashline#1{\noalign{\ifnum0=`}\fi
62        \@ifnextchar[%]
63 %                {\adl@cdline[#1]}%
64 %                {\adl@cdline[#1][\dashlinedash/\dashlinegap]}
65                {\adl@inactivecdl[#1]}%
66                {\adl@inactivecdl[#1][\dashlinedash/\dashlinegap]}
67 }
68
69 \def\adl@inactivecdl[#1-#2][#3]{\ifnum0=`{\fi}\cline{#1-#2}}


70 \begin{warpMathJax}
71 \CustomizeMathJax{\newcommand{\firsthdashline}[1][]{\hdashline}}
72 \CustomizeMathJax{\let\lasthdashline\firsthdashline}
73 \CustomizeMathJax{\let\cdashline\cline}
74 \end{warpMathJax}
```

File 31 **lwarp-asymptote.sty**

§ 143   Package   # asymptote

(*Emulates or patches code by* ANDY HAMMERLINDL, JOHN BOWMAN, TOM PRINCE.)

asymptote (*Pkg*)   asymptote is patched for use by lwarp.

To compile:

```
pdflatex project.tex
asy project-*.asy
pdflatex project.tex

lwarpmk print
asy project-*.asy
lwarpmk print1
lwarpmk print1

lwarpmk html
asy project_html-*.asy
lwarpmk html1
lwarpmk html1
lwarpmk limages
```

for HTML output: 1 `\LWR@ProvidesPackagePass{asymptote}[2016/11/26]`

2 `\BeforeBeginEnvironment{asy}{%`
3 `    \begin{lateximage}[-asymptote-~\PackageDiagramAltText]?%`
4 `}`
5 `\AfterEndEnvironment{asy}{\end{lateximage}}`
6
7 `\VerifyCommand[lwarp][asymptote]{\asyinclude}{A4F9DF668FC457768E7DFB83FAF7B343}`
8
9 `\xpatchcmd{\asyinclude}`
10 `    {\begingroup}`
11 `    {\begin{lateximage}[-asymptote-~\PackageDiagramAltText]?}`
12 `    {}`
13 `    {\LWR@patcherror{asymptote}{asyinclude-begingroup}}`
14
15 `\xpatchcmd{\asyinclude}`
16 `    {\endgroup}`
17 `    {\end{lateximage}}`
18 `    {}`
19 `    {\LWR@patcherror{asymptote}{asyinclude-endgroup}}`

File 32 **`lwarp-atbegshi.sty`**

§ 144   Package   # atbegshi

(*Emulates or patches code by* HEIKO OBERDIEK.)

atbegshi (*Pkg*)   atbegshi is ignored.

for HTML output: Discard all options for lwarp-atbegshi:

1 `\LWR@ProvidesPackageDrop{atbegshi}[2011/10/05]`

2 `\let\AtBeginShipout\relax`
3 `\let\AtBeginShipoutNext\relax`
4 `\let\AtBeginShipoutFirst\relax`
5 `\let\AtBeginShipoutDiscard\relax`
6 `\let\AtBeginShipoutInit\relax`
7 `\let\AtBeginShipoutAddToBox\relax`

```
8 \let\AtBeginShipoutAddToBoxForeground\relax
9 \let\AtBeginShipoutUpperLeft\relax
10 \let\AtBeginShipoutUpperLeftForeground\relax
11 \let\AtBeginShipoutOriginalShipout\relax
12
13 \newcommand*{\AtBeginShipout}[1]{}
14 \newbox\AtBeginShipoutBox
15 \newcommand*{\AtBeginShipoutNext}[1]{}
16 \newcommand*{\AtBeginShipoutFirst}[1]{}
17 \newcommand*{\AtBeginShipoutDiscard}{}
18 \newcommand*{\AtBeginShipoutInit}{}
19 \newcommand*{\AtBeginShipoutAddToBox}[1]{}
20 \newcommand*{\AtBeginShipoutAddToBoxForeground}[1]{}
21 \newcommand*{\AtBeginShipoutUpperLeft}[1]{}
22 \newcommand*{\AtBeginShipoutUpperLeftForeground}[1]{}
23 \newcommand*{\AtBeginShipoutOriginalShipout}[1]{}
24 \def\AtBeginShipoutBoxWidth{0pt}
25 \def\AtBeginShipoutBoxHeight{0pt}
26 \def\AtBeginShipoutBoxDepth{0pt}
```

File 33   **`lwarp-attachfile.sty`**

§ 145   Package   **attachfile**

(*Emulates or patches code by* SCOTT PAKIN.)

attachfile (*Pkg*)   attachfile is patched for use by lwarp.

⚠   Metadata is ignored for now.

for HTML output: 1 \LWR@ProvidesPackagePass{attachfile}[2016/09/18]

Encloses each icon:

```
2 \newenvironment*{LWR@attachfile@icon}
3 {
4     \begin{lateximage}*%
5         [-attachfile-]%
6         [%
7             \detokenize\expandafter{\atfi@icon@icon}-%
8             \detokenize\expandafter{\atfi@color@rgb}%
9         ]%
10 }
11 {
12     \end{lateximage}
13 }
```

Each icon is enclosed inside a LWR@attachfile@icon environment:

```
14 \xpretocmd{\atfi@acroGraph}{\LWR@attachfile@icon}{}{}
15 \xapptocmd{\atfi@acroGraph}{\endLWR@attachfile@icon}{}{}
16
17 \xpretocmd{\atfi@acroPaperclip}{\LWR@attachfile@icon}{}{}
18 \xapptocmd{\atfi@acroPaperclip}{\endLWR@attachfile@icon}{}{}
19
20 \xpretocmd{\atfi@acroPushPin}{\LWR@attachfile@icon}{}{}
21 \xapptocmd{\atfi@acroPushPin}{\endLWR@attachfile@icon}{}{}
```

```
22
23 \xpretocmd{\atfi@acroTag}{\LWR@attachfile@icon}{}{}
24 \xapptocmd{\atfi@acroTag}{\endLWR@attachfile@icon}{}{}
```

Disable PDF file embedding:

```
25 \DeclareRobustCommand{\atfi@embedfile}[1]{}
```

The displayed output for an \attachfile reference:

```
26 \newcommand*{\LWR@attachfile@appearance}{}
27
28 \DeclareRobustCommand{\atfi@set@appearance}[1]{%
29    \def\LWR@attachfile@appearance{#1}%
30 }
```

A file annotation becomes a reference:

```
31 \DeclareRobustCommand{\atfi@insert@file@annot}[1]{%
32    \LWR@href@partsanitized{#1}{\LWR@attachfile@appearance}%
33 }
```

File 34   **`lwarp-attachfile2.sty`**

§ 146   Package   **attachfile2**

(*Emulates or patches code by* HEIKO OBERDIEK.)

attachfile2 (*Pkg*)   attachfile2 is patched for use by lwarp.

⚠   Metadata is ignored for now.

for HTML output: 1 \LWR@ProvidesPackagePass{attachfile2}[2016/05/16]

Adds memory of the selected color:

```
2 \def\LWR@attachfiletwo@color{}%
3
4 \define@key{AtFi}{color}{%
5    \def\LWR@attachfiletwo@color{#1}%    lwarp
6  \HyColor@AttachfileColor{#1}%
7          \atfi@color@tex\atfi@color@inline\atfi@color@annot
8          {attachfile2}{color}%
9 }
```

Encloses each icon:

```
10 \newenvironment*{LWR@attachfile@icon}
11 {
12    \begin{lateximage}*%
13        [-attachfile-]%
14        [%
15            \detokenize\expandafter{\atfi@icon@icon}-%
16            \detokenize\expandafter{\LWR@attachfiletwo@color}%
17        ]%
18 }
```

```
19 {
20     \end{lateximage}
21 }
```

Each icon is enclosed inside a `LWR@attachfile@icon` environment:

```
22 \xpretocmd{\atfi@acroGraph}{\LWR@attachfile@icon}{}{}
23 \xapptocmd{\atfi@acroGraph}{\endLWR@attachfile@icon}{}{}
24
25 \xpretocmd{\atfi@acroPaperclip}{\LWR@attachfile@icon}{}{}
26 \xapptocmd{\atfi@acroPaperclip}{\endLWR@attachfile@icon}{}{}
27
28 \xpretocmd{\atfi@acroPushPin}{\LWR@attachfile@icon}{}{}
29 \xapptocmd{\atfi@acroPushPin}{\endLWR@attachfile@icon}{}{}
30
31 \xpretocmd{\atfi@acroTag}{\LWR@attachfile@icon}{}{}
32 \xapptocmd{\atfi@acroTag}{\endLWR@attachfile@icon}{}{}
```

Disable PDF file embedding:

```
33 \DeclareRobustCommand{\atfi@embedfile}[1]{}
```

The displayed output for an `\attachfile` reference:

```
34 \newcommand*{\LWR@attachfile@appearance}{}
35
36 \def\atfi@set@appearance@icon{%
37     \atfi@set@appearance{\csname atfi@acro\atfi@icon@icon\endcsname}%
38 }
39
40 \DeclareRobustCommand{\atfi@set@appearance}[1]{%
41     \def\LWR@attachfile@appearance{#1}%
42 }
```

A file annotation becomes a reference:

```
43 \DeclareRobustCommand{\atfi@insert@file@annot}[1]{%
44     \LWR@href@partsanitized{#1}{\LWR@attachfile@appearance}%
45 }
```

Modified for text color:

```
46 \VerifyCommand[lwarp][attachfile2]{\notextattachfile}{CE78259EFC576D4A15920EADF824D7EF}
47
48 \DeclareRobustCommand{\notextattachfile}[2][]{%
49   \begingroup
50     \atfi@setup{#1}%
51     \ifatfi@print
52       \leavevmode
53       \begingroup
54         \HyColor@UseColor\atfi@color@tex
55         \LWR@textcurrentcolor{#2}%        lwarp
56 % \strut
57       \endgroup
58 %    \else
59 %      \sbox\ltx@zero{#2\strut}%
60 %      \makebox[\wd0]{}%
61     \fi
62   \endgroup
```

63 `}`

Modified to draw the icon:

64 `\VerifyCommand[lwarp][attachfile2]{\noattachfile}{CE78259EFC576D4A15920EADF824D7EF}`
65
66 `\DeclareRobustCommand{\noattachfile}[1][]{%`
67 `  \begingroup`
68 `    \atfi@setup{#1}%`
69 `    \atfi@set@appearance@icon`
70 `    \ifatfi@print`
71 `      \LWR@attachfile@appearance%      lwarp`
72 `%      \expandafter`
73 `%      \atfi@refxform\csname atfi@appobj@\atfi@icon@icon\endcsname`
74 `%    \else`
75 `%      \makebox[\atfi@appearancewidth]{}%`
76 `    \fi`
77 `  \endgroup`
78 `}`

File 35 **`lwarp-authblk.sty`**

§ 147 Package # authblk

(*Emulates or patches code by* Patrick W. Daly.)

authblk (*Pkg*) authblk is patched for HTML.

package support lwarp supports the native LaTeX titling commands, and also supports the packages
⚠ load order authblk and titling. If both are used, authblk should be loaded before titling.

\published and \subtitle If using the titling package, additional titlepage fields for \published and \subtitle
may be added by using \AddSubtitlePublished in the preamble. See section 71.8.

(*Emulates or patches code by* Patrick W. Daly.)

for HTML output: Require that authblk be loaded before titling:

1 `\IfPackageLoadedTF{titling}{`
2 `    \PackageError{lwarp-authblk}`
3 `        {Package authblk must be loaded before titling}`
4 `        {%`
5 `            Titling appends authblk's author macro,`
6 `            so authblk must be loaded first.%`
7 `        }`
8 `}`
9 `{\relax}`

Load authblk:

10 `\LWR@ProvidesPackagePass{authblk}[2001/02/27]`

Patch to add a class for the affiliation:

11 `\LetLtxMacro\LWRAB@affil\affil`
12
13 `\renewcommand{\affil}[2][]{%`

14 \LWRAB@affil[#1]{\protect\InlineClass{affiliation}{#2}}
15 }

Create an HTML break for an \authorcr:

16 \renewcommand*{\authorcr}{\protect\LWR@newlinebr}

File 36   **`lwarp-autobreak.sty`**

§ 148   Package   **autobreak**

(*Emulates or patches code by* TAKAHIRO UEDA.)

autobreak (*Pkg*)   **autobreak** is used as-is for SVG math, and nullified for MATHJAX.

for HTML output: 1 \LWR@ProvidesPackagePass{autobreak}[2017/02/23]

For MATHJAX. The modified align environment is used for SVG math, but is reverted to its original for MATHJAX. (Extraneous commas were appearing in the result.)

2 \begin{warpMathJax}
3 \renewenvironment{autobreak}{\newcommand{\MoveEqLeft}[1]{}}{}
4 \let\start@align\@autobreak@oldstart@align
5 \let\endalign\@autobreak@oldendalign
6 \CustomizeMathJax{\newenvironment{autobreak}{}{}}
7 \CustomizeMathJax{\newcommand{\MoveEqLeft}[1][]{}}
8 \CustomizeMathJax{\newcommand{\everybeforeautobreak}[1]{}}
9 \CustomizeMathJax{\newcommand{\everyafterautobreak}[1]{}}
10 \end{warpMathJax}

File 37   **`lwarp-autonum.sty`**

§ 149   Package   **autonum**

autonum (*Pkg*)   **autonum** is ignored.

⚠   numbering, +   All equations are numbered in HTML output. MATHJAX does not support the "+" environments.

for HTML output: 1 \LWR@ProvidesPackageDrop{autonum}[2015/01/18]

2 \RequirePackage{amsmath}
3
4
5 \newenvironment{equation+}{\equation}{\endequation}
6
7
8 \newenvironment{gather+}{\gather}{\endgather}
9
10 \BeforeBeginEnvironment{gather+}{\LWR@amsmathenv@@before{gather+}}
11
12 \AfterEndEnvironment{gather+}{\LWR@amsmathenv@@after}
13

14
15 \newenvironment{multline+}{\multline}{\endmultline}
16
17 \BeforeBeginEnvironment{multline+}{\LWR@amsmathenv@@before{multline+}}
18
19 \AfterEndEnvironment{multline+}{\LWR@amsmathenv@@after}

20 \newenvironment{flalign+}{\flalign}{\endflalign}
21
22 \BeforeBeginEnvironment{flalign+}{\LWR@amsmathenv@@before{flalign+}}
23
24 \AfterEndEnvironment{flalign+}{\LWR@amsmathenv@@after}
25
26
27 \newenvironment{align+}{\align}{\endalign}
28
29 \BeforeBeginEnvironment{align+}{\LWR@amsmathenv@@before{aline+}}
30
31 \AfterEndEnvironment{align+}{\LWR@amsmathenv@@after}
32
33
34 \newenvironment{alignat+}{\alignat}{\endalignat}
35
36 \BeforeBeginEnvironment{alignat+}{\LWR@amsmathenv@@before{alineat+}}
37
38 \AfterEndEnvironment{alignat+}{\LWR@amsmathenv@@after}
39
40
41 \newenvironment{split+}{\split}{\endsplit}

File 38   **`lwarp-awesomebox.sty`**

§ 150   Package   # awesomebox

(*Emulates or patches code by* Étienne Deparis.)

awesomebox (*Pkg*)   awesomebox is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{awesomebox}[2019/07/27]

2 \newcommand*{\LWR@awesomebox@boxborders}{}%
3 \newcommand*{\LWR@awesomebox@contentsborders}{}%
4
5 \newcommand*{\LWR@awesomebox@ruleborders}{%
6   border-top: 1px solid black ;
7   border-bottom: 1px solid black%
8 }
9
10 % \awesomebox[1:vrulecolor][2:hrule][3:title]{4:vrulewidth}{5:icon}{6:iconcolor}{7:content}
11 \RenewDocumentCommand \awesomebox { O{abvrulecolor} O{} o m m m +m }{%
12   \begin{awesomeblock}[#1][#2][#3]{#4}{#5}{#6}
13   #7
14   \end{awesomeblock}
15 }
16
17 % \begin{awesomeblock}[1:vrulecolor][2:hrule][3:title]{4:vrulewidth}{5:icon}{6:iconcolor}

```
18 % <contents>
19 % \end{awesomeblock}
20 \RenewDocumentEnvironment{awesomeblock}{ O{abvrulecolor} O{} o m m m }
21 {%
22     \LWR@forceminwidth{#4}%
23     \convertcolorspec{named}{#1}{HTML}\LWR@tempcolor%
24     \renewcommand*{\LWR@awesomebox@boxborders}{}%
25     \renewcommand*{\LWR@awesomebox@contentsborders}{}%
26     \ifdefstrequal{\abShortLine}{#2}{%
27      \renewcommand*{\LWR@awesomebox@contentsborders}{\LWR@awesomebox@ruleborders}%
28     }{}%
29     \ifdefstrequal{\abLongLine}{#2}{%
30      \renewcommand*{\LWR@awesomebox@boxborders}{\LWR@awesomebox@ruleborders}%
31     }{}%
32     \begin{BlockClass}[\LWR@awesomebox@boxborders]{awesomebox}
33     \begin{BlockClass}[%
34         margin-left: 2\% ;
35         vertical-align: top
36     ]{minipage}
37         \color{#6}\Huge #5
38     \end{BlockClass}
39     \begin{BlockClass}[%
40         width:75\% ;
41         vertical-align: top ;
42         padding-left: 1em ;
43         \LWR@awesomebox@contentsborders ;
44         border-left: \LWR@printlength{\LWR@atleastonept} %
45             solid \LWR@origpound\LWR@tempcolor%
46     ]{minipage}
47         \IfValueTF{#3}{#3\newline}{}
48 }
49 {%
50     \end{BlockClass}
51     \end{BlockClass}
52 }
```

File 39 **`lwarp-axessibility.sty`**

§ 151   Package   **axessibility**

axessibility (*Pkg*)   axessibility is ignored.

for HTML output:
```
1 \PackageInfo{lwarp}{Using the lwarp version of package `axessibility'.}%
2 \ProvidesPackage{lwarp-axessibility}% no date is declared by the original
3
4 \newif\iftagpdfopt
5
6 \DeclareOption{accsupp}{
7   \tagpdfoptfalse
8 }
9
10 \DeclareOption{tagpdf}{
11   \tagpdfopttrue
12 }
13
14 \ProcessOptions\relax
15
```

```
16 \iftagpdfopt
17    \RequirePackage{tagpdf}
18 \else
19    \RequirePackage{accsupp}
20 \fi
```

```
21 \long\def\wrap#1{}
22 \long\def\wrapml#1{}
23 \long\def\wrapmlstar#1{}
24 \long\def\wrapmlalt#1{}
```

For MathJax. These usually will not be needed.

```
25 \begin{warpMathJax}
26 \CustomizeMathJax{\newcommand{\wrap}[1]{}}
27 \CustomizeMathJax{\newcommand{\wrapml}[1]{}}
28 \CustomizeMathJax{\newcommand{\wrapmlstar}[1]{}}
29 \CustomizeMathJax{\newcommand{\wrapmlalt}[1]{}}
30 \end{warpMathJax}
```

File 40   `lwarp-axodraw2.sty`

§ 152   Package   # axodraw2

(*Emulates or patches code by* JOHN C. COLLINS, J.A.M. VERMASEREN.)

axodraw2 (*Pkg*)   axodraw2 is patched for use by lwarp.

for HTML output: `1 \LWR@ProvidesPackagePass{axodraw2}[2018/02/15]`

```
2 \BeforeBeginEnvironment{axopicture}{%
3    \begin{lateximage}[-axopicture-~\PackageDiagramAltText]?%
4 }
5
6 \AfterEndEnvironment{axopicture}{\end{lateximage}}
```

File 41   `lwarp-backnaur.sty`

§ 153   Package   # backnaur

(*Emulates or patches code by* ADRIAN P. ROBSON.)

backnaur (*Pkg*)   backnaur is patched for use by lwarp, and emulated for MathJax.

for HTML output: `1 \LWR@ProvidesPackagePass{backnaur}[2019/06/18]`

```
2 \renewenvironment{bnf}{\eqnarray}{\endeqnarray}
3 \renewenvironment{bnf*}{\csuse{eqnarray*}}{\csuse{endeqnarray*}}
```

For MathJax:

```
4 \begin{warpMathJax}
5 \CustomizeMathJax{\newcommand{\bnfpn}[1]{\langle \text{\textrm{#1}} \rangle}}
```

```
6 \CustomizeMathJax{\newcommand{\bnfor}{\; \mid \;}}
7 \CustomizeMathJax{\newcommand{\bnfsp}{\;}}
8 \IfPackageLoadedWithOptionsTF{backnaur}{perp}{
9    \CustomizeMathJax{\newcommand{\bnfes}{\perp}}
10 }{
11    \IfPackageLoadedWithOptionsTF{backnaur}{epsilon}{
12        \CustomizeMathJax{\newcommand{\bnfes}{\epsilon}}
13    }{
14        \CustomizeMathJax{\newcommand{\bnfes}{\lambda}}
15    }
16 }
17 \IfPackageLoadedWithOptionsTF{backnaur}{tsrm}{
18    \CustomizeMathJax{\newcommand{\bnfts}[1]{\text{#1}}}
19 }{
20    \CustomizeMathJax{\newcommand{\bnfts}[1]{\text{\texttt{#1}}}}
21 }
22 \CustomizeMathJax{\newcommand{\bnftd}[1]{\text{\textit{#1}}}}
23 \CustomizeMathJax{\newcommand{\bnfsk}{\dots}}
24 \IfPackageLoadedWithOptionsTF{backnaur}{altpo}{
25    \CustomizeMathJax{\newcommand{\bnfpo}{::=}}
26 }{
27    \CustomizeMathJax{\newcommand{\bnfpo}{\models}}
28 }
29 \CustomizeMathJax{\newcommand{\bnfprod}{\ifstar{\LWRbnfprodnn}{\LWRbnfprodyn}}}
30 \CustomizeMathJax{\newcommand{\LWRbnfprodyn}[2]{\bnfpn{#1} & \bnfpo & #2}}
31 \CustomizeMathJax{\newcommand{\LWRbnfprodnn}[2]{\nonumber \bnfpn{#1} & \bnfpo & #2}}
32 \CustomizeMathJax{\newcommand{\bnfmore}{\ifstar{\LWRbnfmorenn}{\LWRbnfmoreyn}}}
33 \CustomizeMathJax{\newcommand{\LWRbnfmoreyn}[1]{ &  & #1}}
34 \CustomizeMathJax{\newcommand{\LWRbnfmorenn}[1]{\nonumber &  & #1}}
35 \end{warpMathJax}
```

File 42 **`lwarp-backref.sty`**

§ 154   Package   # backref

(*Emulates or patches code by* David Carlisle and Sebastian Rahtz.)

backref (*Pkg*)   backref is patched for use by lwarp.

for HTML output: `1 \LWR@ProvidesPackagePass{backref}[2023-11-26]`

Force the hyperref option:

```
2 \def\backref{}
3
4 \long\def\hyper@section@backref#1#2#3{\LWR@refwithsection{#3}}
5
6 \let\backrefxxx\hyper@section@backref
7
8 \VerifyCommand[lwarp][backref]{\Hy@backout}
9    {277444C1A7F4620D9563D82EFD29877E}
10
11 \xpatchcmd{\Hy@backout}
12    {\@currentHref}%
13    {\BaseJobname-autopage-\arabic{LWR@previousautopagelabel}}
14    {}
15    {\LWR@patcherror{backref}{Hy@backout}}
```

File 43   **lwarp-balance.sty**

§ 155   Package   # balance

(*Emulates or patches code by* PATRICK W. DALY.)

balance (*Pkg*)   balance is ignored.

for HTML output: Discard all options for lwarp-balance:

```
1 \LWR@ProvidesPackageDrop{balance}[1999/02/23]
```

```
2 \newcommand*{\balance}{}
3 \newcommand*{\nobalance}{}
```

File 44   **lwarp-bbding.sty**

§ 156   Package   # bbding

(*Emulates or patches code by* KAREL HORAK, PETER MØLLER NEERGAARD.)

bbding (*Pkg*)   bbding is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{bbding}[1999/04/15]

```
2 \newcommand*{\LWR@bbdingsymbol}[2]{\HTMLunicode{#2}}
3
4 \newcommand{\LWR@HTML@ScissorRightBrokenBottom}{\LWR@bbdingsymbol{000}  {2701}}
5 \newcommand{\LWR@HTML@ScissorRight}{\LWR@bbdingsymbol{001}           {2702}}
6 \newcommand{\LWR@HTML@ScissorRightBrokenTop}{\LWR@bbdingsymbol{002}    {2703}}
7 \newcommand{\LWR@HTML@ScissorLeftBrokenBottom}{\LWR@bbdingsymbol{003}   {2701}}
8 \newcommand{\LWR@HTML@ScissorLeft}{\LWR@bbdingsymbol{004}            {2702}}
9 \newcommand{\LWR@HTML@ScissorLeftBrokenTop}{\LWR@bbdingsymbol{005}     {2703}}
10 \newcommand{\LWR@HTML@ScissorHollowRight}{\LWR@bbdingsymbol{006}      {2704}}
11 \newcommand{\LWR@HTML@ScissorHollowLeft}{\LWR@bbdingsymbol{007}       {2704}}
12 \newcommand{\LWR@HTML@Phone}{\LWR@bbdingsymbol{010}                  {260E}}
13 \newcommand{\LWR@HTML@PhoneHandset}{\LWR@bbdingsymbol{011}           {2706}}
14 \newcommand{\LWR@HTML@Tape}{\LWR@bbdingsymbol{012}                   {2707}}
15 \newcommand{\LWR@HTML@Plane}{\LWR@bbdingsymbol{013}                  {2708}}
16 \newcommand{\LWR@HTML@Envelope}{\LWR@bbdingsymbol{014}               {2709}}
17 \newcommand{\LWR@HTML@HandCuffRight}{\LWR@bbdingsymbol{015}          {261B}}
18 \newcommand{\LWR@HTML@HandCuffLeft}{\LWR@bbdingsymbol{016}           {261A}}
19 \newcommand{\LWR@HTML@HandCuffRightUp}{\LWR@bbdingsymbol{017}         {261D}}
20 \newcommand{\LWR@HTML@HandCuffLeftUp}{\LWR@bbdingsymbol{020}          {261F}}
21 \newcommand{\LWR@HTML@HandRight}{\LWR@bbdingsymbol{021}              {261E}}
22 \newcommand{\LWR@HTML@HandLeft}{\LWR@bbdingsymbol{022}               {261C}}
23 \newcommand{\LWR@HTML@HandRightUp}{\LWR@bbdingsymbol{023}            {261D}}
24 \newcommand{\LWR@HTML@HandLeftUp}{\LWR@bbdingsymbol{024}             {261F}}
25 \newcommand{\LWR@HTML@Peace}{\LWR@bbdingsymbol{025}                  {270C}}
26 \newcommand{\LWR@HTML@HandPencilLeft}{\LWR@bbdingsymbol{026}          {270D}}
27 \newcommand{\LWR@HTML@PencilRight}{\LWR@bbdingsymbol{027}            {270F}}
28 \newcommand{\LWR@HTML@PencilLeft}{\LWR@bbdingsymbol{030}             {270F}}
29 \newcommand{\LWR@HTML@PencilRightUp}{\LWR@bbdingsymbol{031}           {2710}}
```

```
30 \newcommand{\LWR@HTML@PencilLeftUp}{\LWR@bbdingsymbol{032}          {2710}}
31 \newcommand{\LWR@HTML@PencilRightDown}{\LWR@bbdingsymbol{033}        {270E}}
32 \newcommand{\LWR@HTML@PencilLeftDown}{\LWR@bbdingsymbol{034}         {270E}}
33 \newcommand{\LWR@HTML@NibRight}{\LWR@bbdingsymbol{035}               {2711}}
34 \newcommand{\LWR@HTML@NibLeft}{\LWR@bbdingsymbol{036}                {2711}}
35 \newcommand{\LWR@HTML@NibSolidRight}{\LWR@bbdingsymbol{037}          {2712}}
36 \newcommand{\LWR@HTML@NibSolidLeft}{\LWR@bbdingsymbol{040}           {2712}}
37 \newcommand{\LWR@HTML@Checkmark}{\LWR@bbdingsymbol{041}              {2713}}
38 \newcommand{\LWR@HTML@CheckmarkBold}{\LWR@bbdingsymbol{042}          {2714}}
39 \newcommand{\LWR@HTML@XSolid}{\LWR@bbdingsymbol{043}                 {2715}}
40 \newcommand{\LWR@HTML@XSolidBold}{\LWR@bbdingsymbol{044}             {2716}}
41 \newcommand{\LWR@HTML@XSolidBrush}{\LWR@bbdingsymbol{045}            {2717}}
42 \newcommand{\LWR@HTML@PlusOutline}{\LWR@bbdingsymbol{046}            {2719}}
43 \newcommand{\LWR@HTML@Plus}{\LWR@bbdingsymbol{047}                   {271A}}
44 \newcommand{\LWR@HTML@PlusCenterOpen}{\LWR@bbdingsymbol{050}         {271C}}
45 \newcommand{\LWR@HTML@PlusThinCenterOpen}{\LWR@bbdingsymbol{051}     {271B}}
46 \newcommand{\LWR@HTML@Cross}{\LWR@bbdingsymbol{052}                  {271D}}
47 \newcommand{\LWR@HTML@CrossOpenShadow}{\LWR@bbdingsymbol{053}        {271E}}
48 \newcommand{\LWR@HTML@CrossOutline}{\LWR@bbdingsymbol{054}           {271F}}
49 \newcommand{\LWR@HTML@CrossBoldOutline}{\LWR@bbdingsymbol{055}       {271F}}
50 \newcommand{\LWR@HTML@CrossMaltese}{\LWR@bbdingsymbol{056}           {2720}}
51 \newcommand{\LWR@HTML@DavidStarSolid}{\LWR@bbdingsymbol{057}         {2721}}
52 \newcommand{\LWR@HTML@DavidStar}{\LWR@bbdingsymbol{060}              {2721}}
53 \newcommand{\LWR@HTML@FourAsterisk}{\LWR@bbdingsymbol{061}           {2722}}
54 \newcommand{\LWR@HTML@JackStar}{\LWR@bbdingsymbol{062}               {2723}}
55 \newcommand{\LWR@HTML@JackStarBold}{\LWR@bbdingsymbol{063}           {2724}}
56 \newcommand{\LWR@HTML@CrossClowerTips}{\LWR@bbdingsymbol{064}        {2725}}
57 \newcommand{\LWR@HTML@FourStar}{\LWR@bbdingsymbol{065}               {2726}}
58 \newcommand{\LWR@HTML@FourStarOpen}{\LWR@bbdingsymbol{066}           {2727}}
59 \newcommand{\LWR@HTML@FiveStarLines}{\LWR@bbdingsymbol{067}          {2729}}
60 \newcommand{\LWR@HTML@FiveStar}{\LWR@bbdingsymbol{070}               {2605}}
61 \newcommand{\LWR@HTML@FiveStarOpen}{\LWR@bbdingsymbol{071}           {2729}}
62 \newcommand{\LWR@HTML@FiveStarOpenCircled}{\LWR@bbdingsymbol{072}    {272A}}
63 \newcommand{\LWR@HTML@FiveStarCenterOpen}{\LWR@bbdingsymbol{073}     {272B}}
64 \newcommand{\LWR@HTML@FiveStarOpenDotted}{\LWR@bbdingsymbol{074}     {272C}}
65 \newcommand{\LWR@HTML@FiveStarOutline}{\LWR@bbdingsymbol{075}        {272D}}
66 \newcommand{\LWR@HTML@FiveStarOutlineHeavy}{\LWR@bbdingsymbol{076}   {272E}}
67 \newcommand{\LWR@HTML@FiveStarConvex}{\LWR@bbdingsymbol{077}         {272F}}
68 \newcommand{\LWR@HTML@FiveStarShadow}{\LWR@bbdingsymbol{100}         {2730}}
69 \newcommand{\LWR@HTML@AsteriskBold}{\LWR@bbdingsymbol{101}           {2731}}
70 \newcommand{\LWR@HTML@AsteriskCenterOpen}{\LWR@bbdingsymbol{102}     {2732}}
71 \newcommand{\LWR@HTML@AsteriskThin}{\LWR@bbdingsymbol{103}           {273B}}
72 \newcommand{\LWR@HTML@AsteriskThinCenterOpen}{\LWR@bbdingsymbol{104} {273C}}
73 \newcommand{\LWR@HTML@EightStarTaper}{\LWR@bbdingsymbol{105}         {2733}}
74 \newcommand{\LWR@HTML@EightStarConvex}{\LWR@bbdingsymbol{106}        {2735}}
75 \newcommand{\LWR@HTML@SixStar}{\LWR@bbdingsymbol{107}                {2736}}
76 \newcommand{\LWR@HTML@EightStar}{\LWR@bbdingsymbol{110}              {2737}}
77 \newcommand{\LWR@HTML@EightStarBold}{\LWR@bbdingsymbol{111}          {2738}}
78 \newcommand{\LWR@HTML@TwelweStar}{\LWR@bbdingsymbol{112}             {2739}}
79 \newcommand{\LWR@HTML@SixteenStarLight}{\LWR@bbdingsymbol{113}       {273A}}
80 \newcommand{\LWR@HTML@SixFlowerPetalRemoved}{\LWR@bbdingsymbol{114}  {273B}}
81 \newcommand{\LWR@HTML@SixFlowerOpenCenter}{\LWR@bbdingsymbol{115}    {273C}}
82 \newcommand{\LWR@HTML@Asterisk}{\LWR@bbdingsymbol{116}               {273D}}
83 \newcommand{\LWR@HTML@SixFlowerAlternate}{\LWR@bbdingsymbol{117}     {273E}}
84 \newcommand{\LWR@HTML@FiveFlowerPetal}{\LWR@bbdingsymbol{120}        {273F}}
85 \newcommand{\LWR@HTML@SixFlowerPetalDotted}{\LWR@bbdingsymbol{121}   {2740}}
86 \newcommand{\LWR@HTML@FiveFlowerOpen}{\LWR@bbdingsymbol{122}         {2740}}
87 \newcommand{\LWR@HTML@EightFlowerPetal}{\LWR@bbdingsymbol{123}       {2741}}
88 \newcommand{\LWR@HTML@SunshineOpenCircled}{\LWR@bbdingsymbol{124}    {2742}}
89 \newcommand{\LWR@HTML@SixFlowerAltPetal}{\LWR@bbdingsymbol{125}      {2743}}
```

```
90 \newcommand{\LWR@HTML@FourCloverOpen}{\LWR@bbdingsymbol{126}          {273F}}
91 \newcommand{\LWR@HTML@FourCloverSolid}{\LWR@bbdingsymbol{127}         {273F}}
92 \newcommand{\LWR@HTML@AsteriskRoundedEnds}{\LWR@bbdingsymbol{130}     {2749}}
93 \newcommand{\LWR@HTML@EightFlowerPetalRemoved}{\LWR@bbdingsymbol{131}  {274A}}
94 \newcommand{\LWR@HTML@EightAsterisk}{\LWR@bbdingsymbol{132}           {274B}}
95 \newcommand{\LWR@HTML@SixFlowerRemovedOpenPetal}{\LWR@bbdingsymbol{133}  {2740}}
96 \newcommand{\LWR@HTML@SparkleBold}{\LWR@bbdingsymbol{134}             {2748}}
97 \newcommand{\LWR@HTML@Sparkle}{\LWR@bbdingsymbol{135}                 {2747}}
98 \newcommand{\LWR@HTML@SnowflakeChevron}{\LWR@bbdingsymbol{136}        {2744}}
99 \newcommand{\LWR@HTML@SnowflakeChevronBold}{\LWR@bbdingsymbol{137}    {2746}}
100 \newcommand{\LWR@HTML@Snowflake}{\LWR@bbdingsymbol{140}              {2744}}
101 \newcommand{\LWR@HTML@CircleSolid}{\LWR@bbdingsymbol{141}            {25CF}}
102 \newcommand{\LWR@HTML@Ellipse}{\LWR@bbdingsymbol{142}               {274D}}
103 \newcommand{\LWR@HTML@EllipseSolid}{\LWR@bbdingsymbol{143}           {25CF}}
104 \newcommand{\LWR@HTML@CircleShadow}{\LWR@bbdingsymbol{144}           {274D}}
105 \newcommand{\LWR@HTML@EllipseShadow}{\LWR@bbdingsymbol{145}          {274D}}
106 \newcommand{\LWR@HTML@Square}{\LWR@bbdingsymbol{146}                {25A1}}
107 \newcommand{\LWR@HTML@SquareSolid}{\LWR@bbdingsymbol{147}            {25A0}}
108 \newcommand{\LWR@HTML@SquareShadowBottomRight}{\LWR@bbdingsymbol{150}   {2751}}
109 \newcommand{\LWR@HTML@SquareShadowTopRight}{\LWR@bbdingsymbol{151}     {2752}}
110 \newcommand{\LWR@HTML@SquareShadowTopLeft}{\LWR@bbdingsymbol{152}      {2752}}
111 \newcommand{\LWR@HTML@SquareCastShadowBottomRight}{\LWR@bbdingsymbol{153} {2751}}
112 \newcommand{\LWR@HTML@SquareCastShadowTopRight}{\LWR@bbdingsymbol{154}   {2752}}
113 \newcommand{\LWR@HTML@SquareCastShadowTopLeft}{\LWR@bbdingsymbol{155}    {2752}}
114 \newcommand{\LWR@HTML@TriangleUp}{\LWR@bbdingsymbol{156}             {25B2}}
115 \newcommand{\LWR@HTML@TriangleDown}{\LWR@bbdingsymbol{157}           {25BC}}
116 \newcommand{\LWR@HTML@DiamondSolid}{\LWR@bbdingsymbol{160}           {25C6}}
117 \newcommand{\LWR@HTML@OrnamentDiamondSolid}{\LWR@bbdingsymbol{161}    {2756}}
118 \newcommand{\LWR@HTML@HalfCircleRight}{\LWR@bbdingsymbol{162}        {25D7}}
119 \newcommand{\LWR@HTML@HalfCircleLeft}{\LWR@bbdingsymbol{163}         {25D6}}
120 \newcommand{\LWR@HTML@RectangleThin}{\LWR@bbdingsymbol{164}          {2758}}
121 \newcommand{\LWR@HTML@Rectangle}{\LWR@bbdingsymbol{165}              {2759}}
122 \newcommand{\LWR@HTML@RectangleBold}{\LWR@bbdingsymbol{166}          {275A}}
123 \newcommand{\LWR@HTML@ArrowBoldRightStrobe}{\LWR@bbdingsymbol{167}     {27A0}}
124 \newcommand{\LWR@HTML@ArrowBoldUpRight}{\LWR@bbdingsymbol{170}        {27A6}}
125 \newcommand{\LWR@HTML@ArrowBoldDownRight}{\LWR@bbdingsymbol{171}      {27A5}}
126 \newcommand{\LWR@HTML@ArrowBoldRightShort}{\LWR@bbdingsymbol{172}     {27A7}}
127 \newcommand{\LWR@HTML@ArrowBoldRightCircled}{\LWR@bbdingsymbol{173}    {27B2}}
128
129
130 \LWR@formatted{ScissorRightBrokenBottom}
131 \LWR@formatted{ScissorRight}
132 \LWR@formatted{ScissorRightBrokenTop}
133 \LWR@formatted{ScissorLeftBrokenBottom}
134 \LWR@formatted{ScissorLeft}
135 \LWR@formatted{ScissorLeftBrokenTop}
136 \LWR@formatted{ScissorHollowRight}
137 \LWR@formatted{ScissorHollowLeft}
138 \LWR@formatted{Phone}
139 \LWR@formatted{PhoneHandset}
140 \LWR@formatted{Tape}
141 \LWR@formatted{Plane}
142 \LWR@formatted{Envelope}
143 \LWR@formatted{HandCuffRight}
144 \LWR@formatted{HandCuffLeft}
145 \LWR@formatted{HandCuffRightUp}
146 \LWR@formatted{HandCuffLeftUp}
147 \LWR@formatted{HandRight}
148 \LWR@formatted{HandLeft}
149 \LWR@formatted{HandRightUp}
```

```
150 \LWR@formatted{HandLeftUp}
151 \LWR@formatted{Peace}
152 \LWR@formatted{HandPencilLeft}
153 \LWR@formatted{PencilRight}
154 \LWR@formatted{PencilLeft}
155 \LWR@formatted{PencilRightUp}
156 \LWR@formatted{PencilLeftUp}
157 \LWR@formatted{PencilRightDown}
158 \LWR@formatted{PencilLeftDown}
159 \LWR@formatted{NibRight}
160 \LWR@formatted{NibLeft}
161 \LWR@formatted{NibSolidRight}
162 \LWR@formatted{NibSolidLeft}
163 \LWR@formatted{Checkmark}
164 \LWR@formatted{CheckmarkBold}
165 \LWR@formatted{XSolid}
166 \LWR@formatted{XSolidBold}
167 \LWR@formatted{XSolidBrush}
168 \LWR@formatted{PlusOutline}
169 \LWR@formatted{Plus}
170 \LWR@formatted{PlusCenterOpen}
171 \LWR@formatted{PlusThinCenterOpen}
172 \LWR@formatted{Cross}
173 \LWR@formatted{CrossOpenShadow}
174 \LWR@formatted{CrossOutline}
175 \LWR@formatted{CrossBoldOutline}
176 \LWR@formatted{CrossMaltese}
177 \LWR@formatted{DavidStarSolid}
178 \LWR@formatted{DavidStar}
179 \LWR@formatted{FourAsterisk}
180 \LWR@formatted{JackStar}
181 \LWR@formatted{JackStarBold}
182 \LWR@formatted{CrossClowerTips}
183 \LWR@formatted{FourStar}
184 \LWR@formatted{FourStarOpen}
185 \LWR@formatted{FiveStarLines}
186 \LWR@formatted{FiveStar}
187 \LWR@formatted{FiveStarOpen}
188 \LWR@formatted{FiveStarOpenCircled}
189 \LWR@formatted{FiveStarCenterOpen}
190 \LWR@formatted{FiveStarOpenDotted}
191 \LWR@formatted{FiveStarOutline}
192 \LWR@formatted{FiveStarOutlineHeavy}
193 \LWR@formatted{FiveStarConvex}
194 \LWR@formatted{FiveStarShadow}
195 \LWR@formatted{AsteriskBold}
196 \LWR@formatted{AsteriskCenterOpen}
197 \LWR@formatted{AsteriskThin}
198 \LWR@formatted{AsteriskThinCenterOpen}
199 \LWR@formatted{EightStarTaper}
200 \LWR@formatted{EightStarConvex}
201 \LWR@formatted{SixStar}
202 \LWR@formatted{EightStar}
203 \LWR@formatted{EightStarBold}
204 \LWR@formatted{TwelweStar}
205 \LWR@formatted{SixteenStarLight}
206 \LWR@formatted{SixFlowerPetalRemoved}
207 \LWR@formatted{SixFlowerOpenCenter}
208 \LWR@formatted{Asterisk}
209 \LWR@formatted{SixFlowerAlternate}
```

```
210 \LWR@formatted{FiveFlowerPetal}
211 \LWR@formatted{SixFlowerPetalDotted}
212 \LWR@formatted{FiveFlowerOpen}
213 \LWR@formatted{EightFlowerPetal}
214 \LWR@formatted{SunshineOpenCircled}
215 \LWR@formatted{SixFlowerAltPetal}
216 \LWR@formatted{FourCloverOpen}
217 \LWR@formatted{FourCloverSolid}
218 \LWR@formatted{AsteriskRoundedEnds}
219 \LWR@formatted{EightFlowerPetalRemoved}
220 \LWR@formatted{EightAsterisk}
221 \LWR@formatted{SixFlowerRemovedOpenPetal}
222 \LWR@formatted{SparkleBold}
223 \LWR@formatted{Sparkle}
224 \LWR@formatted{SnowflakeChevron}
225 \LWR@formatted{SnowflakeChevronBold}
226 \LWR@formatted{Snowflake}
227 \LWR@formatted{CircleSolid}
228 \LWR@formatted{Ellipse}
229 \LWR@formatted{EllipseSolid}
230 \LWR@formatted{CircleShadow}
231 \LWR@formatted{EllipseShadow}
232 \LWR@formatted{Square}
233 \LWR@formatted{SquareSolid}
234 \LWR@formatted{SquareShadowBottomRight}
235 \LWR@formatted{SquareShadowTopRight}
236 \LWR@formatted{SquareShadowTopLeft}
237 \LWR@formatted{SquareCastShadowBottomRight}
238 \LWR@formatted{SquareCastShadowTopRight}
239 \LWR@formatted{SquareCastShadowTopLeft}
240 \LWR@formatted{TriangleUp}
241 \LWR@formatted{TriangleDown}
242 \LWR@formatted{DiamondSolid}
243 \LWR@formatted{OrnamentDiamondSolid}
244 \LWR@formatted{HalfCircleRight}
245 \LWR@formatted{HalfCircleLeft}
246 \LWR@formatted{RectangleThin}
247 \LWR@formatted{Rectangle}
248 \LWR@formatted{RectangleBold}
249 \LWR@formatted{ArrowBoldRightStrobe}
250 \LWR@formatted{ArrowBoldUpRight}
251 \LWR@formatted{ArrowBoldDownRight}
252 \LWR@formatted{ArrowBoldRightShort}
253 \LWR@formatted{ArrowBoldRightCircled}
```

File 45 **`lwarp-beamerarticle.sty`**

§ 157   Package   # beamerarticle

(*Emulates or patches code by* Till Tantau, Vedran Miletić, Louis Stuart, Joseph Wright.)

beamerarticle (*Pkg*)   beamerarticle is patched for use by lwarp.

for HTML output: `1 \LWR@ProvidesPackagePass{beamerarticle}[2021/05/26]`

```
2 \renewcommand<>{\textcolor}{\only#1{\beameroriginal{\textcolor}}}
3
```

```
 4 \AtBeginDocument{

 5

 6 \renewcommand<>{\LWR@listitem}{%
 7     \only#1{%
 8         \beameroriginal{\LWR@listitem}%
 9     }%
10 }

11

12 \renewcommand<>{\LWR@itemizeitem}{%
13     \only#1{%
14         \beameroriginal{\LWR@itemizeitem}%
15     }%
16 }

17

18 \renewcommand<>{\LWR@descitem}{%
19     \only#1{%
20         \beameroriginal{\LWR@descitem}%
21     }%
22 }

23

24 \renewcommand<>{\abstract}{%
25     \only#1{%
26         \beameroriginal{\abstract}%
27     }%
28 }

29

30 \renewcommand<>{\LWR@includegraphicsb}{%
31     \only#1{%
32         \beameroriginal{\LWR@includegraphicsb}%
33     }%
34 }

35

36 \xpretocmd\frame
37     {
38             \LWR@forcenewpage
39             \BlockClass{beamerframe}%
40     }
41     {}
42     {\LWR@patcherror{beamerarticle}{frame}}

43

44 \xapptocmd\beamer@endframe
45     {\endBlockClass}
46     {}
47     {\LWR@patcherror{beamerarticle}{beamer@endframe}}
```

An example in the beamer docs for `\includegraphics` shows the use of `\llap` in a frame.

```
48 \xpretocmd\beamer@article@startframe
49     {\LWR@nulllistfills}
50     {}
51     {\LWR@patcherror{beamerarticle}{beamer@article@startframe}}

52

53 }% AtBeginDocument

54

55 \let\beamer@@tmpop@frametitle@default\relax
56 \defbeamertemplate<article>*{frametitle}{default}{%
57     \paragraph*{\insertframetitle}\ \par%
58     \ifdefempty{\insertframesubtitle}{}{%
59         \noindent\emph{\insertframesubtitle}\par%
```

```
60        }%
61 }
62
63
64 \NewDocumentCommand{\LWR@beamer@itemize}{o}{%
65        \LWR@itemizestart\LWR@origitemize%
66 }%
67 \NewDocumentCommand{\LWR@beamer@description}{o o}{%
68        \LWR@descriptionstart\LWR@origdescription%
69 }%
70
71 \xapptocmd{\LWR@patchlists}
72        {%
73            \LetLtxMacro\itemize\LWR@beamer@itemize%
74            \LetLtxMacro\description\LWR@beamer@description%
75        }
76        {}
77        {\LWR@patcherror{beamerarticle}{LWR@patchlists}}
78
79
80 \LetLtxMacro\maketitle\LWR@maketitle
81
82 \renewcommand{\subtitle}[2][]{
83        \gdef\@subtitle{#2}
84        \def\insertsubtitle{#2}
85 }
```

Add subtitle if not already present:

```
86 \AtBeginDocument{
87 \IfPackageLoadedTF{lwarp-scrextend}
88        {}% komascript already has subtitle
89        {% not komascript
90            \xpatchcmd{\@maketitle}
91                {%
92                    \LWR@htmltag{\LWR@tagtitleend}%
93                    \LWR@startpars%
94                }%
95                {%
96                    \LWR@htmltag{\LWR@tagtitleend}%
97                    \ifdefvoid{\@subtitle}{}{%
98                        \begin{BlockClass}{subtitle}%
99                        \@subtitle%
100                        \end{BlockClass}%
101                    }%
102                    \LWR@startpars%
103                }%
104                {}
105                {\LWR@patcherror{beamerarticle}{@maketitle}}
106        }% not komascript
107 }
108
109 \RequirePackage{fancyvrb}
110 \DefineVerbatimEnvironment{semiverbatim}{Verbatim}{commandchars=\\\{\}}
```

File 46 **`lwarp-biblatex.sty`**

§ 158   Package **biblatex**

(*Emulates or patches code by* PHILIPP LEHMAN.)

biblatex (*Pkg*)   When biblatex is used, modifications from newfloat may have to be undone.

for HTML output:

1. lwarp uses newfloat.

2. For classes with chapters which newfloat does not know about, such as CTEX-related classes, newfloat may modify \addtocontents.

3. biblatex, though, wants to patch \addtocontents, which causes an error if \addtocontents has been changed.

4. Therefore, \addtocontents is restored to its original here, since biblatex is about to be loaded.

5. This means that the newfloat's chapterlistsgaps option may no longer work.

```
1 \ifdef{\newfloat@addtocontents@ORI}{
2     \let\addtocontents\newfloat@addtocontents@ORI
3 }{}
```

hyperref emulation is loaded \AtBeginDocument to avoid an options clash.

```
4 \AtBeginDocument{\RequirePackage{hyperref}}
5
6 \LWR@ProvidesPackagePass{biblatex}[2018/03/04]
```

The following create hyperlinks to the references. The original code to use hyperref is recreated here, because hyperref is emulated.

```
7 \AfterPreamble{
```

Not using \VerifyCommand because this may be defined several ways.

```
8 \let\blx@anchors\@empty
9 \protected\def\blx@anchor{%
10     \xifinlist{\the\c@refsection @\abx@field@entrykey}{\blx@anchors}
11         {}
12         {\listxadd\blx@anchors{\the\c@refsection @\abx@field@entrykey}%
13         \hypertarget{cite.\the\c@refsection @\abx@field@entrykey}{}}}
14
15 \protected\def\blx@imc@bibhyperref{%
16     \@ifnextchar[%
17         {\blx@bibhyperref}
18         {\blx@bibhyperref[\abx@field@entrykey]}}%
19
20 \long\def\blx@bibhyperref[#1]#2{%
21 %        \blx@sfsave
22         \hyperlink{cite.\the\c@refsection @#1}{%
```

```
23 %            \blx@sfrest
24         #2%
25 %        \blx@sfsave
26       }%
27 % \blx@sfrest%
28 }%% \def\blx@nohyperref[#1]#2{#2}%
29
30 \protected\long\def\blx@imc@bibhyperlink#1#2{%
31 %        \blx@sfsave
32       \hyperlink{cite.\the\c@refsection:#1}{%
33 %         \blx@sfrest
34        #2%
35 %        \blx@sfsave
36       }%
37 %          \blx@sfrest%
38 }%
39
40 \protected\long\def\blx@imc@bibhypertarget#1#2{%
41 %        \blx@sfsave%
42       \hypertarget{cite.\the\c@refsection:#1}{%
43 %         \blx@sfrest
44        #2%
45 %        \blx@sfsave%
46       }%
47 %          \blx@sfrest%
48 }
49
50 \let\blx@imc@ifhyperref\@firstoftwo
```

Ensure that an autopage reference is current where each \cite is used, although
this is nullified inside footnotes since they now use a LATEX box.

```
51 \xpretocmd{\blx@citecmdinit}
52     {\LWR@newautopagelabel{page}}%
53     {}
54     {\LWR@patcherror{biblatex}{blx@citecmdinit}}
```

Ensure that an autopage reference is current for each backref. If the citation is in
a footnote, the backref will point to whatever preceeded the footnotes.

```
55 \VerifyCommand[lwarp][biblatex]{\blx@addbackref@i}{C820E8B12CF2904906644302E07EBE88}
56
57 \xpatchcmd{\blx@addbackref@i}
58     {\thepage}
59     {\theLWR@previousautopagelabel}% ref to the most recent object
60     {}
61     {\LWR@patcherror{biblatex}{blx@addbackref@i A}}
62
63 \xpatchcmd{\blx@addbackref@i}
64     {\c@page}
65     {\c@LWR@previousautopagelabel}% ref to the most recent object
66     {}
67     {\LWR@patcherror{biblatex}{blx@addbackref@i B}}
```

The following patches are for back page references.

```
68 \DeclareListFormat{pageref}{%
69   \ifnumless{\abx@pagerefstyle}{0}
70     {\usebibmacro{list:plain}}%
```

```
71      \ifhyperref
72      {%
73 %            \hyperlink{page.#1}{#1}%
74            \LWR@refwithsection{\BaseJobname-autopage-#1}%  lwarp
75      }
76      {#1}}
77    {\ifnumequal{\value{listcount}}{1}
78      {\usebibmacro{pageref:init}}
79      {}%
80    \usebibmacro{pageref:comp}{#1}%
81    \ifnumequal{\value{listcount}}{\value{liststop}}
82      {\usebibmacro{pageref:dump}}
83      {}}}

84

85 \expandafter\VerifyCommand\expandafter{\csname abx@macro@pageref:comp\endcsname}
86    {019E018D2EBB4F3D02578439F03128D8}

87

88 \renewbibmacro*{pageref:comp}[1]{%
89   \numdef\abx@range@prev{\abx@range@prev+1}%
90   \ifinteger{#1}
91     {\def\abx@range@num{#1}%
92      \def\abx@range@this{1}%
93      \ifnumequal{\abx@range@this}{\abx@range@last}
94        {}
95        {\def\abx@range@prev{-1}}}
96     {\ifrmnum{#1}
97        {\numdef\abx@range@num{\rmntonum{#1}}%
98         \def\abx@range@this{2}%
99         \ifnumequal{\abx@range@this}{\abx@range@last}
100          {}
101          {\def\abx@range@prev{-1}}}
102        {\undef\abx@range@num
103         \def\abx@range@this{0}%
104         \def\abx@range@prev{-1}}}%
105   \ifdef\abx@range@num
106     {\ifnumequal{\abx@range@num}{\abx@range@prev}
107        {\def\abx@range@hold{#1}%
108         \numdef\abx@range@diff{\abx@range@diff+1}}
109        {\usebibmacro{pageref:dump}%
110         \ifnumgreater{\abx@range@last}{-1}
111           {\printdelim{multilistdelim}}
112           {}%
113         \ifhyperref
114 %          {\hyperlink{page.#1}{#1}}
115           {\LWR@refwithsection{\BaseJobname-autopage-#1}}%  lwarp
116           {#1}}%
117      \edef\abx@range@prev{\abx@range@num}}
118     {\usebibmacro{pageref:dump}%
119      \ifnumgreater{\abx@range@last}{-1}
120        {\printdelim{multilistdelim}}
121        {}%
122      \ifhyperref
123 %        {\hyperlink{page.#1}{#1}}
124        {\LWR@refwithsection{\BaseJobname-autopage-#1}}%  lwarp
125        {#1}%
126      \def\abx@range@prev{-1}}%
127   \edef\abx@range@last{\abx@range@this}}

128

129 \expandafter\VerifyCommand\expandafter{\csname abx@macro@pageref:dump\endcsname}
130    {9BD1165E771053A5DA8957BE4E2E7B9E}
```

```
131
132 \renewbibmacro*{pageref:dump}{%
133   \ifnumgreater{\abx@range@diff}{0}
134     {\ifcase\abx@pagerefstyle\relax % two
135       \bibrangedash
136       \ifhyperref
137 %        {\hyperlink{page.\abx@range@hold}{\abx@range@hold}}
138         {\LWR@refwithsection{\BaseJobname-autopage-\abx@range@hold}}% lwarp
139         {\abx@range@hold}%
140     \or % three
141       \ifnumless{\abx@range@diff}{2}
142         {\printdelim{multilistdelim}}
143         {\bibrangedash}%
144       \ifhyperref
145 %        {\hyperlink{page.\abx@range@hold}{\abx@range@hold}}
146         {\LWR@refwithsection{\BaseJobname-autopage-\abx@range@hold}}% lwarp
147         {\abx@range@hold}%
148     \or % two+
149       \ifnumless{\abx@range@diff}{2}
150         {\sqspace
151          \ifhyperref
152 %           {\hyperlink{page.\abx@range@hold}{\bibstring{sequens}}}
153          {\LWR@refwithsection{\BaseJobname-autopage-\abx@range@hold}}% lwarp
154             {\bibstring{sequens}}}
155         {\bibrangedash
156          \ifhyperref
157 %           {\hyperlink{page.\abx@range@hold}{\abx@range@hold}}
158          {\LWR@refwithsection{\BaseJobname-autopage-\abx@range@hold}}% lwarp
159             {\abx@range@hold}}%
160     \or % three+
161       \ifnumless{\abx@range@diff}{2}
162         {\sqspace
163          \ifhyperref
164 %           {\hyperlink{page.\abx@range@hold}{\bibstring{sequens}}}
165          {\LWR@refwithsection{\BaseJobname-autopage-\abx@range@hold}}% lwarp
166             {\bibstring{sequens}}}
167         {\ifnumless{\abx@range@diff}{3}
168           {\sqspace
169            \ifhyperref
170 %             {\hyperlink{page.\abx@range@hold}{\bibstring{sequentes}}}
171            {\LWR@refwithsection{\BaseJobname-autopage-\abx@range@hold}}% lwarp
172               {\bibstring{sequentes}}}
173           {\bibrangedash
174            \ifhyperref
175 %             {\hyperlink{page.\abx@range@hold}{\abx@range@hold}}
176            {\LWR@refwithsection{\BaseJobname-autopage-\abx@range@hold}}% lwarp
177               {\abx@range@hold}}%
178     \else % all+
179       \ifnumless{\abx@range@diff}{2}
180         {\sqspace
181          \ifhyperref
182 %           {\hyperlink{page.\abx@range@hold}{\bibstring{sequens}}}
183          {\LWR@refwithsection{\BaseJobname-autopage-\abx@range@hold}}% lwarp
184             {\bibstring{sequens}}}
185         {\sqspace
186          \ifhyperref
187 %           {\hyperlink{page.\abx@range@hold}{\bibstring{sequentes}}}
188          {\LWR@refwithsection{\BaseJobname-autopage-\abx@range@hold}}% lwarp
189             {\bibstring{sequentes}}}%
190     \fi
```

```
191        \def\abx@range@diff{0}}
192      {}}
193
194 }% \AfterPreamble
```

File 47   **`lwarp-bibunits.sty`**

§ 159   Package   # bibunits

(*Emulates or patches code by* THORSTEN HANSEN.)

bibunits (*Pkg*)   bibunits is patched for use by lwarp.

for HTML output: `1 \LWR@ProvidesPackagePass{bibunits}[2004/05/12]`

`2 \def\bu@bibdata{\BaseJobname}`

File 48   **`lwarp-bigdelim.sty`**

§ 160   Package   # bigdelim

(*Emulates or patches code by* PIET VAN OOSTRUM, ØYSTEIN BACHE, JERRY LEICHTER.)

bigdelim (*Pkg*)   bigdelim is used as-is for print or lateximage, and patched for HTML.

The delimiters are displayed in HTML by printing the delimiter, the text, and a thick border across the side of the \multirow which indicates the actual height of the delimiter. The delimiter character is given a <span> class of ldelim or rdelim, and the default CSS sets this to font-size:200%

⚠ use \mrowcell   \ldelim and \rdelim use \multirow, so \mrowcell must be used in the proper number of empty cells in the same column below \ldelim or \rdelim, but not in cells which are above or below the delimiter:

```
\begin{tabular}{lll}
<empty> & a & b \\
\ldelim{\{}{3}{.25in}[left ] & c & d \\
\mrowcell & e & f \\
\mrowcell & g & h \\
<empty> & i & j \\
\end{tabular}
```

```
       <->   a   b
             ⎧ c   d
             ⎪
       left  ⎨ e   f
             ⎪
             ⎩ g   h
       <->   i   j
```

For MATHJAX, limited emulation is provided which merely prints the delimter and optional text in the first row.

for HTML output: First, remove the temporary definitions of \ldelim and \rdelim, which were previously defined for tabular scanning in case bigdelim was not loaded:

```
1 \let\ldelim\relax
2 \let\rdelim\relax
```

Next, load the package's new definitions:

```
3 \LWR@ProvidesPackagePass{bigdelim}[2021/03/15]
```

\ldelim   {⟨*1:delimiter*⟩} {⟨*2:#rows*⟩} [⟨*3: vmove*⟩] {⟨*4:width*⟩} [⟨*5:text*⟩]
\rdelim

```
4 \NewDocumentCommand{\LWR@HTML@ldelim}{m m o m O{}}{%
5 \renewcommand{\LWR@multirowborder}{right}%
6 \multirow{#2}{#4}{#5 \InlineClass{ldelim}{#1}}%
7 }
8
9 \LWR@formatted{ldelim}
10
11 \NewDocumentCommand{\LWR@HTML@rdelim}{m m o m O{}}{%
12 \renewcommand{\LWR@multirowborder}{left}%
13 \multirow{#2}{#4}{\InlineClass{rdelim}{#1} #5}%
14 }
15
16 \LWR@formatted{rdelim}
```

Limited emulation for MATHJAX. The delimiter is printed on the first row, along with any optional text.

```
17 \begin{warpMathJax}
18 % \ldelim ( {n}{width}[text]
19 \CustomizeMathJax{\newcommand{\LWRldelimtwo}[1][]{\text{#1}~\LWRbigdelim}}
20 \CustomizeMathJax{\newcommand{\LWRldelimone}[2][]{\LWRldelimtwo}}
21 \CustomizeMathJax{\def\ldelim#1#2{\def\LWRbigdelim{#1}\LWRldelimone}}
22 % \rdelim ) {n}{width}[text]
23 \CustomizeMathJax{\newcommand{\LWRrdelimtwo}[1][]{\LWRbigdelim~\text{#1}}}
24 \CustomizeMathJax{\newcommand{\LWRrdelimone}[2][]{\LWRrdelimtwo}}
25 \CustomizeMathJax{\def\rdelim#1#2{\def\LWRbigdelim{#1}\LWRrdelimone}}
26 \end{warpMathJax}
```

File 49   `lwarp-bigfoot.sty`

§ 161   Package   # bigfoot

bigfoot (*Pkg*)   bigfoot is emulated.

for HTML output: `1 \LWR@ProvidesPackageDrop{bigfoot}[2015/08/30]`

```
2 \RequirePackage{manyfoot}
3 \RequirePackage{perpage}
4
5 \def\RestyleFootnote#1#2{}
6 \def\FootnoteSpecific#1{}
7 \def\DefineFootnoteStack#1{}
8 \def\PushFootnoteMark#1{}
9 \def\PopFootnoteMark#1{}
```

```
10 \def\hfootfraction{0.9}
11 \def\vtypefraction{0.7}
12 \def\FootnoteMinimum{1sp}
13 \def\FootnoteMainMinimum{0pt}
14 \newcount\bigfoottolerance
15 \bigfoottolerance=100
16 \providecommand\footnotecarryratio{2}
```

File 50   **`lwarp-bigstrut.sty`**

§ 162   Package   # bigstrut

(*Emulates or patches code by* Piet van Oostrum, Øystein Bache, Jerry Leichter.)

bigstrut (*Pkg*)   bigstrut is used as-is for print or lateximage, and patched for HTML.

for HTML output: `1 \LWR@ProvidesPackagePass{bigstrut}[2018/08/03]`

```
2 \LetLtxMacro\LWR@origbigstrut\bigstrut
3
4 \renewcommand\bigstrut[1][x]{}
5
6 \appto\LWR@restoreorigformatting{%
7 \LetLtxMacro\bigstrut\LWR@origbigstrut%
8 }
9
```

```
10 \begin{warpMathJax}
11 \CustomizeMathJax{\newcommand{\bigstrut}[1][]{}}
12 \end{warpMathJax}
```

File 51   **`lwarp-bitpattern.sty`**

§ 163   Package   # bitpattern

(*Emulates or patches code by* Jean-Marc Bourguet.)

bitpattern (*Pkg*)   bitpattern is patched for use by lwarp.

for HTML output: `1 \LWR@ProvidesPackagePass{bitpattern}[2015/12/11]`

```
2 \VerifyCommand[lwarp][bitpattern]{\bitpattern}{379A39416C9C5E48DBCEEF730D51C5BF}
3
4 \xpatchcmd{\bitpattern}
5    {\begingroup}
6    {\begin{lateximage}[-bitpattern-~\PackageDiagramAltText]?}%
7    {}
8    {\LWR@patcherror{bitpattern}{bitpattern}}
9
10 \VerifyCommand[lwarp][bitpattern]{\bp@Done}{4F2F6DDB41FE31051ACA3CA9F58E3395}
11
12 \xpatchcmd{\bp@Done}
13    {\endgroup}
```

```
14      {\end{lateximage}}
15      {}
16      {\LWR@patcherror{bitpattern}{bp@Done}}
```

File 52   **lwarp-blowup.sty**

§ 164   Package   # blowup

blowup (*Pkg*)   blowup is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{blowup}[2018/01/02]

2 \newcommand*\blowUp[1]{}

File 53   **lwarp-bm.sty**

§ 165   Package   # bm

(*Emulates or patches code by* DAVID CARLISLE, FRANK MITTELBACH.)

bm (*Pkg*)   bm is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{bm}[2019/07/24]

\DeclareBoldMathCommand must only be used in the preamble, since it adds to the MATHJAX setup code.

```
2 \begin{warpMathJax}
3 \LetLtxMacro\LWR@orig@DeclareBoldMathCommand\DeclareBoldMathCommand
4
5 \renewcommand\DeclareBoldMathCommand[3][bold]{%
6     \LWR@orig@DeclareBoldMathCommand[#1]{#2}{#3}%
7     \CustomizeMathJax{\newcommand{#2}{\boldsymbol{#3}}}%
8 }
9
10 \@onlypreamble\DeclareBoldMathCommand
11
12 \CustomizeMathJax{\newcommand{\bm}[1]{\boldsymbol{#1}}}
13 \end{warpMathJax}
```

File 54   **lwarp-booklet.sty**

§ 166   Package   # booklet

(*Emulates or patches code by* PETER WILSON.)

booklet (*Pkg*)   booklet is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{booklet}[2009/09/02]

```
2 \newdimen\pageseplength
3 \newdimen\pagesepwidth
4 \newdimen\pagesepoffset
5 \newif\ifsidebyside      \sidebysidetrue
6 \newif\ifuselandscape     \uselandscapefalse
7 \newif\ifprintoption      \printoptionfalse
8 \newcommand*{\pagespersignature}[1]{}
9 \def\magstepminus#1{}
10 \newcommand*{\target}[3]{}
11 \newcommand*{\source}[3]{}
12 \newcommand*{\setpdftargetpages}{}
13 \newcommand*{\setdvipstargetpages}{}
14 \newcommand*{\targettopbottom}{}
15 \newcommand*{\twoupemptypage}{}
16 \newcommand*{\twoupclearpage}{}
17 \newcommand*{\checkforlandscape}{}
```

File 55   `lwarp-bookmark.sty`

§ 167   Package   # bookmark

(*Emulates or patches code by* HEIKO OBERDIEK.)

bookmark (*Pkg*)   **bookmark** is ignored.

for HTML output: Discard all options for lwarp-bookmark:

```
1 \LWR@ProvidesPackageDrop{bookmark}[2016/05/17]
```

```
2 \newcommand*{\bookmarksetup}[1]{}
3 \newcommand*{\bookmarksetupnext}[1]{}
4 \newcommand*{\bookmark}[2][]{}
5 \newcommand*{\bookmarkdefinestyle}[2]{}
6 \newcommand*{\bookmarkget}[1]{}
7 \newcommand{\BookmarkAtEnd}[1]{}
```

File 56   `lwarp-booktabs.sty`

§ 168   Package   # booktabs

(*Emulates or patches code by* SIMON FEAR.)

booktabs (*Pkg*)   **booktabs** is emulated during HTML output, and used as-is during print output and inside an HTML lateximage.

⚠   \cmidrule   For MATHJAX, emulation is provided in math mode, but \cmidrule trim must not be used.

for HTML output: If **booktabs** has already been loaded before lwarp, such as by memoir, use it as-is. If not, the lwarp core will have placed some dummy macros which should be removed before loading the actual **booktabs** definitions.

```
1 \IfPackageLoadedTF{booktabs}{}{
2     \LetLtxMacro\toprule\relax
3     \LetLtxMacro\midrule\relax
```

```
 4      \LetLtxMacro\cmidrule\cline
 5      \LetLtxMacro\bottomrule\relax
 6      \LetLtxMacro\addlinespace\relax
 7      \LetLtxMacro\morecmidrules\relax
 8      \LetLtxMacro\specialrule\relax
 9 }
```

Next, load the booktabs package:

```
10 \LWR@ProvidesPackagePass{booktabs}[2019/10/08]
```

Adjust to work even if xltabular is loaded:

```
11 % \def\LWR@HTML@@BLTrule{\@BTnormal}
12 %
13 % \LWR@formatted{@BLTrule}
14 \LetLtxMacro\@BLTrule\@BTnormal


15 \DeclareDocumentCommand{\LWR@HTML@toprule}{o d()}%
16     {%
17         \IfValueTF{#1}%
18             {\LWR@docmidrule[#1](){1-\arabic{LWR@tabletotalLaTeXcols}}}%
19             {%
20                 \ifbool{FormatWP}%
21                 {\LWR@docmidrule[#1](){1-\arabic{LWR@tabletotalLaTeXcols}}}%
22                     {\booltrue{LWR@doingtbrule}}%
23             }%
24     \LWR@getmynexttoken}
25
26 \LWR@expandableformatted{toprule}
27
28 \DeclareDocumentCommand{\LWR@HTML@midrule}{o d()}%
29     {%
30         \IfValueTF{#1}%
31             {\LWR@docmidrule[#1](){1-\arabic{LWR@tabletotalLaTeXcols}}}%
32             {%
33                 \ifbool{FormatWP}%
34                 {\LWR@docmidrule[#1](){1-\arabic{LWR@tabletotalLaTeXcols}}}%
35                     {\defaddtocounter{LWR@hlines}{1}}%
36             }%
37     \LWR@getmynexttoken}
38
39 \LWR@expandableformatted{midrule}
40
41 \DeclareDocumentCommand{\LWR@HTML@cmidrule}{O{\LWR@cmidrulewidth} d() m}{%
42     \LWR@docmidrule[#1](#2){#3}%
43     \LWR@getmynexttoken%
44 }%
45
46 \LWR@expandableformatted{cmidrule}
47
48 \DeclareDocumentCommand{\LWR@HTML@bottomrule}{o d()}{%
49     \IfValueTF{#1}%
50         {\LWR@docmidrule[#1](){1-\arabic{LWR@tabletotalLaTeXcols}}}%
51         {%
52             \ifbool{FormatWP}%
53                 {\LWR@docmidrule[#1](){1-\arabic{LWR@tabletotalLaTeXcols}}}%
54                 {\booltrue{LWR@doingtbrule}}%
55         }%
```

```
56    \LWR@getmynexttoken%
57 }%
58
59 \LWR@expandableformatted{bottomrule}
60
61 \DeclareDocumentCommand{\LWR@HTML@addlinespace}{o}{}%
62
63 \LWR@expandableformatted{addlinespace}
64
65 \DeclareDocumentCommand{\LWR@HTML@morecmidrules}{}{}%
66
67 \LWR@expandableformatted{morecmidrules}
68
69 \DeclareDocumentCommand{\LWR@HTML@specialrule}{m m m d()}%
70    {\LWR@docmidrule[#1]()(1-\arabic{LWR@tabletotalLaTeXcols}}\LWR@getmynexttoken}%
71
72 \LWR@expandableformatted{specialrule}
```

For MATHJAX:

```
73 \begin{warpMathJax}
74 \CustomizeMathJax{\newcommand{\toprule}[1][]{\hline}}
75 \CustomizeMathJax{\let\midrule\toprule}
76 \CustomizeMathJax{\let\bottomrule\toprule}
77 \CustomizeMathJax{\def\LWRbooktabscmidruleparen(#1)#2{}}
78 \CustomizeMathJax{\newcommand{\LWRbooktabscmidrulenoparen}[1]{}}
79 \CustomizeMathJax{\newcommand{\cmidrule}[1][]{%
80    \ifnextchar(\LWRbooktabscmidruleparen\LWRbooktabscmidrulenoparen%
81 }}
82 \CustomizeMathJax{\newcommand{\morecmidrules}{}}
83 \CustomizeMathJax{\newcommand{\specialrule}[3]{\hline}}
84 \CustomizeMathJax{\newcommand{\addlinespace}[1][]{}}
85 \end{warpMathJax}
```

File 57   `lwarp-bophook.sty`

§ 169   Package   # bophook

bophook (*Pkg*)   bophook is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{bophook}[2001/03/29]

2 \newcommand*{\AtBeginPage}[1]{}
3 \newcommand*{\PageLayout}[1]{}

File 58   `lwarp-bounddvi.sty`

§ 170   Package   # bounddvi

bounddvi (*Pkg*)   bounddvi is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{bounddvi}[2016/12/28]

File 59   `lwarp-boxedminipage.sty`

§ 171   Package   **boxedminipage**

(*Emulates or patches code by* SCOTT PAKIN.)

boxedminipage (*Pkg*)   boxedminipage is emulated for HTML, and used as-is for lateximages.

for HTML output: 1 \LWR@ProvidesPackagePass{boxedminipage}[2020/04/19]

```
2 \newenvironment{LWR@HTML@boxedminipage}{%
3     \LWR@stoppars%
4     \begin{BlockClass}{framebox}%
5     \minipage%
6 }
7 {%
8     \endminipage%
9     \end{BlockClass}%
10    \LWR@startpars%
11 }
12 \LWR@formattedenv{boxedminipage}
```

File 60   `lwarp-boxedminipage2e.sty`

§ 172   Package   **boxedminipage2e**

(*Emulates or patches code by* SCOTT PAKIN.)

boxedminipage2e (*Pkg*)   boxedminipage2e has been renamed boxedminipage by the author.

for HTML output: Automatically loads boxedminipage:

1 \LWR@ProvidesPackagePass{boxedminipage2e}

File 61   `lwarp-braket.sty`

§ 173   Package   **braket**

(*Emulates or patches code by* DONALD ARSENEAU.)

braket (*Pkg*)   braket works as-is for HTML with SVG math. For MATHJAX, the MATHJAX extension is used.

for HTML output: 1 \LWR@ProvidesPackagePass{braket}% No date is provided by the file.

```
2 \begin{warpMathJax}
3     \CustomizeMathJax{\require{braket}}
4 \end{warpMathJax}
```

File 62   **`lwarp-breakurl.sty`**

§ 174   Package   # breakurl

(*Emulates or patches code by* VILAR CAMARA NETO.)

breakurl (*Pkg*)   breakurl is emulated.

for HTML output: 1 \LWR@ProvidesPackageDrop{breakurl}[2013/04/10]

```
2 \LetLtxMacro\burl\LWR@url
3
4 \NewDocumentCommand{\LWR@burlaltb}{O{} +m m}{%
5     \LWR@ensuredoingapar%
6     \LWR@subhyperref{#2}%
```

If use \LWR@subhyperreftext@sanitized here, some forms of text may not expand correctly, and thus break.

```
7     \LWR@subhyperreftext@unsanitized{#3}%
8     \endgroup% restore catcodes
9 }
10
11 \newrobustcmd*{\burlalt}{%
12     \begingroup%
13     \LWR@linkcatcodes%
14     \LWR@burlaltb%
15 }
16
17 \LetLtxMacro\urlalt\burlalt
```

File 63   **`lwarp-breqn.sty`**

§ 175   Package   # breqn

(*Emulates or patches code by* MICHAEL J. DOWNES, MORTEN HØGHOLM.)

breqn (*Pkg*)   breqn is patched for use by lwarp.

⚠ darray   darray is not supported, and in fact does not work in the print version either.

While using MATHJAX, breqn objects are converted to SVG images.

for HTML output: 1 \LWR@ProvidesPackagePass{breqn}[2017/01/27]

```
2 \setkeys{breqn}{spread={5pt}}
3
4 \def\eqnumside{R}
5 % \def\eqnumplace{T}
6
7 \BeforeBeginEnvironment{dmath}{
8     \begin{BlockClass}{displaymathnumbered}
```

```
 9      \LWR@newautoidanchor%
10      \booltrue{LWR@indisplaymathimage}%
11      \begin{eximage}[-breqn dmath- \MathImageAltText]?%
12 }
13
14 \AfterEndEnvironment{dmath}{
15      \end{eximage}\end{BlockClass}
16 }
17
18 \BeforeBeginEnvironment{dmath*}{
19      \begin{BlockClass}{displaymath}
20      \LWR@newautoidanchor%
21      \booltrue{LWR@indisplaymathimage}%
22      \begin{eximage}[-breqn dmath*- \MathImageAltText]?%
23 }
24
25 \AfterEndEnvironment{dmath*}{
26      \end{eximage}\end{BlockClass}
27 }
28
29 \BeforeBeginEnvironment{dseries}{
30      \begin{BlockClass}{displaymathnumbered}
31      \LWR@newautoidanchor%
32      \booltrue{LWR@indisplaymathimage}%
33      \begin{eximage}[-breqn dseries- \MathImageAltText]?%
34 }
35
36 \AfterEndEnvironment{dseries}{
37      \end{eximage}\end{BlockClass}
38 }
39
40 \BeforeBeginEnvironment{dseries*}{
41      \begin{BlockClass}{displaymath}
42      \LWR@newautoidanchor%
43      \booltrue{LWR@indisplaymathimage}%
44      \begin{eximage}[-breqn dseries*- \MathImageAltText]?%
45 }
46
47 \AfterEndEnvironment{dseries*}{
48      \end{eximage}\end{BlockClass}
49 }
50
51 \BeforeBeginEnvironment{dgroup}{
52      \begin{BlockClass}{displaymath}
53      \LWR@newautoidanchor%
54      \booltrue{LWR@indisplaymathimage}%
55      \begin{eximage}[-breqn dgroup- \MathImageAltText]?%
56 }
57
58 \AfterEndEnvironment{dgroup}{
59      \end{eximage}\end{BlockClass}
60 }
61
62 \BeforeBeginEnvironment{dgroup*}{
63      \begin{BlockClass}{displaymath}
64      \LWR@newautoidanchor%
65      \booltrue{LWR@indisplaymathimage}%
66      \begin{eximage}[-breqn dgroup*- \MathImageAltText]?%
67 }
68
```

```
69 \AfterEndEnvironment{dgroup*}{
70     \end{lateximage}\end{BlockClass}
71 }
```

File 64   **`lwarp-bsheaders.sty`**

§ 176   Package   **bsheaders**

bsheaders (*Pkg*)   bsheaders is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{bsheaders}[1997/10/06]

File 65   **`lwarp-bussproofs.sty`**

§ 177   Package   **bussproofs**

(*Emulates or patches code by* Samuel R. Buss.)

bussproofs (*Pkg*)   bussproofs is used as-is for HTML, and emulated by MathJax's extension.

⚠ \DisplayProof   If not using MathJax, inline proofs with \DisplayMath must be placed inside a math expression.

If using MathJax, only the prooftree environment may be used, not \DisplayProof.

for HTML output: 1 \LWR@ProvidesPackagePass{bussproofs}% no date in file

```
2 \ifbool{mathjax}{
3     \CustomizeMathJax{\require{bussproofs}}
4
5     \NewEnviron{LWR@HTML@prooftree}%
6         {%
7             \boolfalse{LWR@HTMLsanitize@tmpb@removebackslash}%
8             \LWR@doequation{\BODY}{prooftree}%
9         }%
10        [\LWR@doendequation{prooftree}]
11    \LWR@formattedenv{prooftree}
12 }{% SVG HTML
13    \BeforeBeginEnvironment{prooftree}{%
14        \begin{lateximage}[-bussproofs-~\PackageDiagramAltText]?%
15    }
16    \AfterEndEnvironment{prooftree}{\end{lateximage}}
17 }
```

File 66   **`lwarp-bxpapersize.sty`**

§ 178   Package   **bxpapersize**

bxpapersize (*Pkg*)   bxpapersize is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{bxpapersize}[2017/10/08]

```
2 \providecommand*\papersizesetup{\bxpapersizesetup}
3 \newcommand*\bxpapersizesetup[1]{}
```

File 67   **`lwarp-bytefield.sty`**

§ 179   Package   # bytefield

(*Emulates or patches code by* SCOTT PAKIN.)

bytefield (*Pkg*)   bytefield is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{bytefield}[2017/09/15]

```
2 \BeforeBeginEnvironment{bytefield}{%
3     \begin{lateximage}[-bytefield-~\PackageDiagramAltText]?%
4 }
5
6 \AfterEndEnvironment{bytefield}{\end{lateximage}}
```

File 68   **`lwarp-cancel.sty`**

§ 180   Package   # cancel

cancel (*Pkg*)   cancel is used as-is for SVG math, and emulated for HTML text output.

for HTML output: 1 \LWR@origRequirePackage{lwarp-xcolor}% for \convertcolorspec
2 \LWR@ProvidesPackagePass{cancel}[2013/04/12]

\cancelto is math-only, so is used as-is.

\LWR@cancelcolor   {⟨*text*⟩} {⟨*color*⟩} {⟨*class*⟩} {⟨*colorstyle*⟩} {⟨*FormatWPstyle*⟩}

Add colors if not empty:

```
3 \newcommand{\LWR@cancelcolor}[5]{%
4     \ifcsempty{#2}%
5     {\InlineClass(#5){#3}{#1}}%
6     {\LWR@htmlspanclass[#5;#4:\LWR@origpound\LWR@tempcolor]{#3}{#1}}%
7 }
```

\cancel   {⟨*text*⟩}

```
8 \DeclareRobustCommand{\LWR@HTML@cancel}[1]{%
9     \begingroup%
10    \CancelColor%
11    \LWR@findcurrenttextcolor%
12    \color{black}%
13    \LWR@cancelcolor{#1}{LWR@tempcolor}{sout}{text-decoration-color}%
14        {text-decoration:line-through}%
15    \endgroup%
16 }
17 \LWR@formatted{cancel}%
18
19 \LetLtxMacro\bcancel\cancel
20 \LetLtxMacro\xcancel\cancel
```

For MathJax:

```
21 \begin{warpMathJax}
22 \PackageNoteNoLine{lwarp, cancel}{The MathJax v3 extension will be used}
23 \CustomizeMathJax{\require{cancel}}
24 \end{warpMathJax}
```

File 69 **`lwarp-canoniclayout.sty`**

§ 181   Package   # canoniclayout

canoniclayout (*Pkg*)   canoniclayout is ignored.

for HTML output: § \LWR@ProvidesPackageDrop{canoniclayout}[2011/11/05]

```
2 \newcommand*{\currentfontletters}{}
3 \newcommand*{\charactersperpage}{}
```

File 70   **`lwarp-caption.sty`**

§ 182   Package   # caption

(*Emulates or patches code by* AXEL SOMMERFELDT.)

caption (*Pkg*)   caption is patched for use by lwarp.

for HTML output:
```
1 \typeout{---}
2 \typeout{Packages lwarp and caption:}
3 \typeout{If a ``Missing \protect\begin\protect{document\protect}'' error occurs here,}
4 \typeout{try using: \space \protect\usepackage\protect{caption\protect}\space%
5     \protect\captionsetup{options}}
6 \typeout{instead of: \protect\usepackage[options]\protect{caption\protect}.}
7 \typeout{---}
8
9 \LWR@ProvidesPackagePass{caption}[2023/08/05]

10 \VerifyCommand[lwarp][caption]{\caption@iibox@}{AD79C5FACDA9F8F9977188D922E8AC12}
11
12 \long\def\caption@iibox@#1#2#3#4{%
13 %   \setbox\@tempboxa\hbox{#4}%
14   \caption@iiibox{#1}{#2}{#3}%
15 %      [\wd\@tempboxa]%
16     []%                                lwarp
17     [\captionbox@innerpos@default]%
18 %      {\unhbox\@tempboxa}%
19     {{#4}}%                            lwarp
20 }

21 \VerifyCommand[lwarp][caption]{\caption@iiiibox}{62FC9237FCA80F5A607BF02D88C61601}
22
23 \long\def\caption@iiiibox#1#2#3#4#5[#6][#7]#8{%
24   \begingroup
25   #1*% set \caption@position
26   \caption@iftop{%
27     \endgroup
```

```
28      \minipagefullwidth%                          lwarp
29      \parbox[t]{\linewidth}{%
30        #1\relax
31        \caption@setposition t%
32 %        #2%
33              {\caption#4{#5}}%
34 %      \captionbox@hrule
35 %      \csname caption@justification@#7\endcsname
36        #8%
37      }%
38 }{%
39      \endgroup

40 %      \parbox[b]{#6}{%
41      \minipagefullwidth%                          lwarp
42      \parbox[b]{\linewidth}{%                      lwarp
43        #1\relax
44        \caption@setposition b%
45 %        \csname caption@justification@#7\endcsname
46        #8%
47 %      \captionbox@hrule
48 %        #3
49              {\caption#4{#5}}%
50      }%
51    }%
52 }
```

\caption@makecaption

```
53 \VerifyCommand[lwarp][caption]{\caption@makecaption}{9E0A92DF71E248B2C7A3B4BB5190A2C5}
54
55 \long\def\caption@makecaption#1#2{%
56 %    \caption@make@above
57    \caption@@make{#1}{#2}%
58 %    \caption@make@below
59 }
60
61 \AtBeginDocument{
62      \let\@makecaption\caption@makecaption
63 }
```

Appended to look ahead to the next token for \centering, etc:

```
64 \AtBeginDocument{
65 \xapptocmd{\@xfloat}
66      {\LWR@futurenonspacelet\LWR@mynexttoken\LWR@floatalignment}
67      {}
68      {\LWR@patcherror{caption}{@xfloat}}
69
70 \xapptocmd{\@xdblfloat}
71      {\LWR@futurenonspacelet\LWR@mynexttoken\LWR@floatalignment}
72      {}
73      {\LWR@patcherror{caption}{@xdblfloat}}
74 }
75
76 \VerifyCommand[lwarp][caption]{\caption@@@text}{C7253081E4F8EA695FF193E21855AA0A}
76
77 \long\def\caption@@@text#1#2#3[#4]#5{%
78      \begin{BlockClass}{figurecaption}%         lwarp
```

```
79    \begingroup
80      #3{\csname c@#1\endcsname #4\relax}%
81      #2{\caption@fnum{#1}}{#5}%
82    \endgroup%
83      \end{BlockClass}%                              lwarp
84 }
```

Updates for late patches for scrextend:

```
85 \caption@AtBeginDocument{
86 \IfPackageLoadedTF{lwarp-scrextend}{
87    \LetLtxMacro\captionbelow\caption
88    \LetLtxMacro\captionabove\caption
89    \LetLtxMacro\captionofbelow\captionof
90    \LetLtxMacro\captionofabove\captionof
91 }{}
92 }
```

File 71   **`lwarp-caption3.sty`**

§ 183   Package   **caption3**

(*Emulates or patches code by* AXEL SOMMERFELDT.)

caption3 (*Pkg*)   caption3 is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{caption3}[2023/07/31]

\caption@@@make   {⟨*caption label*⟩} {⟨*caption text*⟩}

```
2 \IfPackageAtLeastTF{caption3}{2020/08/23}{
3 %
4 \VerifyCommand[lwarp][caption3]{\caption@@@make}{F09A9BB05CE4EDF5A477D3CC2AE04F81}
5 %
6 \renewcommand\caption@@@make[2]{%
7 \LWR@traceinfo{caption@@@make}%

8    \LWR@stoppars%                                     lwarp

9 %  \ifx\caption@fmt\@undefined\caption@format\fi
10 %  \let\caption@lfmt\caption@labelformat
11 %  \global\def\caption@tempa{gobbletwo}%
12 %  \global\def\caption@tempb{}%
13 %  \sbox\@tempboxa{%
14 %    \let\caption@ignorespaces\ignorespaces
15 %    \def\ignorespaces{%
16 %      \global\def\caption@tempb{two}% "gobble" -> "gobbletwo"
17 %                            % if \ignorespaces is used addtionally
18 %      \caption@ignorespaces}%
19 %    #1%
20 %      {\global\let\caption@tempa\@undefined\aftergroup\@gobble}%
21 %      {\global\def\caption@tempa{gobble\caption@tempb}}}%
22 %  \ifdim\wd\@tempboxa=\z@
23 %    \gdef\caption@tempa{none}%
24 %  \fi
25 %  \ifx\caption@tempa\@undefined \else
26 %    \@expandtwoargs\caption@set{labelseparator}{\caption@tempa}%
```

```
27 %   \fi
28    \caption@ifempty{#2}{%
29      \caption@set{labelseparator}{none}%
30      \caption@set{textformat}{simple}%
31    }%
32    \caption@labelseparator % defines \caption@iflabelfont,
33 %       \caption@labelsep and \caption@labelsep@name
34 %       (the latter is needed by \caption@fmt)
35 %
36 %    \@setpar{\@@par\caption@@par}\caption@@par
37    \caption@applyfont
```

\caption@fmt with plain format is defined as {#1#2#3\par}:

```
38 %    \caption@fmt
39    {\ifcaption@star\else
40      \begingroup
41        \captionlabelfont
42        \LWR@isolate{#1}%                          lwarp
43      \endgroup
44     \fi}%
45    {\ifcaption@star\else
46      \begingroup
47        \caption@iflabelfont\captionlabelfont
48        \relax\caption@labelsep
49      \endgroup
50     \fi}%
51    {{\captiontextfont
52        \let\\\newline%                            lwarp
53 %
54      \caption@textstart
55 %      \caption@ifstrut
56 %        {\vrule\@height\ht\strutbox\@width\z@}%
57 %        {}%
58 %      \nobreak\hskip\z@skip % enable hyphenation
59        \LWR@isolate{\caption@textformat{#2}}%     lwarp
60 %        \caption@ifstrut
61 %          {\ifhmode\@finalstrut\strutbox\fi}%
62 %          {}%
63        \caption@textend}}%
64      \LWR@startpars%                              lwarp
65 \LWR@traceinfo{caption@@@make done}%
66 }
67 }% later than 2020/08/23
68 {% earlier than 2020/08/23
69 \renewcommand\caption@@@make[2]{%
70 \LWR@traceinfo{caption@@@make}%
71      \LWR@stoppars%                               lwarp
72 %    \sbox\@tempboxa{#1}%
73 %    \ifdim\wd\@tempboxa=\z@
74 %      \let\caption@lsep\relax
75 %    \fi
76    \caption@ifempty{#2}{%
77      \let\caption@lsep\@empty
78      \let\caption@tfmt\@firstofone
79    }%
```

```
80 %    \@setpar{\@@par\caption@@par}\caption@@par
81    \caption@applyfont
```

\caption@fmt with plain format is defined as {#1#2#3\par}:

```
82 %      \caption@fmt
83      {\ifcaption@star\else
84        \begingroup
85         \captionlabelfont
86          \LWR@isolate{#1}%                        lwarp
87        \endgroup
88      \fi}%
89      {\ifcaption@star\else
90        \begingroup
91          \caption@iflf\captionlabelfont
92          \relax
93          \caption@lsep
94        \endgroup
95      \fi}%
96      {{%
97          \captiontextfont
98          \let\\\newline%                          lwarp
99 %        \caption@ifstrut
100 %          {\vrule\@height\ht\strutbox\@width\z@}%
101 %          {}%
102 %        \nobreak\hskip\z@skip % enable hyphenation
103          \LWR@isolate{\caption@tfmt{#2}}%        lwarp
104 %        \caption@ifstrut
105 %          {\ifhmode\@finalstrut\strutbox\fi}%
106 %          {}%
107        }}%
108      \LWR@startpars%                              lwarp
109 \LWR@traceinfo{caption@@make done}%
110 }
111 }% earlier than 2020/08/23
```

\caption@@make@   {⟨⟩}{⟨⟩}

```
112 \VerifyCommand[lwarp][caption3]{\caption@@make@}{AD348E907B8F8B0BCAE57E72DE4A2035}
113 %
114 \renewcommand{\caption@@make@}[2]{%
115   \caption@stepthecounter%
116   \caption@beginhook%
117 %   \caption@box\hsize{%
118 %   \caption@singlelinecheck{\caption@slc{#1}{#2}\caption@singleline\caption@multiline}{\caption@mu
119 %     \caption@calcmargin
120 %     \caption@make@leftmargin
121 %     \caption@make@parbox{%
122 %       \caption@make@indention
123        \caption@@@make{#1}{#2}%
124 %     }
125 %     \caption@make@rightmargin
126 %   }%
127   \caption@endhook%
128 }

129 \DeclareCaptionBox{none}{#2}
130 \DeclareCaptionBox{parbox}{%
131     #2%
132 }
```

```
133 \DeclareCaptionBox{colorbox}{%
134    #2%
135 }
```

File 72   **`lwarp-cases.sty`**

§ 184   Package   # cases

(*Emulates or patches code by* DONALD ARSENEAU.)

cases (*Pkg*)   cases is patched for use by lwarp.

While using MATHJAX, cases objects are converted to SVG math images. The Math-Jax 3.2 cases package does not yet work with lwarp.

for HTML output:
```
1 \LWR@ProvidesPackagePass{cases}[2020/03/29]
```

```
2 \BeforeBeginEnvironment{numcases}{
3    \begin{BlockClass}{displaymathnumbered}
4    \LWR@newautoidanchor%
5    \booltrue{LWR@indisplaymathimage}%
6    \begin{lateximage}[-cases- \MathImageAltText]?%
7 }
8
9 \AfterEndEnvironment{numcases}{
10    \end{lateximage}\end{BlockClass}
11 }
12
13 \BeforeBeginEnvironment{subnumcases}{
14    \begin{BlockClass}{displaymathnumbered}
15    \LWR@newautoidanchor%
16    \booltrue{LWR@indisplaymathimage}%
17    \begin{lateximage}[-cases- \MathImageAltText]?%
18 }
19
20 \AfterEndEnvironment{subnumcases}{
21    \end{lateximage}\end{BlockClass}
22 }
```

File 73   **`lwarp-ccicons.sty`**

§ 185   Package   # ccicons

(*Emulates or patches code by* MICHAEL UMMELS.)

ccicons (*Pkg*)   ccicons is used as SVG images for HTML.

for HTML output: Discard all options for lwarp-ccicons:

```
1 \LWR@ProvidesPackagePass{ccicons}[2017/10/30]
```

```
2 \newcommand{\LWR@ccicons}[2]{%
3    {\begin{lateximage}*[#1]?\ccicons@font\char#2\end{lateximage}}%
4 }
```

```
5 \renewcommand{\ccicons@logo}{\LWR@ccicons{ccLogo}{0}}
6 \renewcommand{\ccicons@by}{\LWR@ccicons{ccAttribution}{1}}
7 \renewcommand{\ccicons@sa}{\LWR@ccicons{ccShareAlike}{2}}
8 \renewcommand{\ccicons@nd}{\LWR@ccicons{ccNoDerivatives}{3}}
9 \renewcommand{\ccicons@nc}{\LWR@ccicons{ccNonCommercial}{4}}
10 \renewcommand{\ccicons@nceu}{\LWR@ccicons{ccNonCommercialEU}{5}}
11 \renewcommand{\ccicons@ncjp}{\LWR@ccicons{ccNonCommercialJP}{6}}
12 \renewcommand{\ccicons@pd}{\LWR@ccicons{ccPublicDomain}{7}}
13 \renewcommand{\ccicons@zero}{\LWR@ccicons{ccZero}{8}}
14 \renewcommand{\ccicons@sampling}{\LWR@ccicons{ccSampling}{9}}
15 \renewcommand{\ccicons@share}{\LWR@ccicons{ccShare}{10}}
16 \renewcommand{\ccicons@remix}{\LWR@ccicons{ccRemix}{11}}
17 \renewcommand{\ccicons@copy}{\LWR@ccicons{ccCopy}{12}}
18 \renewcommand{\ccicons@pdalt}{\LWR@ccicons{ccPublicDomainAlt}{13}}
```

File 74   `lwarp-centerlastline.sty`

§ 186   Package   # centerlastline

centerlastline (*Pkg*)   centerlastline is ignored.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{centerlastline}[2020/10/12]
```

```
2 \providecommand{\centerlastline}{}
3 \def\endcenterlastline{\par}
```

File 75   `lwarp-centernot.sty`

§ 187   Package   # centernot

(*Emulates or patches code by* Heiko Oberdiek.)

centernot (*Pkg*)   centernot is used as-is for svg math, and emulated for Mathjax.

for HTML output:
```
1 \LWR@ProvidesPackagePass{centernot}[2016/05/16]
```

```
2 \begin{warpMathJax}
3 \CustomizeMathJax{\require{centernot}}
4 \end{warpMathJax}
```

File 76   `lwarp-changebar.sty`

§ 188   Package   # changebar

changebar (*Pkg*)   changebar is ignored.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{changebar}[2018/03/09]
```

```
2 \newcommand*{\cbstart}{}
3 \newcommand*{\cbend}{}
4 \newenvironment*{\changebar}{}{}
```

```
 5 \newcommand*{\cbdelete}{}
 6 \newcommand*{\nochnagebars}{}
 7 \newcommand*{\cbcolor}[1]{}
 8 \newlength{\changebarwidth}
 9 \newlength{\deletebarwidth}
10 \newlength{\changebarsep}
11 \newcounter{changebargrey}
```

File 77   **`lwarp-changelayout.sty`**

§ 189   Package   # changelayout

(*Emulates or patches code by* AHMED MUSA.)

changelayout (*Pkg*)   changelayout is patched for use by lwarp.

for HTML output: `1 \LWR@ProvidesPackagePass{changelayout}[2009/10/07]`

```
 2 \renewrobustcmd\cpl@backtodefaults{}
 3
 4 \renewrobustcmd\cpl@checkifoddpage{%
 5   \cpl@oddpagefalse%
 6 }
 7
 8 \renewrobustcmd\changepagelayout[1]{%
 9   \setkeys[KV]{changelay}{#1}%
10 }
11
12 \renewrobustcmd{\changetextlayout}[1]{\changepagelayout{#1}}
13
14 \renewrobustcmd\adjustpagelayout[1]{%
15   \setkeys[KV@X]{changelay}{#1}%
16 }
17
18 \renewrobustcmd{\adjusttextlayout}[1]{\adjustpagelayout{#1}}
19
20 \renewrobustcmd\adjusttextwidth[1]{%
21   \setkeys[KV]{changelay}{#1}%
22   \begin{BlockClass}[color:\LWR@colorstyle{named}{\cpl@textcolor}]{changelayout}
23       \color{\cpl@textcolor}%
24       \cpl@content
25   \end{BlockClass}
26 }
```

File 78   **`lwarp-changepage.sty`**

§ 190   Package   # changepage

(*Emulates or patches code by* PETER WILSON.)

changepage (*Pkg*)   changepage is ignored.

for HTML output: Discard all options for lwarp-changepage:

`1 \LWR@ProvidesPackageDrop{changepage}[2009/10/20]`

```
2 \newif\ifoddpage
3 \DeclareRobustCommand{\checkoddpage}{\oddpagetrue}
4 \DeclareRobustCommand{\changetext}[5]{}
5 \DeclareRobustCommand{\changepage}[9]{}
6
7 \@ifundefined{adjustwidth}{
8 \newenvironment{adjustwidth}[2]{}{}
9 \newenvironment{adjustwidth*}[2]{}{}
10 }{
11 \renewenvironment{adjustwidth}[2]{}{}
12 \renewenvironment{adjustwidth*}[2]{}{}
13 }


14 \DeclareDocumentCommand{\strictpagecheck}{}{}
15 \DeclareDocumentCommand{\easypagecheck}{}{}
```

File 79   **`lwarp-changes.sty`**

§ 191   Package   # changes

(*Emulates or patches code by* EKKART KLEINOD.)

changes (*Pkg*)   changes is patched for use by lwarp.

⚠   \comment   Use commandnameprefix=ifneeded to avoid a conflict with the \comment command when using lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{changes}[2021/07/15]

\BaseJobname is added to the label in case xr or xr-hyper are used.

```
2 \renewcommand{\ChangesListline}[4]{%
3     \IfIsInList{#1}{\Changes@loc@show}{%
4         \LWR@startpars%
5         #2: #3 \qquad
6         \nameref{\BaseJobname-autopage-#4}%
7         \LWR@stoppars%
8     }{}%
9 }
10
11 \VerifyCommand[lwarp][changes]{\listofchanges}{CDE77F21854A8C807FDF2CF756286B55}
12
13 \renewcommand{\listofchanges}[1][\@empty]{%
14 \setkeys{Changes@loc}{#1}%
15 \ifbool{Changes@optiondraft}%
16 {%
17 \IfIsInList{\Changes@loc@style}{list|summary|compactsummary}%
18 {}%
19 {%
20 \PackageWarning{changes}{Wrong style for list of changes:%
21     '\Changes@loc@style', using 'list' instead.}%
22 \def\Changes@loc@style{}%
23 }%
24 \IfIsEmpty{\Changes@loc@style}%
25 {\def\Changes@loc@style{list}}%
26 {}%
```

```
27 \IfStrEq{\Changes@loc@show}{all}%
28 {\def\Changes@loc@show{added|deleted|replaced|highlight|comment}}%
29 {}%
30 \IfIsInList{\Changes@loc@show}{added|deleted|replaced|highlight|comment}%
31 {}%
32 {%
33 \PackageWarning{changes}{Wrong show-value for list of changes: '\Changes@loc@show', using 'all' instead
34 \def\Changes@loc@show{}%
35 }%
36 \IfIsEmpty{\Changes@loc@show}%
37 {\def\Changes@loc@show{added|deleted|replaced|highlight|comment}}%
38 {}%
39 \IfIsEmpty{\Changes@loc@title}%
40 {%
41 \IfStrEq{\Changes@loc@style}{list}%
42 {\def\Changes@heading{\listofchangesname}}{}%
43 \IfStrEq{\Changes@loc@style}{summary}%
44 {\def\Changes@heading{\summaryofchangesname}}{}%
45 \IfStrEq{\Changes@loc@style}{compactsummary}%
46 {\def\Changes@heading{\compactsummaryofchangesname}}{}%
47 }%
48 {\def\Changes@heading{\Changes@loc@title}}%
49 \section*{\Changes@heading}
50 \IfIsInList{\Changes@loc@style}{list}%
51 {%
52 \IfFileExists{\jobname.\Changes@locextension}%
53 {%
54 \newread\Changes@InFile%
55 \openin\Changes@InFile=\jobname.\Changes@locextension%
56 \loop\unless\ifeof\Changes@InFile%
57 \read\Changes@InFile to \Changes@Line%
58 \ifeof\Changes@InFile\else%
59 \Changes@Line%
60 \fi
61 \repeat
62 \closein\Changes@InFile%
63 }{%
64 \emph{\changesnoloc}%
65 \PackageWarning{changes}{LaTeX rerun needed for list of changes}%
66 }%
67 }{}%
68 \IfIsInList{\Changes@loc@style}{summary|compactsummary}%
69 {%
70 \IfFileExists{\jobname.\Changes@socextension}%
71 {%
72 \newread\Changes@InFile%
73 \openin\Changes@InFile = \jobname.\Changes@socextension%
74 \loop\unless\ifeof\Changes@InFile%
75 \read\Changes@InFile to \Changes@Line%
76 \ifeof\Changes@InFile\else%
77 \expandafter\changes@chopline\Changes@Line\\%
78 \textbf{%
79 \IfIsColored%
80 {\color{\Changes@Incolor}}%
81 {}%
82 \IfIsAnonymous{\Changes@Inid}%
83 {%
84     \LWR@textcurrentcolor{%      lwarp
85         \changesauthorname: \changesanonymousname%
86     }%  lwarp
```

```
87 }%
88 {%
89     \LWR@textcurrentcolor{%     lwarp
90 \changesauthorname: \Changes@Inid%
91     }%  lwarp
92 \IfIsEmpty{\Changes@Inname}%
93 {}%
94 { %
95     \LWR@textcurrentcolor{%      lwarp
96 (\Changes@Inname)%
97     }%  lwarp
98 }%
99 }%
100 }\\%
101 \numdef{\Changes@InSum}{0}%
102 \renewcommand*{\do}[1]{%
103 \numdef{\Changes@InSum}{\Changes@InSum + \csuse{Changes@In########1}}%
104 }%
105 \expandafter\dopsvlist\expandafter{\Changes@loc@show}%
106 \ifnumcomp{\Changes@InSum}{=}{0}%
107 {%
108 % \parbox{\Changes@summary@width}{% lwarp
109     \changesnochanges%
110 % }%   lwarp
111 % \\[1ex]%    lwarp
112     \par%  \lwarp
113 }%
114 {%
115 \numdef{\Changes@InCount}{0}%
116 \renewcommand*{\do}[1]{%
117 \numdef{\Changes@InCount}{\Changes@InCount + \csuse{Changes@In########1}}%
118 \ifboolexpr{%
119 not test {\IfStrEq{\Changes@loc@style}{compactsummary}} or%
120 test {\ifnumgreater{\csuse{Changes@In########1}}{0}}%
121 }%
122 {%
123 % \parbox{\Changes@summary@width}{%  lwarp
124 \csuse{changes########1name}~%
125 % \let\cleaders\leaders\dotfill~%    lwarp
126 \dotfill~%   \lwarp
127 \csuse{Changes@In########1}%
128 % }%    lwarp
129 % \ifnumless{\Changes@InCount}{\Changes@InSum}%      lwarp
130 {\\}%
131 % {\\[1ex]}%    lwarp
132 }%
133 {}%
134 }%
135 \expandafter\dopsvlist\expandafter{\Changes@loc@show}%
136     \par%   lwarp
137 }%
138 \fi%
139 \repeat
140 \closein\Changes@InFile%
141 }{%
142 \emph{\changesnosoc}%
143 \PackageWarning{changes}{LaTeX rerun needed for summary of changes}%
144 }%
145 }{}%
146 }{}%
```

```
147 }
148
149 \VerifyCommand[lwarp][changes]{\Changes@Markup@comment}{23057A40141C9D1A0A173DCF1BD5DE55}
150
151 \renewcommand{\Changes@Markup@comment}[3]{%
152 \IfStrEq{\Changes@optioncommentmarkup}{todo}%
153 {%
154 \IfIsColored%
155 {\colorlet{Changes@todocolor}{authorcolor}}%
156 {\colorlet{Changes@todocolor}{black}}%
157 \todo[color=Changes@todocolor!10, bordercolor=Changes@todocolor, linecolor=Changes@todocolor!70, noli
158 }{}%
159 \IfStrEq{\Changes@optioncommentmarkup}{margin}%
160 {%
161 \marginpar{%
162 \IfIsColored%
163 {\leavevmode\color{authorcolor}}%
164 {}%
165     \LWR@textcurrentcolor{%        lwarp
166 \textbf{[\IfIsAnonymous{#2}{}{#3~}\arabic{Changes@commentCount#2}]:} #1%
167     }%  lwarp
168 }%
169 }{}%
170 \IfStrEq{\Changes@optioncommentmarkup}{footnote}%
171 {%
172 \footnote{%
173     \LWR@textcurrentcolor{%        lwarp
174 \textbf{[\IfIsAnonymous{#2}{}{#3~}\arabic{Changes@commentCount#2}]:} #1%
175     }%  lwarp
176 }%
177 }{}%
178 \IfStrEq{\Changes@optioncommentmarkup}{uwave}%
179 {%
180 {%
181 \IfIsColored%
182 {\color{authorcolor}}%
183 {}%
184 \allowbreak%
185 \uwave{%
186 \textbf{[\IfIsAnonymous{#2}{}{#3~}\arabic{Changes@commentCount#2}]:} #1%
187 }%
188 }%
189 }{}%
190 }
191
192 \VerifyCommand[lwarp][changes]{\Changes@output}{BD1ACDECB4BBA2D9181885F9EDC87F77}
193
194 \renewrobustcmd\Changes@output[7]{%
195 \ifbool{Changes@optiondraft}%
196 {%
197 \Changes@check@author{#2}%
198 \Changes@set@color{#2}%
199 {%
200 \IfIsInList{#1}{added|deleted|replaced|highlight}%
201 {%
202 \IfIsEmpty{#5}%
203 {%
204 \IfIsAuthorEmptyAtPosition{#2}{left}%
205 {}%
206 {{%
```

```
207 \IfIsColored%
208 {\color{authorcolor}}%
209 {}%
210     \LWR@textcurrentcolor{%      lwarp
211 \Changes@Markup@author{\Changes@output@author@position{#2}{left}}}%
212     }%  lwarp
213 }}%
214 }{}%
215 {%
216 \IfStrEq{#1}{highlight}%
217 {}{%
218 \IfIsColored%
219 {\color{authorcolor}}%
220 {}%
221 }%
222     \LWR@textcurrentcolor{%      lwarp
223 \IfStrEq{#1}{added}{\Changes@Markup@added{#3}}{}%
224 \IfStrEq{#1}{deleted}{\Changes@Markup@deleted{#4}}{}%
225 \IfStrEq{#1}{replaced}{{\Changes@Markup@added{#3}}\allowbreak\Changes@Markup@deleted{#4}}{}%
226 \IfStrEq{#1}{highlight}{\Changes@Markup@highlight{#3}}{}%
227     }%  lwarp
228 }%
229 \IfIsEmpty{#5}%
230 {%
231 \IfIsAuthorEmptyAtPosition{#2}{right}%
232 {}%
233 {{%
234 \IfIsColored%
235 {\color{authorcolor}}%
236 {}%
237     \LWR@textcurrentcolor{%      lwarp
238 \Changes@Markup@author{\Changes@output@author@position{#2}{right}}}%
239     }%  lwarp
240 }}%
241 }{}%
242 \stepcounter{Changes@#1Count#2}%
243 }{}%
244 \IfIsEmpty{#5}%
245 {}%
246 {%
247 \stepcounter{Changes@commentCount#2}%
248 \Changes@set@commentcount{#2}%
249 \Changes@Markup@comment%
250 {#5}%
251 {#2}%
252 {\Changes@output@author{#2}}%
253 }%
254 }%
255 \IfIsEmpty{#2}%
256 {\def\Changes@locid{}}%
257 {\def\Changes@locid{~(#2)}}%
258 \addtocontents{\Changes@locextension}{\protect\ChangesListline{#1}{#6\Changes@locid}{#7}{\thepage}}%
259 }%
260 {%
261 \IfIsEmpty{#3}%
262 {\@bsphack\@esphack}%
263 {#3}%
264 }%
265 }
```

File 80   **lwarp-chappg.sty**

§ 192   Package   # chappg

(*Emulates or patches code by* ROBIN FAIRBAIRNS.)

chappg (*Pkg*)   chappg is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{chappg}[2006/05/09]

2 \renewcommand{\pagenumbering}[2][]{}
3 \providecommand{\chappgsep}{--}

File 81   **lwarp-chapterbib.sty**

§ 193   Package   # chapterbib

(*Emulates or patches code by* DONALD ARSENEAU.)

chapterbib (*Pkg*)   chapterbib is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{chapterbib}[2010/09/18]

2 \xdef\@savedjobname{\BaseJobname}
3 \let\@currentipfile\@savedjobname

File 82   **lwarp-chemfig.sty**

§ 194   Package   # chemfig

(*Emulates or patches code by* CHRISTIAN TELLECHEA.)

chemfig (*Pkg*)   chemfig is patched for use by lwarp.

If using \polymerdelim to add delimiters to a \chemfig, wrap both inside a single lateximage:

```
\begin{lateximage}[-chemfig-~\PackageDiagramAltText]
\chemfig{. . . }
\polymerdelim[. . . ]{. . . }
\end{lateximage}
```

The images are not hashed because they depend on external settings which may be changed at any time, and are unlikely to be reused inline anyhow.

for HTML output: 1 \LWR@ProvidesPackagePass{chemfig}[2021/02/28]

2 \catcode`_=11
3
4 \IfPackageAtLeastTF{chemfig}{2020/03/05}

```
 5 {
 6    \xpretocmd\charge{\begin{lateximage}[-chemfig-~\PackageDiagramAltText]?}%
 7        {}{\LWR@patcherror{chemfig}{charge}}
 8    \xpretocmd\Charge{\begin{lateximage}[-chemfig-~\PackageDiagramAltText]?}%
 9        {}{\LWR@patcherror{chemfig}{Charge}}
10    \xapptocmd\charge_c{\end{lateximage}}
11        {}{\LWR@patcherror{chemfig}{charge_c}}
12 }{}
13
14 \IfPackageAtLeastTF{chemfig}{2019/04/18}%
15 {% 2019/04/18 or newer
16
17    \xpretocmd{\CF_chemfiga}
18        {\begin{lateximage}[-chemfig-~\PackageDiagramAltText]?}%
18        {}{\LWR@patcherror{chemfig}{CF_chemfiga}}
19
20    \VerifyCommand[lwarp][chemfig]{\CF_chemfigb}{7B199210755F37B1BCD036567614BA34}
21
22    \xpatchcmd{\CF_chemfigb}
23        {\let\CF_flipstate\CF_zero}
24        {\end{lateximage}\let\CF_flipstate\CF_zero}
25        {}{\LWR@patcherror{chemfig}{CF_chemfigb}}
26
27    \GlobalLetLtxMacro\LWR@chemfig@origCF_lewisc\CF_lewisc
28    \gdef\CF_lewisc#1,#2\_nil{%
29    \begin{lateximage}[-chemfig-~\PackageDiagramAltText?]%
30    \LWR@chemfig@origCF_lewisc#1,#2\_nil
31    \end{lateximage}
32    }
33
34    \gpreto{\schemestart}{%
35        \begin{lateximage}[-chemfig-~\PackageDiagramAltText]?%
36    }
37    \gappto{\CF_schemestop}{\end{lateximage}}
38
39 }% 2019/04/18 or newer
40 {% older than 2019/04/18
41
42    \LetLtxMacro\LWR@chemfig@origchemfig\chemfig
43
44    \DeclareDocumentCommand\chemfig{s O{} O{} m}{%
45        \begin{lateximage}[-chemfig-~\PackageDiagramAltText]?%
46        \IfBooleanTF{#1}{%
47            \LWR@chemfig@origchemfig*[#2][#3]{#4}%
48        }{%
49            \LWR@chemfig@origchemfig[#2][#3]{#4}%
50        }
51        \end{lateximage}%
52    }
53
54    \LetLtxMacro\LWR@chemfig@origCF@lewis@b\CF@lewis@b
55
56    \def\CF@lewis@b#1#2{%
57    \begin{lateximage}[-chemfig-~\PackageDiagramAltText]?%
58    \LWR@chemfig@origCF@lewis@b{#1}{#2}%
59    \end{lateximage}%
60    }
61
62    \preto{\schemestart}{%
63        \begin{lateximage}[-chemfig-~\PackageDiagramAltText]?%
```

```
64     }
65     \appto{\CF@schemestop}{\end{lateximage}}
66
67 }% older than 2019/04/18
68
69 \catcode`\_=8%
70
71
72
73 \LetLtxMacro\LWR@chemfig@origchemleft\chemleft
74
75 \def\chemleft#1#2\chemright#3{%
76 \begin{lateximage}[-chemfig-~\PackageDiagramAltText]?%
77 \LWR@chemfig@origchemleft#1#2\chemright#3%
78 \end{lateximage}%
79 }
80
81 \LetLtxMacro\LWR@chemfig@origchemup\chemup
82
83 \def\chemup#1#2\chemdown#3{%
84 \begin{lateximage}[-chemfig-~\PackageDiagramAltText]?%
85 \LWR@chemfig@origchemup#1#2\chemdown#3%
86 \end{lateximage}%
87 }
```

File 83   **`lwarp-chemformula.sty`**

§ 195   Package   # chemformula

(*Emulates or patches code by* CLEMENS NIEDERBERGER.)

chemformula (*Pkg*)   chemformula is patched for use by lwarp.

The svg images are hashed according to contents and local options. Global options are assumed to be constant document-wide.

⚠ chemformula with   chemformula works best without MATHJAX. If MATHJAX is used, \displaymathother
MATHJAX   must be used before array, and then \displaymathnormal may be used after. (The chemformula package adapts to array, but does not know about MATHJAX, and MATHJAX does not know about chemformula.)

While using MATHJAX, \displaymathother may also be used for other forms of display and inline math which contain chemformula expressions.

for HTML output: 1 \LWR@ProvidesPackagePass{chemformula}[2022/01/23]

2 \ExplSyntaxOn

\ch   Enclose in an inline svg image or MATHJAX. The alt tag is is the contents of the \ch expression. The filename is hashed, and also has additional hashing information based on the local options.

3 \VerifyCommand[lwarp][chemformula]{\ch}{6D4331F95E9FF2E011E310B1F8C5487E}
4
5 \RenewDocumentCommand \ch { O{}m }
6   {%

To work inside align with \displaymathother, a simple version must be used to work with chemformula's adaptation to align.

```
7    \ifnumcomp{\value{LWR@laterimagedepth}}{>}{0}%  lwarp
8    {%
9        \chemformula_ch:nn {#1} {#2}%   original
10   }%
```

If used as the outer level, must temporarily ensure MATHJAX is disabled:

```
11   {%
12       \begingroup%
13       \boolfalse{mathjax}%
```

An inline image is used, adjusted for the baseline:

```
14       \LWR@subsingledollar*{% lwarp
15           \textbackslash{}%
16           ch%
17           \{%
18               \LWR@HTMLsanitizedetokenized{\detokenize{#2}}%
19           \}% alt text
20       }{%
21         \protect\LWR@HTMLsanitizedetokenized{%
22             \detokenize\expandafter{#1}%
23         }% add'l hashing
24       }%
25       {%
26           \chemformula_ch:nn {#1} {#2}%   original
27       }%
28       \endgroup%
29   }%
30   }
```

\chcpd   Similar to \ch.

```
31 \IfPackageAtLeastTF{chemformula}{2019/10/13}{
32
33 \VerifyCommand[lwarp][chemformula]{\chemformula_chcpd:nn}
34    {C1E882F2C1137D429AE4F789C84E7428}
35
36 \cs_gset_protected:Npn \chemformula_chcpd:nn #1#2
37   {
38     \begingroup%
39     \boolfalse{mathjax}%
40     \LWR@subsingledollar*{% lwarp
41         \textbackslash{}% alt tag
42         chcpd%
43         \{%
44             \LWR@HTMLsanitizedetokenized{\detokenize{#2}}%
45         \}%
46     }{%
47         \protect\LWR@HTMLsanitizedetokenized{\detokenize\expandafter{#1}}%
48     }{% original
49     \group_begin:
50       \tl_if_blank:nF {#2}
51       {
52         \keys_set:nn {chemformula} {#1}
53         \__chemformula_save_catcodes:
54         \__chemformula_sanitize:Nn
55           \l__chemformula_chemformula_tmpa_tl
56           {#2}
```

```
57            \__chemformula_input_compound_no_check:NV
58              \l__chemformula_compound_tl
59              \l__chemformula_chemformula_tmpa_tl
60            \__chemformula_prepare_output:NV
61              \l__chemformula_compound_tl
62              \l__chemformula_catcodes_tl
63            \chemformula_write:V \l__chemformula_compound_tl
64          }
65      \group_end:
66      }
67      \endgroup
68    }
69 }% later than 2019/10/13
70 {% earlier than 2019/10/13
71 % \changes{v0.903}{2021/12/18}{\pkg{chemformula}: Improved alt tag sanitization.}
72 \cs_gset_protected:Npn \chemformula_chcpd:nn #1#2
73   {
74      \begingroup%
75      \boolfalse{mathjax}%
76      \LWR@subsingledollar*{% lwarp
77          \textbackslash{}% alt tag
78          chcpd%
79          \{%
80              \LWR@HTMLsanitizedetokenized{\detokenize{#2}}%
81          \}%
82      }{%
83          \protect\LWR@HTMLsanitizedetokenized{\detokenize\expandafter{#1}}%
84      }{% original
85      \group_begin:
86        \tl_if_blank:nF {#2}
87          {
88            \keys_set:nn {chemformula} {#1}
89            \__chemformula_save_catcodes:
90            \__chemformula_sanitize:Nn
91              \l__chemformula_chemformula_tmpa_tl
92              {#2}
93            \__chemformula_input_compound_no_check:NV
94              \l__chemformula_compound_tl
95              \l__chemformula_chemformula_tmpa_tl
96            \__chemformula_prepare_output:N \l__chemformula_compound_tl
97            \chemformula_write:V \l__chemformula_compound_tl
98          }
99      \group_end:
100      }
101      \endgroup
102   }
103 }% earlier than 2019/10/13
```

\charrow   If standalone, appears in a regular lateximage.

```
104 \VerifyCommand[lwarp][chemformula]{\charrow}
105      {31D2B3405541C0B128504C94C5046713}
106
107 \RenewDocumentCommand \charrow { mO{}O{} }
108 {
109      \begin{lateximage}[-chemformula- charrow]
110      \group_begin:
111        \__chemformula_draw_arrow:nnn {#1} {#2} {#3}
112      \group_end:
113      \end{lateximage}
```

114 }

\chname   If standalone, appears in a regular lateximage, hashed according to contents.

115 \VerifyCommand[lwarp][chemformula]{\chname}
116   {3C697C09415EE601DE035EEDD6D3BA4D}
117
118 \RenewDocumentCommand \chname { R(){}R(){} }
119   {
120   \begin{lateximage}*[%
121   \textbackslash{}%
122   chname%
123   (\LWR@HTMLsanitizedetokenized{\detokenize{#1}})%
124   (\LWR@HTMLsanitizedetokenized{\detokenize{#2}})%
125   ]*?%
126   \chemformula_chwritebelow:nn {#1} {#2}%
127   \end{lateximage}%
128   }

\chlewis   Placed inline, hashed according to contents and options.

129 \VerifyCommand[lwarp][chemformula]{\chlewis}
130   {371F2DD32AA98170F43CFDA71177226B}
131
132 \RenewDocumentCommand \chlewis { O{}mm }
133   {
134   \begingroup%
135   \boolfalse{mathjax}%
136   \LWR@subsingledollar*{\textbackslash{}chlewis\{#2\}\{#3\}}%
137   {
138   \protect\LWR@HTMLsanitizedetokenized{\detokenize\expandafter{#1}}%
139   }{
140   \chemformula_lewis:nnn {#1} {#2} {#3}
141   }
142   \endgroup%
143   }

lwarp redefines the $ character, so special handling is required to escape math expressions inside \ch.

This boolean tracks a new kind of escaped math:

144 \bool_new:N   \l__chemformula_first_last_LWRdollar_bool

\chemformula_input_escape_math

Adds additional escaping for the new dollar definition:

145 \VerifyCommand[lwarp][chemformula]{__chemformula_input_escape_math:n}
146   {5318E84E9168C2F123781D2EA8CFA871}
147
148 \cs_gset_protected:Npn __chemformula_input_escape_math:n #1
149   {
150   __chemformula_first_last_math:n {#1}
151   \bool_if:NT \l__chemformula_first_last_dollar_bool
152   {
153   \bool_set_true:N \l__chemformula_first_last_math_bool
154   __chemformula_read_escape_dollar:w #1 \q_nil
155   }

```
156        \bool_if:NT \l__chemformula_first_last_mathbraces_bool
157          {
158             \bool_set_true:N \l__chemformula_first_last_math_bool
159             \__chemformula_read_escape_mathbraces:w #1 \q_nil
160          }
```

Added by lwarp:

```
161        \bool_if:NT \l__chemformula_first_last_LWRdollar_bool%      lwarp
162          {                                                         lwarp
163             \bool_set_true:N \l__chemformula_first_last_math_bool%  lwarp
164             \__chemformula_read_escape_LWRdollar:w #1 \q_nil%       lwarp
165          }
166   }
```

\chemformula_read_escape_LWRdollar

The following parses the contents inside the new dollars.

lwarp keeps the dollar as its original math shift until the document starts. While chemmacros is being patched, the dollar must temporarily be set to its new meaning during the following definition.

```
167 \begingroup
168 \catcode`\$=\active
169
170 \cs_new_protected:Npn \__chemformula_read_escape_LWRdollar:w $#1$ \q_nil
171 {
172     \__chemformula_read_escape_math:n {#1}
173 }
174
175 \endgroup
```

\chemformula_bool_set_if_first_last

The following looks at the first and last tokens for delimiters to escape math inside \ch. The original definition is modified to look for the control sequences which are used by the new meaning of $.

```
176 \cs_new_protected:Npn \__chemformula_bool_cs_set_if_first_last:NnNN #1#2#3#4
177   {
178     \int_zero:N \l__chemformula_tmpa_int
179     \int_zero:N \l__chemformula_tmpb_int
180     \int_set:Nn \l__chemformula_tmpa_int { \tl_count:n {#2} }
181     \tl_map_inline:nn {#2}
182       {
183         \int_incr:N \l__chemformula_tmpb_int
184         \int_compare:nT { \l__chemformula_tmpb_int = 1 }
185           {
```

At the start, the cs_ version compares control sequences:

```
186             \ifdefstrequal{##1}{#3}% lwarp
187                {
188                   \bool_set_true:N #1
189                }% lwarp
190                {}
191           }
```

At the end, compare more control sequences:

```
192        \int_compare:nT { \l__chemformula_tmpb_int = \l__chemformula_tmpa_int }
193          {
194            \ifdefstrequal{##1}{#4}
195              {}
196              {
197                  \bool_set_false:N #1
198              }
199          }
200      }
201  }
```

\chemformula_first_last_math

Modified to check for the new meaning of $ at first/last:

```
202 \VerifyCommand[lwarp][chemformula]{\__chemformula_first_last_math:n}
203    {B464BC6E81CAC84BE00FEE988970CE96}
204
205 \cs_gset_protected:Npn \__chemformula_first_last_math:n #1
206  {
207    \bool_set_false:N \l__chemformula_first_last_math_bool
208    \bool_set_false:N \l__chemformula_first_last_dollar_bool
209    \bool_set_false:N \l__chemformula_first_last_LWRdollar_bool%    lwarp
210    \bool_set_false:N \l__chemformula_first_last_mathbraces_bool
211    \__chemformula_bool_set_if_first_last:Nnnn
212      \l__chemformula_first_last_dollar_bool
213      {#1}
214      { $ } { $ }
215    \bool_if:NF \l__chemformula_first_last_dollar_bool
216      {
217        \__chemformula_bool_set_if_first_last:Nnnn
218          \l__chemformula_first_last_mathbraces_bool
219          {#1}
220          { \( } { \) }
```

Added by lwarp:

```
221            \bool_if:NF \l__chemformula_first_last_mathbraces_bool%    lwarp
222            {
223                \__chemformula_bool_cs_set_if_first_last:NnNN
224                \l__chemformula_first_last_LWRdollar_bool
225                {#1}
226                { \LWR@newsingledollar } { \LWR@newsingledollar }
227            }% lwarp
228      }
229  }
```

```
230 \ExplSyntaxOff
```

File 84   **lwarp-chemgreek.sty**

§ 196   Package   # chemgreek

(*Emulates or patches code by* CLEMENS NIEDERBERGER.)

chemgreek (*Pkg*)   chemgreek is patched for use by lwarp.

Greek symbols   To use text-mode symbols, use packages textalpha or textgreek. Using the other

⚠ package selection   packages supported by chemgreek will result in math-mode greek characters, which will result in svg images being used. These images will be hashed.

⚠ X_EL^AT_EX, LuaL^AT_EX   If using X_EL^AT_EX or LuaL^AT_EX, select the fontspec mapping:

> \selectchemgreekmapping{fontspec}

for HTML output: 1 \LWR@ProvidesPackagePass{chemgreek}[2020/01/16]

```
2 \ExplSyntaxOn
3
4 \cs_gset_protected:Npn \chemgreek_text:n #1
5   { { \text {#1} } }
6
7 \appto\LWR@restoreorigformatting{%
8 \cs_set_protected:Npn \chemgreek_text:n #1%
9   { \ensuremath { \text {#1} } }%
10 }
11
12 \ExplSyntaxOff
```

File 85   **lwarp-chemmacros.sty**

§ 197   Package   # **chemmacros**

(Emulates or patches code by CLEMENS NIEDERBERGER.*)*

chemmacros (*Pkg*)   chemmacros is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{chemmacros}[2022/02/13]

svg file hashing assumes that the relevent options are constant for the entire document.

§ 197.1 **Changes to the user's document**

⚠ \makepolymerdelims   When using \makepolymerdelims, enclose the entire expression inside a polymerdelims environment, such as (from the chemmacros manual):

```
\begin{polymerdelims}
\chemfig{-[@{op,.75}]CH_2-CH(-[6]Cl)-[@{cl,0.25}]]}
\makepolymerdelims{5pt}[27pt]{op}{cl}
\end{polymerdelims}
```

⚠ redox reactions   Redox reactions must be enclosed inside a redoxreaction environment. For print output, extra space must be included above and/or below the result, so they are declared as arguments to the environment, instead of being manually entered as per the chemmacros manual. For HTML output, the extra space is ignored and a lateximage is used instead.

```
\begin{redoxreaction}{7mm}{7mm}
\OX{a,Na} $\rightarrow$ \OX{b,Na}\pch\redox(a,b){oxidation}
\end{redoxreaction}
```

§ 197.2 **Code**

```
2 \ExplSyntaxOn
```

§ 197.3 **Loading packages**

Also accept the lwarp version. \VerifyCommand not used here because it doesn't work with the conditional.

```
3 \prg_set_conditional:Npnn \chemmacros_if_package_loaded:n #1 {p,T,F,TF}
4  {
5    \cs_if_exist:cTF {ver@#1.sty}
6      { \prg_return_true: }
7      {
8        \cs_if_exist:cTF {ver@lwarp-#1.sty}
9            { \prg_return_true: }
10           { \prg_return_false: }
11     }
12 }
```

Nullify hyperref detection:

```
13 \hook_gput_code:nnn {begindocument/end} {chemmacros}
14  {
15      \bool_set_false:N \l__chemmacros_hyperref_bool
16  }
```

§ 197.4 **Loading modules**

Patching chemmacros modules must be done \AtBeginDocument, since modules are invoked by the user in the preamble, and each patch is only done if the module is loaded.

§ 197.5 **New environments**

\makepolymerdelims and redox reactions must be enclosed in a lateximage during HTML output. These environments are provided here in HTML mode, and in the lwarp core in print mode, as a high-level semantic syntax which automatically embeds the contents in a lateximage with an appropriate alt tag.

Env polymerdelims

```
17 \DeclareDocumentEnvironment{polymerdelims}{}
18 {\begin{lateximage}[-chemmacros- polymer]}
19 {\end{lateximage}}
```

Env redoxreaction   {⟨*space above*⟩} {⟨*space below*⟩}

For HTML output, the above and below space is ignored, and a lateximage is used instead. For the print output version, see section 92.

```
20 \DeclareDocumentEnvironment{redoxreaction}{m m}
21 {\begin{lateximage}[-chemmacros- redoxreaction]}
22 {\end{lateximage}}
```

§ 197.6 **Acid-base**

```
23 \AtBeginDocument{
24 \chemmacros_module_if_loaded:nTF{{acid-base}}{
25 \PackageInfo{lwarp}{Patching~chemmacros~module~acid-base}
26
27 \VerifyCommand[lwarp][chemmacros]{\chemmacros_p:n}
28     {D95080E9783CB80E34C51221236CF370}
29
30 \cs_gset_protected:Npn \chemmacros_p:n #1
31  {
32     \begingroup
33     \boolfalse{mathjax}
34     \LWR@subsingledollar*{
35         \textbackslash{}%
36         p%
37         \{%
38             \LWR@HTMLsanitizedetokenized{\detokenize{#1}}%
39         \}
40     }{
41         chemmacrosp%
42         \protect\LWR@HTMLsanitizedetokenized{\detokenize\expandafter{#1}}%
43     }{
44     \group_begin:
45       \mbox
46         {
47           \chemmacros_p_style:n {p}
48           \ensuremath {#1}
49         }
50     \group_end:
51     }
52     \endgroup
53  }
54
55 \VerifyCommand[lwarp][chemmacros]{\pH}
56     {BA70E24367521CBBF282F40F65758016}
57
58 \RenewDocumentCommand \pH  {} {
59     \begingroup
60     \boolfalse{mathjax}
61     \LWR@subsingledollar*{\textbackslash{}pH}{chemmacros}{
62         \chemmacros_p:n { \chemmacros_formula:n {H} }
63     }
64     \endgroup
65 }
66
67 \VerifyCommand[lwarp][chemmacros]{\pOH}
68     {EFBC86417A28B1777F8A783D68225A3C}
69
70 \RenewDocumentCommand \pOH {} {
71     \begingroup
72     \boolfalse{mathjax}
73     \LWR@subsingledollar*{\textbackslash{}pOH}{chemmacros}{
74         \chemmacros_p:n { \chemmacros_formula:n {OH} }
75     }
76     \endgroup
77 }
78
79 \VerifyCommand[lwarp][chemmacros]{\pKa}
80     {C4141E480C360A8EDAE38B65F71F5B1F}
```

```
 81
 82 \RenewDocumentCommand \pKa {O{}}
 83   {
 84     \begingroup
 85     \boolfalse{mathjax}
 86     \LWR@subsingledollar*{\textbackslash{}pKa{[}#1{]}}{chemmacros #1}{
 87         \chemmacros_p:n
 88         {
 89             \Ka \ifblank {#1} {}
 90             { {} \c_math_subscript_token { \chemmacros_bold:n {#1} } }
 91         }
 92     }
 93     \endgroup
 94   }
 95
 96 \VerifyCommand[lwarp][chemmacros]{\pKb}
 97     {00A20E25465C2E0D2E3731634F39B0FA}
 98
 99 \RenewDocumentCommand \pKb {O{}}
100   {
101     \begingroup
102     \boolfalse{mathjax}
103     \LWR@subsingledollar*{\textbackslash{}pKb{[}#1{]}}{chemmacros #1}{
104         \chemmacros_p:n
105         {
106             \Kb \ifblank {#1} {}
107             { {} \c_math_subscript_token { \chemmacros_bold:n {#1} } }
108         }
109     }
110     \endgroup
111   }
112
113 \LetLtxMacro\LWR@chemmacros@origKa\Ka
114 \renewcommand*{\Ka}{%
115     \begingroup
116     \boolfalse{mathjax}
117     \LWR@subsingledollar*{\textbackslash{}Ka}{chemmacros}{%
118         \LWR@chemmacros@origKa%
119     }%
120     \endgroup
121 }
122
123 \LetLtxMacro\LWR@chemmacros@origKb\Kb
124 \renewcommand*{\Kb}{%
125     \begingroup
126     \boolfalse{mathjax}
127     \LWR@subsingledollar*{\textbackslash{}Kb}{chemmacros}{%
128         \LWR@chemmacros@origKb%
129     }%
130     \endgroup
131 }
132
133 \LetLtxMacro\LWR@chemmacros@origKw\Kw
134 \renewcommand*{\Kw}{%
135     \begingroup
136     \boolfalse{mathjax}
137     \LWR@subsingledollar*{\textbackslash{}Kw}{chemmacros}{
138         \LWR@chemmacros@origKw
139     }
140     \endgroup
```

```
141 }
142
143 }{}% module loaded
144 }% AtBeginDocument
```

§ 197.7 Charges

```
145 \AtBeginDocument{
146 \chemmacros_module_if_loaded:nTF{{charges}}{
147 \PackageInfo{lwarp}{Patching~chemmacros~module~charges}
148
149 \VerifyCommand[lwarp][chemmacros]{\fplus}
150    {F6F7137115BC798D4CA779782DCCEB6D}
151
152 \cs_gset_protected:Npn \fplus  {
153    \begingroup
154    \boolfalse{mathjax}
155    \LWR@subsingledollar*{\textbackslash{}fplus}{chemmacros}
156    { \LWR@origensuredmath{\chemformula_fplus:}  }
157    \endgroup
158 }
159
160 \VerifyCommand[lwarp][chemmacros]{\fminus}
161    {A7ED8520C49A794F33AA6122E2411746}
162
163 \cs_gset_protected:Npn \fminus {
164    \begingroup
165    \boolfalse{mathjax}
166    \LWR@subsingledollar*{\textbackslash{}fminus}{chemmacros}
167    { \LWR@origensuredmath{\chemformula_fminus:} }
168    \endgroup
169 }
170
171 }{}% Module loaded.
172 }% AtBeginDocument
```

§ 197.8 Nomenclature

```
173 \AtBeginDocument{
174 \chemmacros_module_if_loaded:nTF{{nomenclature}}{
175 \PackageInfo{lwarp}{Patching~chemmacros~module~nomenclature}
176
177 \VerifyCommand[lwarp][chemmacros]{\chemmacros_charge:n}
178    {258D97BF6FF3FA5C995D4FDCC44B0E63}
179
180 \cs_gset_protected:Npn \chemmacros_charge:n #1
181   {
182    \ifnumcomp{\value{LWR@laateximagedepth}}{>}{0}
183    {\chemmacros_formula:n { {}^{#1} }}
184    {
185       \ifmmode
186          {\chemmacros_formula:n { {}^{#1} }}
187       \else
188          { \textsuperscript{\ensuremath{#1}} }
189       \fi
190    }
191   }
192
```

```
193 \hook_gput_code:nnn {begindocument/end} {chemmacros}
194 {
195 \protected\def\LWR@HTML@chemprime { \HTMLunicode{2032} }
196 \LWR@formatted{chemprime}
197 }

198 \VerifyCommand[lwarp][chemmacros]{\chemmacros_cip:n}
199    {EEF7D8AF4D975C2D11D879A77ABDFF88}
200
201 \cs_gset_protected:Npn \__chemmacros_cip:n #1
202   {
203     \tl_set:Nn \l__chemmacros_tmpa_tl {#1}
204     \int_step_inline:nnnn {0} {1} {9}
205       {
206         \tl_replace_all:Nnn \l__chemmacros_tmpa_tl
207           {##1}
208           { { \l__chemmacros_cip_number_tl ##1} }
209       }
210     {
211       \l__chemmacros_cip_inner_tl
212       \LWR@textcurrentcolor{\LWR@textcurrentfont{% lwarp
213         \l__chemmacros_tmpa_tl
214       }}% lwarp
215     }
216   }

217 \VerifyCommand[lwarp][chemmacros]{\Sconf}
218    {D7A87543D1C944168CBAC59E9F45DF9A}
219
220 \RenewDocumentCommand \Sconf { O{S} } {
221 \begin{lateximage}[\textbackslash{}Sconf{[}#1{]}]*?%
222    \chemmacros_sconf:n {#1}%
223 \end{lateximage}
224 }
225
226 \VerifyCommand[lwarp][chemmacros]{\Rconf}
227    {AB922016338B2F2C2635E6104311DAF2}
228
229 \RenewDocumentCommand \Rconf { O{R} } {
230 \begin{lateximage}[\textbackslash{}Rconf{[}#1{]}]*?%
231    \chemmacros_rconf:n {#1}%
232 \end{lateximage}
233 }

234 \VerifyCommand[lwarp][chemmacros]{\chemmacros_hapto:n}
235    {FCEEDAB3292A95E65B69F4F8C8849B26}
236
237 \cs_gset_protected:Npn \chemmacros_hapto:n #1
238   {
239     \begingroup
240     \boolfalse{mathjax}
241     \LWR@subsingledollar*{\textbackslash{}hapto\{#1\}}{chemmacros}{
242       \chemmacros_coordination_symbol:nnnn
243       { \l__chemmacros_coord_use_hyphen_bool }
244       {
245         { \c_true_bool }
246       }
247       { \chemeta }
248       {#1}
249     }
```

```
250      \endgroup
251    }
252
253 \VerifyCommand[lwarp][chemmacros]{\chemmacros_dento:n}
254      {E85BBDEF5A35F37215EBAD87AFCB99E8}
255
256 \cs_gset_protected:Npn \chemmacros_dento:n #1
257    {
258      \begingroup
259      \boolfalse{mathjax}
260      \LWR@subsingledollar*{\textbackslash{}dento\{#1\}}{chemmacros}{
261          \chemmacros_coordination_symbol:nnnn
262          { \l__chemmacros_coord_use_hyphen_bool }
263          {
264              { \c_true_bool }
265          }
266          { \chemkappa }
267          {#1}
268      }
269      \endgroup
270    }
271
272 \VerifyCommand[lwarp][chemmacros]{\chemmacros_bridge:n}
273      {5E5D0EA9045A41FF30D4AB315E547B25}
274
275 \cs_gset_protected:Npn \chemmacros_bridge:n #1
276    {
277      \begingroup
278      \boolfalse{mathjax}
279      \LWR@subsingledollar*{\textbackslash{}bridge\{#1\}}{chemmacros}{
280          \chemmacros_coordination_symbol:nnnn
281          { \l__chemmacros_coord_use_hyphen_bool }
282          { \l__chemmacros_bridge_super_bool }
283          { \chemmu }
284          {#1}
285      }
286      \endgroup
287    }
288 }{}% Module loaded.
289 }% AtBeginDocument
```

§ 197.9 **Particles**

```
290 \AtBeginDocument{
291 \chemmacros_module_if_loaded:nTF{{particles}}{
292 \PackageInfo{lwarp}{Patching~chemmacros~module~particles}
293
294 \VerifyCommand[lwarp][chemmacros]{\chemmacros_declare_nucleophile:Nn}
295      {ED9AA7471C8638CEF0757A10A2E3935E}
296
297 \cs_gset_protected:Npn \chemmacros_declare_nucleophile:Nn #1#2
298    {
299      \cs_set_protected:cpn {__chemmacros_ \chemmacros_remove_backslash:N #1:}
300        {
301          \bool_if:NTF \l__chemmacros_nucleophile_elpair_bool
302            {
303              \chemmacros_elpair:n { #2 }
304                { \skip_horizontal:N \l__chemmacros_nucleophile_dim }
305              \chemmacros_formula:n { {}^{-} }
306            }
```

```
307            { \chemmacros_formula:n { #2^{-} } } }
308          }
309      \DeclareDocumentCommand #1 {o}
310        {%
311          \begin{lateximage}%
312          \group_begin:%
313            \IfNoValueF {##1}%
314              { \chemmacros_set_keys:nn {particles} {##1} }%
315            \use:c {__chemmacros_ \chemmacros_remove_backslash:N #1:}%
316          \group_end:%
317          \end{lateximage}%
318        }
319    }
320
321 \RenewChemNucleophile \Nuc {Nu}
322 \RenewChemNucleophile \ba  {ba}
323
324 }{}% Module loaded.
325 }% AtBeginDocument
```

§ 197.10 **Phases**

```
326 \AtBeginDocument{
327 \chemmacros_module_if_loaded:nTF{{phases}}{
328 \PackageInfo{lwarp}{Patching~chemmacros~module~phases}
329
330 \VerifyCommand[lwarp][chemmacros]{\chemmacros_phase:n}
331    {83788F1FCBEDA21B495E919E36DD90A5}
332
333 \cs_undefine:N \chemmacros_phase:n
334 \cs_new_protected:Npn \chemmacros_phase:n #1
335   {
336     \mode_leave_vertical:
337     \bool_if:NTF \l__chemmacros_phases_sub_bool
338       {
339         \ifnumequal{\value{LWR@lateximagedepth}}{0}
340         {
341             \textsubscript{ (#1) }
342         }
343         {
344             \chemformula_subscript:n { (#1) }
345         }
346       }
347       {
348         \skip_horizontal:N \l__chemmacros_phases_space_dim
349         \chemmacros_text:n { (#1) }
350       }
351   }
352
353 }{}% Module loaded.
354 }% AtBeginDocument
```

§ 197.11 **Mechanisms**

```
355 \AtBeginDocument{
356 \chemmacros_module_if_loaded:nTF{{mechanisms}}{
357 \PackageInfo{lwarp}{Patching~chemmacros~module~mechanisms}
358
```

```
359 \chemmacros_define_keys:nn {textmechanisms}
360   {
361     type        .choice: ,
362     type /    .code:n    =
363       {
364         \__chemmacros_set_mechanisms:nnn { S }
365           {
366               \textsubscript{N}
367           }
368         { }
369       } ,
370     type / 1  .code:n    =
371       {
372         \__chemmacros_set_mechanisms:nnn { S }
373           {
374               \textsubscript{N}
375            1
376          }
377        { }
378      } ,
379    type / 2  .code:n    =
380      {
381        \__chemmacros_set_mechanisms:nnn { S }
382          {
383              \textsubscript{N}
384           2
385         }
386       { }
387     } ,
388   type / se .code:n     =
389     {
390       \__chemmacros_set_mechanisms:nnn { S }
391         {
392             \textsubscript{E}
393         }
394       { }
395     } ,
396   type / 1e .code:n     =
397     {
398       \__chemmacros_set_mechanisms:nnn { S }
399         {
400             \textsubscript{E}
401          1
402         }
403       { }
404     } ,
405   type / 2e .code:n     =
406     {
407       \__chemmacros_set_mechanisms:nnn { S }
408         {
409             \textsubscript{E}
410          2
411         }
412       { }
413     } ,
414   type / ar .code:n     =
415     {
416       \__chemmacros_set_mechanisms:nnn { S }
417         {
418             \textsubscript{E}
```

```
419            }
420          { Ar - }
421        } ,
422      type / e   .code:n    =
423        { \__chemmacros_set_mechanisms:nnn { E } { } { } } ,
424      type / e1 .code:n    =
425        { \__chemmacros_set_mechanisms:nnn { E } { 1 } { } } ,
426      type / e2 .code:n    =
427        { \__chemmacros_set_mechanisms:nnn { E } { 2 } { } } ,
428      type / cb .code:n    =
429        {
430          \__chemmacros_set_mechanisms:nnn { E }
431            {
432              1
433                \textsubscript{cb}
434            }
435            { }
436        } ,
437      type      .default:n =
438    }
439
440 \VerifyCommand[lwarp][chemmacros]{\chemmacros_mechanisms:n}
441    {2CF049E0C61235166A36565979D79933}
442
443 \cs_gset_protected:Npn \chemmacros_mechanisms:n #1
444   {
445     \tl_if_blank:nTF {#1}
446       { \chemmacros_set_keys:nn {textmechanisms} { type } }
447       { \chemmacros_set_keys:nn {textmechanisms} { type = #1 } }
448     \mbox
449       {
450         \tl_use:N \l__chemmacros_mechanisms_ar_tl
451         \tl_use:N \l__chemmacros_mechanisms_type_tl
452         \tl_use:N \l__chemmacros_mechanisms_mol_tl
453       }
454   }
455
456 \appto\LWR@restoreorigformatting{%
457 \cs_set_protected:Npn \chemmacros_mechanisms:n #1%
458   {%
459     \tl_if_blank:nTF {#1}%
460       { \chemmacros_set_keys:nn {mechanisms} { type } }%
461       { \chemmacros_set_keys:nn {mechanisms} { type = #1 } }%
462     \mbox%
463       {%
464         \tl_use:N \l__chemmacros_mechanisms_ar_tl%
465         \tl_use:N \l__chemmacros_mechanisms_type_tl%
466         \tl_use:N \l__chemmacros_mechanisms_mol_tl%
467       }%
468   }%
469 }
470
471 }{}% Module loaded.
472 }% AtBeginDocument
```

§ 197.12 **Newman**

There are so many options that it is hard to hash these images for reuse.

```
473 \AtBeginDocument{
```

```
474 \chemmacros_module_if_loaded:nTF{{newman}}{
475 \PackageInfo{lwarp}{Patching~chemmacros~module~newman}
476
477 \VerifyCommand[lwarp][chemmacros]{\newman}
478    {45E815D161E8467A51F5B04150DEC20C}
479
480 \RenewDocumentCommand \newman {od()m}%
481   {
482     \IfValueTF{#2}
483     {\begin{lateximage}[\textbackslash{}newman(#2)\{#3\}]*?}%
484     {\begin{lateximage}[\textbackslash{}newman\{#3\}]*?}%
485     \group_begin:
486       \IfNoValueF  {#1} { \chemmacros_set_keys:nn {newman} {#1} }
487       \IfNoValueTF {#2}
488         { \chemmacros_newman:nn {   } {#3} }
489         { \chemmacros_newman:nn {#2} {#3} }
490     \group_end:
491     \end{lateximage}
492   }%
493
494 }{}% Module loaded.
495 }% AtBeginDocument
```

§ 197.13 **Orbital**

```
496 \AtBeginDocument{
497 \chemmacros_module_if_loaded:nTF{{orbital}}{
498 \PackageInfo{lwarp}{Patching~chemmacros~module~orbital}
499
500 \VerifyCommand[lwarp][chemmacros]{\orbital}
501    {F8E338F96B2EBF6AFE4A91D37A58CD90}
502
503 \RenewDocumentCommand \orbital {om}
504   {
505     \IfValueTF{#1}
506     {
507         \begin{lateximage}[%
508             \textbackslash{}%
509             orbital{[}%
510             \LWR@HTMLsanitizedetokenized{\detokenize{#1}}%
511             {]}\{#2\}%
512         ]*?[][margin-left: 1em ; margin-right: 1em]%
513     }
514     {
515         \begin{lateximage}[%
516             \textbackslash{}orbital\{#2\}%
517         ]*?[][margin-left: 1em ; margin-right: 1em]%
518     }
519     \group_begin:
520       \chemmacros_set_keys:nn {orbital/type} {#2}
521       \IfNoValueTF {#1}
522         { \chemmacros_orbital:n {   } }
523         { \chemmacros_orbital:n {#1} }
524     \group_end:
525     \end{lateximage}
526   }
527
528 }{}% Module loaded.
529 }% AtBeginDocument
```

§ 197.14 **Reactions**

\chemmacros_declare_reaction_env   {⟨*chem*⟩} {⟨*math*⟩} {⟨*args number*⟩} {⟨*argument list ({#2}{#3}…)*⟩}

```
530 \AtBeginDocument{
531 \chemmacros_module_if_loaded:nTF{{reactions}}{
532 \PackageInfo{lwarp}{Patching~chemmacros~module~reactions}
533
534 \VerifyCommand[lwarp][chemmacros]{\__chemmacros_declare_reaction_env:nnnn}
535     {E52CE623404E664FD0647E3A874F2702}
536
537 % #1: chem
538 % #2: math
539 % #3: args number
540 % #4: argument list ({#2}{#3}...)
541 \cs_gset_protected:Npn \__chemmacros_declare_reaction_env:nnnn #1#2#3#4
542   {
543     \exp_args:Nnx \DeclareDocumentEnvironment {#1}
544       { \int_compare:nT { #3+0 = 0 } {!} O{} \prg_replicate:nn {#3+0} {m} }
545       {
546         \boolfalse{mathjax}%                        lwarp
547         \ifdefvoid{\LWR@ThisAltText}{%               lwarp
548             \ThisAltText{-chemmacros-~reaction}%      lwarp
549         }{}%                                         lwarp
550         \chemmacros_add_reaction_description:n {##1}
551         \__chemmacros_begin_reaction:
552         \__chemmacros_reaction_read:nnw {#2} {#4}
553       }
554       {
555         \__chemmacros_end_reaction:
556         \gdef\LWR@ThisAltText{}%                      lwarp
557         \ignorespacesafterend
558       }
559   }
560
561 \cs_generate_variant:Nn \chemmacros_declare_reaction_env:nnnn {nnnV}
562
563 \RenewChemReaction {reaction}   {equation}
564 \RenewChemReaction {reaction*}  {equation*}
565 \RenewChemReaction {reactions}  {align}
566 \RenewChemReaction {reactions*} {align*}
567
568 }{}% Module loaded.
569 }% AtBeginDocument
```

§ 197.15 **Reactants**

Recompiled for tabular ampersand processing, with the only change being \StartDefiningTabulars. \xpatchcmd does not work here.

```
570 \VerifyCommand[lwarp][chemmacros]{\printreactants}
571     {18AD32DCD3F2F49C0369DA9ED9974CD7}
572
573 \StartDefiningTabulars%       lwarp
574
575 % #1: star: include ID in table
576 \RenewDocumentCommand \printreactants {s}
577   {
578     \group_begin:
579       \chemmacros_set_keys:nn {reactants} { switch = false }
```

```
580        \int_step_variable:nNn
581          { \seq_count:N \g_chemnum_initiated_compounds_seq }
582          \l__chemmacros_reactants_tmpa_tl
583          {
584            \seq_put_right:Nx
585              \l__chemmacros_reactants_tmpa_seq
586              {
587                \chemnum_cmpd:nnne { \c_false_bool } { \c_true_bool } {}
588                  {
589                    \seq_item:NV
590                      \g_chemnum_initiated_compounds_seq
591                      \l__chemmacros_reactants_tmpa_tl
592                  }
593                &
594                \bool_if:nT {#1}
595                  {
596                    \seq_item:NV
597                      \g_chemnum_initiated_compounds_seq
598                      \l__chemmacros_reactants_tmpa_tl
599                    &
600                  }
601                % TODO: expl3-command ??
602                \solvent
603                  {
604                    \seq_item:NV
605                      \g_chemnum_initiated_compounds_seq
606                      \l__chemmacros_reactants_tmpa_tl
607                  }
608                \tabularnewline
609              }
610            \tl_set:Nx
611              \l__chemmacros_reactants_tmpb_tl
612              {
613                \seq_item:NV
614                  \g_chemnum_initiated_compounds_seq
615                  \l__chemmacros_reactants_tmpa_tl
616              }
617            \chemmacros_reactants_list_subreactant:Vn
618              \l__chemmacros_reactants_tmpb_tl
619              {#1}
620          }
621      % TODO: longtable ?
622      %       table customizable?
623      % first draft of two styles
624      \par
625      \noindent
626      \bool_if:NTF \l__chemmacros_reactants_printreactants_style_bool
627        {
628          \str_case:Vn \l__chemmacros_reactants_printreactants_style_str
629            {
630              {xltabular}
631              {
632                \chemmacros_if_package_loaded:nTF {xltabular}
633                  {
634                    \bool_if:nTF {#1}
635                      {
636                        \begin {xltabular}
637                          { \textwidth }
638                          { @{}ll>{\raggedright\arraybackslash}X@{} }
639                      }
```

```
640                       {
641                          \begin {xltabular}
642                             { \textwidth }
643                             { @{}l>{\raggedright\arraybackslash}X@{} }
644                       }
645                    \seq_use:Nn \l__chemmacros_reactants_tmpa_seq { }
646                    \end{xltabular}
647                 }
648                 {
649                    \msg_expandable_error:nnnn
650                       {chemmacros}
651                       {package-not-loaded}
652                       { \printreactants }
653                       {xltabular}
654                 }
655              }
656              {longtable}
657              {
658                 \chemmacros_if_package_loaded:nTF {longtable}
659                    {
660                       \bool_if:nTF {#1}
661                          {
662                             \begin {longtable}[l]
663                    { @{}ll>{\raggedright\arraybackslash}p{0.6\textwidth}@{} }
664                          }
665                          {
666                             \begin {longtable}[l]
667                     { @{}l>{\raggedright\arraybackslash}p{0.9\textwidth}@{} }
668                          }
669                       \seq_use:Nn \l__chemmacros_reactants_tmpa_seq { }
670                       \end{longtable}
671                    }
672                    {
673                       \msg_expandable_error:nnnn
674                          {chemmacros}
675                          {package-not-loaded}
676                          { \printreactants }
677                          {longtable}
678                    }
679                 }
680              }
681        }
682           {
683              \msg_warning:nn {chemmacros} {missing-printreactants-style}
684           }
685     \group_end:
686  }
687
688 \VerifyCommand[lwarp][chemmacros]{\chemmacros_reactants_list_subreactant:nn}
689    {50553A53C2149BD3ADA8AE0FAB0C79C4}
690
691 % #1: full ID
692 % #2: star, include ID in table
693 \cs_gset_protected:Npn \chemmacros_reactants_list_subreactant:nn #1#2
694  {
695     \chemnum_if_subcompounds:nT {#1}
696        {
697           \int_step_variable:nNn
698              { \chemnum_count_subcompounds:n {#1} }
699              \l__chemmacros_reactants_tmpa_tl
```

```
700            {
701              \seq_put_right:Nx
702              \l__chemmacros_reactants_tmpa_seq
703                {
704                  \chemnum_cmpd:nnne { \c_false_bool } { \c_true_bool } {}
705                    {
706                      \exp_not:n {#1}
707                      \exp_not:V \l_chemnum_compound_separator_tl
708                      \chemnum_get_subcompound:nV
709                        {#1}
710                        \l__chemmacros_reactants_tmpa_tl
711                    }
712                  &
713                  \bool_if:nT {#2}
714                    {
715                      #1
716                      \l_chemnum_compound_separator_tl
717                      \chemnum_get_subcompound:nV
718                        {#1}
719                        \l__chemmacros_reactants_tmpa_tl
720                      &
721                    }
722                  % TODO: expl3-command ??
723                  \solvent
724                    {
725                      #1
726                      \l_chemnum_compound_separator_tl
727                      \chemnum_get_subcompound:nV
728                        {#1}
729                        \l__chemmacros_reactants_tmpa_tl
730                    }
731                  \tabularnewline
732                }
733            }
734        }
735    }
736 \cs_generate_variant:Nn \chemmacros_reactants_list_subreactant:nn {V}
737
738 \StopDefiningTabulars%        lwarp
```

§ 197.16 **Redox**

```
739 \AtBeginDocument{
740 \chemmacros_module_if_loaded:nTF{{redox}}{
741 \PackageInfo{lwarp}{Patching~chemmacros~module~redox}
742
743 \NewDocumentCommand \LWR@chemmacros@ox { s m >{\SplitArgument{1}{,}}m }
744 {
745    \IfBooleanTF {#1}
746      { \chemmacros_ox:nnnn {#1} {#2} #3 }
747      { \chemmacros_ox:nnnn {  } {#2} #3 }
748 }
749
750 \VerifyCommand[lwarp][chemmacros]{\ox}
751    {06B84CC6B38302F75169D5B90D8D29AA}
752
753 \RenewDocumentCommand \ox { s O{} m }
754 {
755    \begingroup
756    \boolfalse{mathjax}
```

```
757      \IfBooleanTF {#1}
758        {
759          \LWR@subsingledollar*{% yes hash
760              \textbackslash{}%
761              ox*%
762              \{%
763                  \LWR@HTMLsanitizedetokenized{\detokenize{#3}}%
764              \}% alt
765          }{%
766          star \protect\LWR@HTMLsanitizedetokenized{\detokenize\expandafter{#2}}%
767          }{%
768              \LWR@chemmacros@ox* {#2} {#3}% contents
769          }%
770        }
771        {
772          \LWR@subsingledollar*{% yes hash
773              \textbackslash{}%
774              ox%
775              \{%
776                  \LWR@HTMLsanitizedetokenized{\detokenize{#3}}%
777              \}% alt
778          }{%
779           \protect\LWR@HTMLsanitizedetokenized{\detokenize\expandafter{#2}}%
780          }{%
781              \LWR@chemmacros@ox {#2} {#3}% contents
782          }%
783        }
784    \endgroup
785  }
786
787 }{}% Module loaded.
788 }% AtBeginDocument
```

§ 197.17 Scheme

Fix for chemmacros as of v5.8b, when using newfloat and babel:

```
789 \AtBeginDocument{
790 \chemmacros_module_if_loaded:nTF{{scheme}}{
791 \PackageInfo{lwarp}{Patching~chemmacros~module~scheme}
792
793 \ifdefstring{\schemename}{los}{
794 \SetupFloatingEnvironment{scheme}{
795 name = \chemmacros_translate:n {scheme-name}
796 }
797 }{}
798
799 }{}% Module loaded.
800 }% AtBeginDocument
```

§ 197.18   **Spectroscopy**

```
801 \AtBeginDocument{
802 \chemmacros_module_if_loaded:nTF{{spectroscopy}}{
803 \PackageInfo{lwarp}{Patching~chemmacros~module~spectroscopy}
804
805 \VerifyCommand[lwarp][chemmacros]{\__chemmacros_nmr_base:nn}
806     {EDE669CC90B085080E3F96DB754836D5}
807
808 \cs_gset_protected:Npn \__chemmacros_nmr_base:nn #1#2
809   {
810     \group_begin:
811       \tl_use:N \l__chemmacros_nmr_base_format_tl
812       \tl_if_blank:VF \g__chemmacros_nmr_element_coupled_tl
813         {
814           \tl_put_left:Nn \g__chemmacros_nmr_element_coupled_tl { \{ }
815           \tl_put_right:Nn \g__chemmacros_nmr_element_coupled_tl { \} }
816         }
817       \tl_put_left:Nn \g__chemmacros_nmr_element_coupled_tl {#2}
818 %        \chemmacros_formula:n { ^{#1} }
819       \textsuperscript{#1}
820       \tl_if_blank:VF \g__chemmacros_nmr_element_coupled_tl
821         {
822           \bool_if:NTF \l__chemmacros_nmr_parse_bool
823             { \chemformula_ch:nV {} \g__chemmacros_nmr_element_coupled_tl }
824             { \chemmacros_formula:V \g__chemmacros_nmr_element_coupled_tl }
825         }
826       \tl_use:N \l__chemmacros_nmr_element_method_connector_tl
827       \tl_use:N \l__chemmacros_nmr_method_tl
828     \group_end:
829   }
830
831
832 \VerifyCommand[lwarp][chemmacros]{\chemmacros_nmr_position:n}
833     {637FDE0E801CF4052274FF60A12A38F0}
834
835 \cs_gset_protected:Npn \chemmacros_nmr_position:n #1
836   {
837     \chemmacros_formula:x
838       {
839         \exp_not:V \g__chemmacros_nmr_element_tl
840         \bool_if:NF \l__chemmacros_nmr_position_side_bool
841           {
842             \tl_if_eq:NnTF \l__chemmacros_nmr_position_tl {^}% lwarp
843             { \textsuperscript{\exp_not:n { {#1} }} }% lwarp
844             { \textsubscript{\exp_not:n { {#1} }} }% lwarp
845 %             \exp_not:V \l__chemmacros_nmr_position_tl
846 %             \exp_not:n { {#1} }
847           }
848       }
849     \bool_if:NT \l__chemmacros_nmr_position_side_bool
850       {
851         \tl_use:N \l__chemmacros_nmr_position_tl
852         \__chemmacros_nmr_position:n {#1}
853       }
854   }
855
856 \VerifyCommand[lwarp][chemmacros]{\__chemmacros_nmr_coupling:w}
857     {4D1E7321CA2F8C7EA2E4F56FB3A26EED}
858
```

```
859 \cs_gset_protected:Npn \__chemmacros_nmr_coupling:w (#1;#2)
860   {
861     \tl_set:Nn \l__chemmacros_nmr_coupling_bonds_tl
862       {
863         \l__chemmacros_nmr_coupling_bonds_pre_tl
864         #1
865         \l__chemmacros_nmr_coupling_bonds_post_tl
866       }
867     \bool_if:NTF \l__chemmacros_nmr_coupling_nuclei_sub_bool
868       {
869         \tl_set:Nn \l__chemmacros_nmr_coupling_nuclei_tl
870           {
871 %             \c_math_subscript_token
872             \textsubscript% lwarp
873               {
874                 \l__chemmacros_nmr_coupling_nuclei_pre_tl
875                 \chemmacros_formula:n {#2}
876                 \l__chemmacros_nmr_coupling_nuclei_post_tl
877               }
878           }
879       }
880       {
881         \tl_set:Nn \l__chemmacros_nmr_coupling_nuclei_tl
882           {
883             \l__chemmacros_nmr_coupling_nuclei_pre_tl
884             \chemmacros_formula:n {#2}
885             \l__chemmacros_nmr_coupling_nuclei_post_tl
886           }
887       }
888     \__chemmacros_nmr_coupling_aux_i:w
889   }
890 \AfterEndPreamble{% After \AtBeginDocument
891
892 \VerifyCommand[lwarp][chemmacros]{\chemmacros_nmr:nnnn}
893     {FD67505420F044B2CA8E7CBD05B1ECEB}
894
895 % \NMR{<num>,<elem>}(<num>,<unit>)[<solvent>] ALL arguments are optional
896 % \NMR* same but without ": $\delta$" at end
897 \cs_gset_protected:Npn \chemmacros_nmr:nnnn #1#2#3#4
898     {
899     \bool_if:NT \l__chemmacros_nmr_list_bool { \item \scan_stop: }
900     \group_begin:

901         \mode_leave_vertical:

902         \bool_set_false:N \l__chemmacros_nmr_frequency_bool
903         \bool_set_false:N \l__chemmacros_nmr_solvent_bool
904         \tl_if_empty:nF {#3}
905         { \bool_set_true:N \l__chemmacros_nmr_frequency_bool }
906         \tl_if_empty:nF {#4}
907         { \bool_set_true:N \l__chemmacros_nmr_solvent_bool }
908         \bool_if:nT
909         {
910             \l__chemmacros_nmr_frequency_bool
911             ||
912             \l__chemmacros_nmr_solvent_bool
913         }
914         { \bool_set_true:N \l__chemmacros_nmr_delimiters_bool }
915         \bool_if:nT
916         {
```

```
917                \l__chemmacros_nmr_frequency_bool
918                &&
919                \l__chemmacros_nmr_solvent_bool
920            }
921            { \bool_set_true:N \l__chemmacros_nmr_comma_bool }
922            \tl_if_empty:nTF {#2}
923            {
924                \__chemmacros_nmr_nucleus:VV
925                \l__chemmacros_nmr_isotope_default_tl
926                \l__chemmacros_nmr_element_default_tl
927            }
928            { \__chemmacros_nmr_nucleus:w #2 \q_stop }
929            \mode_if_math:TF
930            {
931                \text
932                {
933                    \group_begin:
934                    \tl_use:N \l__chemmacros_nmr_format_tl
935 \LWR@textcurrentcolor{\LWR@textcurrentfont{% lwarp
936                    \__chemmacros_nmr_base:VV
937                        \g__chemmacros_nmr_isotope_tl
938                        \g__chemmacros_nmr_element_tl
939                    \bool_if:NT \l__chemmacros_nmr_delimiters_bool
940                        { ~ ( }
941                    \bool_if:NT \l__chemmacros_nmr_frequency_bool
942                        { \__chemmacros_nmr_frequency:n {#3} }
943                    \bool_if:NT \l__chemmacros_nmr_comma_bool
944                        { , ~ }
945                    \bool_if:NT \l__chemmacros_nmr_solvent_bool
946                        { \chemmacros_formula:n {#4} }
947                    \bool_if:NT \l__chemmacros_nmr_delimiters_bool
948                        { ) }
949                    \tl_if_blank:nT {#1} {:~}
950 }}% lwarp
951                    \group_end:
952                }
953                \tl_if_blank:nT {#1}
954                {
955                    \delta
956                    \text { \l__chemmacros_nmr_delta_tl }
957                    \bool_if:NT \l__chemmacros_nmr_use_equal_bool {=}
958                }
959            }
960            {
961                \group_begin:
962                \tl_use:N \l__chemmacros_nmr_format_tl
963 \LWR@textcurrentcolor{\LWR@textcurrentfont{% lwarp
964                \__chemmacros_nmr_base:VV
965                    \g__chemmacros_nmr_isotope_tl
966                    \g__chemmacros_nmr_element_tl
967                \bool_if:NT \l__chemmacros_nmr_delimiters_bool
968                    {~()
969                \bool_if:NT \l__chemmacros_nmr_frequency_bool
970                    { \__chemmacros_nmr_frequency:n {#3} }
971                \bool_if:NT \l__chemmacros_nmr_comma_bool
972                    {,~}
973                \bool_if:NT \l__chemmacros_nmr_solvent_bool
974                    {
975                    \bool_if:NTF \l__chemmacros_nmr_parse_bool
```

```
976 %                         { \chemformula_ch:nn { } {#4} }%  original
977                         {\ch{#4}}%  lwarp
978                         {#4}
979                     }
980             \bool_if:NT \l__chemmacros_nmr_delimiters_bool
981                 {)}
982 }}% lwarp
983             \tl_if_blank:nT {#1} {:}
984             \group_end:
985             \tl_if_blank:nT {#1}
986             {
987                 \tl_use:N \c_space_tl
988                 \c_math_toggle_token
989                 \delta
990                 \c_math_toggle_token
991                 \l__chemmacros_nmr_delta_tl
992                 \bool_if:NT \l__chemmacros_nmr_use_equal_bool {~=}
993             }
994         }
995     \group_end:
996     }
997 }% AfterEndPremble
998
999
1000 \VerifyCommand[lwarp][chemmacros]{\chemmacros_data:w}
1001     {30A6134DE00E9850E074854B48644833}
1002
1003 \RenewDocumentCommand \chemmacros_data:w { smo }
1004   {
1005     \bool_if:NT \l__chemmacros_nmr_list_bool { \item }
1006       {
1007 %           \tl_use:N \l__chemmacros_nmr_format_tl #2
1008         \tl_use:N \l__chemmacros_nmr_format_tl
1009         \LWR@textcurrentcolor{\LWR@textcurrentfont{% lwarp
1010             #2
1011             \IfNoValueF {#3} { ~ ( #3 ) }
1012         \IfBooleanT {#1} { \bool_if:NT \l__chemmacros_nmr_use_equal_bool { : } }
1013         }}% lwarp
1014       }
1015     \IfBooleanF {#1} { \bool_if:NT \l__chemmacros_nmr_use_equal_bool { ~ = } }
1016   }
1017
1018 }{}% Module loaded.
1019 }% AtBeginDocument
```

§ 197.19   **Thermodynamics**

```
1020 \AtBeginDocument{
1021 \chemmacros_module_if_loaded:nTF{{thermodynamics}}{
1022 \PackageInfo{lwarp}{Patching~chemmacros~module~thermodynamics}
1023
1024 \VerifyCommand[lwarp][chemmacros]{\chemmacros_state:nnnnnn}
1025     {C5B35D9405E380ABE9A9CE849F46EE6D}
1026
1027 \cs_gset_protected:Npn \chemmacros_state:nnnnnn #1#2#3#4#5#6
1028   {
1029     \group_begin:
1030       \chemmacros_set_keys:ne {thermodynamics}
1031         {
1032           \exp_not:n {#1} ,
```

```
1033              \tl_if_novalue:nF {#2} { subscript-left = \exp_not:n {#2} , }
1034              \tl_if_novalue:nF {#3} { superscript-left = \exp_not:n {#3} , }
1035              \tl_if_novalue:nF {#5} { subscript-right = \exp_not:n {#5} , }
1036              \tl_if_novalue:nF {#6} { superscript-right = \exp_not:n {#6} }
1037          }
1038        \LWR@subsingledollar*{% yes hashing
1039            \textbackslash{}state%
1040            \{\LWR@HTMLsanitizedetokenized{\detokenize{#4}}\}% alt
1041        }{%
1042            chemmacros_state% add'l hashing
1043            #1% options
1044            LSP \tl_use:N \l__chemmacros_state_sp_left_tl% super/subscripts
1045            LSB \tl_use:N \l__chemmacros_state_sb_left_tl
1046            RSP \tl_use:N \l__chemmacros_state_sp_right_tl
1047            RSB \tl_use:N \l__chemmacros_state_sb_right_tl
1048        }
1049        {
1050            \LWR@origensuredmath
1051                {
1052                \chemmacros_text:V \l__chemmacros_state_pre_tl
1053                \c_math_superscript_token
1054                    { \chemmacros_text:V \l__chemmacros_state_sp_left_tl }
```

Only add the subscripts if they are being used. This avoids causing an incorrect depth, as the empty subscript will be measured by TEX but cropped out by *pdfcrop*.

```
1055                \tl_if_empty:NTF \l__chemmacros_state_sb_left_tl
1056                {}
1057                {
1058                    \c_math_subscript_token
1059                      { \chemmacros_text:V \l__chemmacros_state_sb_left_tl }
1060                }
1061                #4
1062                \c_math_superscript_token
1063                    { \chemmacros_text:V \l__chemmacros_state_sp_right_tl }
1064                \tl_if_empty:NTF \l__chemmacros_state_sb_right_tl
1065                {}
1066                {
1067                    \c_math_subscript_token
1068                    { \chemmacros_text:V \l__chemmacros_state_sb_right_tl }
1069                }
1070                \chemmacros_text:V \l__chemmacros_state_post_tl
1071                }
1072        }
1073    \group_end:
1074  }
1075 \cs_generate_variant:Nn \chemmacros_state:nnnnnn { nVVVVV }
1076
1077 \VerifyCommand[lwarp][chemmacros]{\chemmacros_declare_state:Nn}
1078    {3C1386935B85ED732A283627DA403FBE}
1079
1080 \cs_gset_protected:Npn \chemmacros_declare_state:Nn #1#2
1081  {
1082    \chemmacros_define_keys:xn
1083      {thermodynamics/\chemmacros_remove_backslash:N #1}
1084      {
1085       pre              .meta:nn = {chemmacros/thermodynamics} { pre = ##1 } ,
1086       post             .meta:nn = {chemmacros/thermodynamics} { post = ##1 } ,
1087      superscript-left  .meta:nn = {chemmacros/thermodynamics} { superscript-left = ##1 } ,
1088      superscript-right .meta:nn = {chemmacros/thermodynamics} { superscript-right = ##1 } ,
```

```
1089          superscript        .meta:n  = { superscript-right = ##1 } ,
1090     subscript-left    .meta:nn = {chemmacros/thermodynamics} { subscript-left = ##1 } ,
1091     subscript-right   .meta:nn = {chemmacros/thermodynamics} { subscript-right = ##1 } ,
1092         subscript       .meta:n   = { subscript-left = ##1 } ,
1093         subscript-pos   .choices:nn =
1094          { left , right }
1095         { \tl_set_eq:NN \l__chemmacros_state_sb_pos_tl \l_keys_choice_tl } ,
1096         symbol              .tl_set:N = \l__chemmacros_state_symbol_tl ,
1097         unit                .tl_set:N = \l__chemmacros_state_unit_tl
1098     }
1099   \DeclareDocumentCommand #1 { sO{}D(){}m }
1100     {
1101       \group_begin:
1102         \chemmacros_set_keys:en
1103           {thermodynamics/\chemmacros_remove_backslash:N #1}
1104           {#2}
1105         \tl_if_blank:nF {##3}
1106           {
1107             \chemmacros_set_keys:ne {thermodynamics}
1108             { subscript-\l__chemmacros_state_sb_pos_tl = \exp_not:n {##3} }
1109           }
1110 %        \LWR@origensuredmath
1111 %          {
1112           \chemmacros_state:nVVVVV
1113             {##2}
1114             \c_novalue_tl
1115             \c_novalue_tl
1116             \l__chemmacros_state_symbol_tl
1117             \c_novalue_tl
1118             \c_novalue_tl
1119         \chemmacros_set_keys_groups:nnn {thermodynamics} {variables} {##2}
1120         \IfBooleanF {##1} { = \qty {##4} { \l__chemmacros_state_unit_tl } }
1121 %          }
1122       \group_end:
1123     }
1124   }
```

The pre-existing macros are redefined with the new definition:

```
1125 \RenewChemState \enthalpy { symbol = H , unit = \kilo\joule\per\mole }
1126 \RenewChemState \entropy  { symbol = S , unit = \joule\per\kelvin\per\mole , pre = }
1127 \RenewChemState \gibbs    { symbol = G , unit = \kilo\joule\per\mole }
1128
1129 }{}% Module loaded.
1130 }% AtBeginDocument
```

```
1131 \ExplSyntaxOff
```

File 86 **`lwarp-chemnum.sty`**

§ 198   Package   # chemnum

(*Emulates or patches code by* CLEMENS NIEDERBERGER.)

chemnum (*Pkg*)   chemnum is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{chemnum}[2016/04/14]

```
2 \ExplSyntaxOn
3
4 \VerifyCommand[lwarp][chemnum]{\chemnum_compound_write:n}
5   {E47ACDCCC4D90FAC40B75B53721EC218}
6
7 \cs_gset_protected:Npn \chemnum_compound_write:n #1
8   {
9     \chemnum_get_compound_property:nn {#1} {pre-main-label-code}
10    \group_begin:
11      \bool_if:NTF \l__chemnum_compound_local_bool
12        { \l__chemnum_local_label_format_tl }
13        { \chemnum_get_compound_property:nn {#1} {label-format} }
14      {
15        \LWR@textcurrentfont{
16            \chemnum_get_compound_property:nn {#1} {counter-representation}
17        }
18      }
19    \group_end:
20    \chemnum_get_compound_property:nn {#1} {post-main-label-code}
21  }
22
23 \VerifyCommand[lwarp][chemnum]{\chemnum_subcompound_write:nn}
24   {F6BB883B91A1FA330EF3B89924BF3679}
25
26 \cs_gset_protected:Npn \chemnum_subcompound_write:nn #1#2
27   {
28     \group_begin:
29       \bool_if:NTF \l__chemnum_compound_local_bool
30         { \l__chemnum_local_label_format_tl }
31         { \chemnum_get_compound_property:nn {#1} {label-format} }
32       {
33         \LWR@textcurrentfont{
34             \chemnum_get_subcompound_property:nnn {#1} {#2}
35             {counter-representation}
36         }
37       }
38     \group_end:
39   }
40
41 \ExplSyntaxOff
```

File 87   **lwarp-chkfloat.sty**

§ 199   Package   # chkfloat

chkfloat (*Pkg*)   chkfloat is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{chkfloat}[2012/08/19]`

File 88   **lwarp-chngpage.sty**

§ 200   Package   # chngpage

(*Emulates or patches code by* PETER WILSON.)

chngpage (*Pkg*)   chngpage is ignored.

for HTML output: Discard all options for lwarp-chngpage:

```
1 \LWR@ProvidesPackageDrop{chngpage}[2009/10/20]
2 \LWR@origRequirePackage{lwarp-changepage}
```

File 89   `lwarp-cite.sty`

§ 201   Package   # cite

(*Emulates or patches code by* Donald Arseneau.)

cite (*Pkg*)   cite is patched for use by lwarp.

for HTML output: `1 \LWR@ProvidesPackagePass{cite}[2015/02/27]`

For the [super] option, the \kern must be removed:

```
2 \def\LWRCT@biblabel#1{\@citess{#1}\kern-\labelsep\,}
3
4 \ifdefstrequal{\@biblabel}{\LWRCT@biblabel}
5 {
6     \def\@biblabel#1{\@citess{#1}}
7 }{}
```

For the [super] option, \textsuperscript is used instead of math superscript:

```
8 \def\@citess#1{\textsuperscript{#1}}
9
10 \DeclareDocumentCommand\citepunct{}{,\,\relax}
```

File 90   `lwarp-citeref.sty`

§ 202   Package   # citeref

(*Emulates or patches code by* Björn Briel.)

citeref (*Pkg*)   citeref is patched for use by lwarp.

for HTML output: `1 \LWR@ProvidesPackagePass{citeref}[1999/27/05]`

```
2 \def\@cprwrite#1={%
3     \write\@auxout{\string\citepageref{#1}{\theLWR@previousautopagelabel}}%
4 }
5
6 \VerifyCommand[lwarp][citeref]{\citepageref}
7     {F5E07AE6603C65E9398417D6D392825D}
8
9 \def\citepageref#1#2{%
10     \xdef\cpr@testa{\@nameuse{cpr@last@#1}}%letzte Zitatstelle
11     \xdef\cpr@testb{#2}% Seite dieser Zitatstelle
12     \ifx\cpr@testa\cpr@testb%
13         \relax% Konsekutive identische Seitenangaben weglassen
```

```
14    \else%
15        \@namexdef{cpr@last@#1}{#2}%
16        \@ifundefined{cpr@#1}%
17        {\@namexdef{cpr@#1}{\LWR@refwithsection{\BaseJobname-autopage-#2}}}%  lwarp
18            {%        lwarp
19                \@namexdef{cpr@#1}{\@nameuse{cpr@#1}, % space
20                \LWR@refwithsection{\BaseJobname-autopage-#2}}%
21            }%
22    \fi
23    }
```

File 91   `lwarp-CJK.sty`

§ 203   Package   # CJK

CJK (*Pkg*)   CJK does not work with lwarp unless called from ctex.

for HTML output:
```
1 \IfPackageLoadedTF{xeCJK}{}{
2     \LWR@loadnever{CJK}{ctex, xeCJK}
3 }
4
5 \LWR@ProvidesPackagePass{CJK}[2015/04/18]
```

File 92   `lwarp-CJKutf8.sty`

§ 204   Package   # CJKutf8

CJKutf8 (*Pkg*)   CJKutf8 does not work with lwarp unless called from ctex.

for HTML output:
```
1 \IfPackageLoadedTF{xeCJK}{}{
2     \LWR@loadnever{CJKutf8}{ctex, xeCJK}
3 }
4
5 \LWR@ProvidesPackagePass{CJKutf8}[2015/04/18]
```

File 93   `lwarp-classicthesis.sty`

§ 205   Package   # classicthesis

(*Emulates or patches code by* ANDRÉ MIEDE AND IVO PLETIKOSIĆ.)

classicthesis (*Pkg*)   classicthesis is emulated.

for HTML output: Discard all options for lwarp-classicthesis:

```
1 \LWR@ProvidesPackageDrop{classicthesis}[2018/06/03]


2 \RequirePackage{scrlayer-scrpage} % provides headers and footers (KOMA Script)
3 \RequirePackage{scrtime} % time access
4 \PassOptionsToPackage{titles}{tocloft}
5 \RequirePackage{textcase} % for \MakeTextUppercase
6 \RequirePackage[newparttoc]{titlesec} % newparttoc to write \part to .toc with \numberline
```

```
7 \RequirePackage{tocloft}
8 \PassOptionsToPackage{headinclude,footinclude}{typearea} % for classes other than KOMA
9 \RequirePackage{typearea}
10 \PassOptionsToPackage{marginal}{footmisc}% marginal flushmargin
11 \RequirePackage{footmisc}%
12 \RequirePackage{prelim2e}
13 \RequirePackage{remreset}%
14
15 \DeclareRobustCommand{\spacedallcaps}[1]{\textsc{\MakeTextUppercase{#1}}}
16 \DeclareRobustCommand{\spacedlowsmallcaps}[1]{\textsc{\MakeTextLowercase{#1}}}
17 \newcommand{\ctparttext}[1]{}
18 \newcommand{\tocEntry}[1]{}
19 \DeclareRobustCommand*{\deactivateaddvspace}{}%
20 \newlength{\beforebibskip}
```

File 94   **`lwarp-cleveref.sty`**

§ 206   Package   # cleveref

(*Emulates or patches code by* TOBY CUBITT.)

cleveref (*Pkg*)   cleveref is patched for HTML, and limited MATHJAX emulation is added.

⚠ cleveref page numbers   cleveref and varioref are supported, but printed page numbers do not map to HTML, so a section name or a text phrase are used for \cpageref and \cpagerefrange. This phrase includes \cpagerefFor, which defaults to "for".

Ex:

    \cpageref{tab:first,tab:second}
  in html becomes:
    "pages **for** table 4.1 and **for** table 4.2"

See \cpagerefFor at page 753 to redefine the message which is printed for page number references.

Table 16 on page 505 shows the data structure of the label/reference system as revised by lwarp and cleveref.

For MATHJAX, each references is printed as an \eqref, without cleveref's description text. Page references are also printed as simple \eqrefs. Multiple labels in a

⚠ multiple labels   single \cref will print as (???) in MATHJAX.

for HTML output: 1 \LWR@ProvidesPackagePass{cleveref}[2018/03/27]

The following patches are applied. Print-mode versions are not required since they all come down to \ref eventually, and \ref has a print-mode version.

\@@@setcref {⟨*kindofref*⟩} {⟨*label*⟩}

For HTML, \@templabel is replaced by the section number.

```
2 \def\LWR@orig@@@setcref#1#2{\cref@getlabel{#2}{\@templabel}#1{\@templabel}{}{}}%
3
4 \ifdefequal{\@@setcref}{\LWR@orig@@@setcref}{% before v0.21
5     \renewcommand*{\@@setcref}[2]{#1{\ref{#2}}{}{}}
```

```
 6 }{
 7     \ifdefequal{\@@@setcref}{\LWR@orig@@@setcref}{% as of v0.21
 8         \renewcommand*{\@@@setcref}[2]{%
 9             #1{\ref{#2}}{}{}}
10     }{
11         \PackageWarningNoLine{lwarp-cleveref}{
12             Unknown version of cleveref.
13             \protect\cref\space will fail.
14         }%
15     }
16 }
```

\@@@setcrefrange {⟨*text*⟩} {⟨*label*⟩} {⟨*label*⟩}

```
17 \def\LWR@orig@@@setcrefrange#1#2#3{%
18   \cref@getlabel{#2}{\@labela}%
19   \cref@getlabel{#3}{\@labelb}%
20   #1{\@labela}{\@labelb}{}{}{}{}%
21
22 \ifdefequal{\@@setcrefrange}{\LWR@orig@@@setcrefrange}{
23     \renewcommand{\@@setcrefrange}[3]{%
24         #1{\ref{#2}}{\ref{#3}}{}{}{}{}%
25     }
26 }{
27     \ifdefequal{\@@@setcrefrange}{\LWR@orig@@@setcrefrange}{
28         \renewcommand{\@@@setcrefrange}[3]{%
29             #1{\ref{#2}}{\ref{#3}}{}{}{}{}%
30         }
31     }{
32         \PackageWarningNoLine{lwarp-cleveref}{
33             Unknown version of cleveref.
34             \protect\crefrange\space will fail.
35         }
36     }
37 }
```

\cpagerefFor   Redefinable word between "page(s)" and the page numbers.

```
38 \newcommand*{\cpagerefFor}{for}
```

\@@@setcpageref {⟨*typeofref*⟩} {⟨*label*⟩}, where typeofref is "page" or "pages"

```
39 \def\LWR@orig@@setcpageref#1#2{% before v0.21
40   \cref@getpageref{#2}{\@temppage}#1{\@temppage}{}{}}%
41
42 \def\LWR@orig@@@setcpageref#1#2{% as of v0.21
43   \cpageref@getlabel{#2}{\@temppage}#1{\@temppage}{}{}}%
44
45 \ifdefequal{\@@setcpageref}{\LWR@orig@@setcpageref}{
46     \renewcommand*{\@@setcpageref}[2]{%
47         #1{\cpagerefFor\ \cref{#2}}{}{}%
48     }
49 }{
50     \ifdefequal{\@@@setcpageref}{\LWR@orig@@@setcpageref}{
51         \renewcommand*{\@@@setcpageref}[2]{%
52             #1{\cpagerefFor\ \cref{#2}}{}{}%
53         }
```

```
54       }
55       {
56           \PackageWarningNoLine{lwarp-cleveref}{
57               Unknown version of cleveref.
58               \protect\cpageref\space will fail.
59           }
60       }
61 }
```

```
62 \def\LWR@orig@@setcpagerefrange#1#2#3{% before v0.21
63   \cref@getpageref{#2}{\@pagea}%
64   \cref@getpageref{#3}{\@pageb}%
65   #1{\@pagea}{\@pageb}{}{}{}{}%
66
67 \def\LWR@orig@@@setcpagerefrange#1#2#3{% as of v0.21
68   \cpageref@getlabel{#2}{\@pagea}%
69   \cpageref@getlabel{#3}{\@pageb}%
70   #1{\@pagea}{\@pageb}{}{}{}{}%
71
72 \ifdefequal{\@@setcpagerefrange}{\LWR@orig@@setcpagerefrange}{
73     \renewcommand*{\@@setcpagerefrange}[3]{%
74         #1{\cpagerefFor\ \cref{#2}}{\cref{#3}}{}{}{}{}%
75     }
76 }{
77     \ifdefequal{\@@@setcpagerefrange}{\LWR@orig@@@setcpagerefrange}{
78         \renewcommand*{\@@@setcpagerefrange}[3]{%
79             #1{\cpagerefFor\ \cref{#2}}{\cref{#3}}{}{}{}{}%
80         }
81     }
82     {
83         \PackageWarningNoLine{lwarp-cleveref}{
84             Unknown version of cleveref.
85             \protect\cpagerefrange\space will fail.
86         }
87     }
88 }
```

If hyperref is loaded, cleveref defines starred versions of the following, but since hyperref is only emulated, starred versions are defined here:

```
89 \LWR@absorbstar{cref}
90 \LWR@absorbstar{Cref}
91 \LWR@absorbstar{crefrange}
92 \LWR@absorbstar{Crefrange}
93 \LWR@absorbstar{cpageref}
94 \LWR@absorbstar{Cpageref}
95 \LWR@absorbstar{cpagerefrange}
96 \LWR@absorbstar{Cpagerefrange}
97 \LWR@absorbstar{labelcref}
98 \LWR@absorbstar{labelcpageref}
```

If hyperref is loaded, cleveref also defines starred versions of varioref macros, so they are defined here.

```
99 \IfPackageLoadedTF{varioref}{
100    \LWR@absorbstar{vref}
101    \LWR@absorbstar{Vref}
102    \LWR@absorbstar{vrefrange}
103    \LWR@absorbstar{Vrefrange}
```

```
104     \LWR@absorbstar{fullref}
105     \LWR@absorbstar{Fullref}
106 }{}% varioref
```

```
107 \IfClassLoadedTF{memoir}{
108 \AtBeginDocument{
109 \def\sf@memsub@label(#1)#2{%
110   \protected@edef\mem@currentlabelname{#1}%
111   \sf@@memsub@label{#2}}
112 }
113 }{}
```

```
114 \IfPackageLoadedTF{subfig}{
115 \def\sf@sub@label(#1)#2{%
116   \ifhyperrefloaded
117     \protected@edef\@currentlabelname{%
118       \expandafter\strip@period #1\relax.\relax\@@@}%
119   \fi
120   \sf@@sub@label{#2}}
121 }{}
```

File 95   **lwarp-clrdblpg.sty**

§ 207   Package   **clrdblpg**

clrdblpg (*Pkg*)   clrdblpg is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{clrdblpg}[2018/04/21]

File 96   **lwarp-cmbright.sty**

§ 208   Package   **cmbright**

(*Emulates or patches code by* Walter Schmidt.)

cmbright (*Pkg*)   cmbright is used as-is for svg math, and is emulated for MathJax.

⚠ limitations   The MathJax emulation ignores all package options, except slantedGreek is honored, and \mathbold is available.

The dedicated macros for upright Greek letters do work correctly.

svg math should appear the same as the printed output.

for HTML output: 1 \LWR@ProvidesPackagePass{cmbright}[2005/04/13]
                        2
                        3 \LWR@infoprocessingmathjax{cmbright}

                        4 \LWR@origRequirePackage{lwarp-common-mathjax-letters}
                        5
                        6 \begin{warpMathJax}
                        7
                        8 \IfPackageLoadedWithOptionsTF{cmbright}{slantedGreek}

```
 9 {
10     \LWR@mathjax@addgreek@u@it*{}{}
11 }
12 {}
13
14 \LWR@mathjax@addgreek@u@up*{up}{}
15
16 \CustomizeMathJax{\newcommand{\mathbold}[1]{\boldsymbol{#1}}}
17
18 \end{warpMathJax}
```

File 97   **lwarp-cmdtrack.sty**

§ 209   Package   # cmdtrack

cmdtrack (*Pkg*)   cmdtrack is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{cmdtrack}[2012/12/18]

2 \newcommand{\untrack}[1]{}

File 98   **lwarp-colonequals.sty**

§ 210   Package   # colonequals

(*Emulates or patches code by* HEIKO OBERDIEK.)

colonequals (*Pkg*)   colonequals is used as-is for SVG math, and is emulated for MATHJAX.

Since UNICODE symbols are not available for each of the following, only two are used for the single and double colons, and the other symbols are derived in a consistent manner. Occasional negative space is added as well. This may need to be undone for some fonts.

for HTML output: 1 \LWR@ProvidesPackagePass{colonequals}[2016/05/16]

```
 2 \begin{warpMathJax}
 3 \LWR@infoprocessingmathjax{colonequals}
 4
 5 \CustomizeMathJax{\newcommand{\ratio}{\mathrel{\unicode{x2236}}}}
 6 \CustomizeMathJax{\newcommand{\coloncolon}{\mathrel{\unicode{x2237}}}}
 7 \CustomizeMathJax{\newcommand{\colonequals}{\mathrel{\unicode{x2236}\!=}}}
 8 \CustomizeMathJax{\newcommand{\coloncolonequals}{\mathrel{\unicode{x2237}\!=}}}
 9 \CustomizeMathJax{\newcommand{\equalscolon}{\mathrel{=\!\unicode{x2236}}}}
10 \CustomizeMathJax{\newcommand{\equalscoloncolon}{\mathrel{=\!\unicode{x2237}}}}
11 \CustomizeMathJax{\newcommand{\colonminus}{\mathrel{\unicode{x2236}-}}}
12 \CustomizeMathJax{\newcommand{\coloncolonminus}{\mathrel{\unicode{x2237}-}}}
13 \CustomizeMathJax{\newcommand{\minuscolon}{\mathrel{-\unicode{x2236}}}}
14 \CustomizeMathJax{\newcommand{\minuscoloncolon}{\mathrel{-\unicode{x2237}}}}
15 \CustomizeMathJax{\newcommand{\colonapprox}{\mathrel{\unicode{x2236}\!\approx}}}
16 \CustomizeMathJax{\newcommand{\coloncolonapprox}{\mathrel{\unicode{x2237}\!\approx}}}
17 \CustomizeMathJax{\newcommand{\approxcolon}{\mathrel{\approx\!\unicode{x2236}}}}
18 \CustomizeMathJax{\newcommand{\approxcoloncolon}{\mathrel{\approx\!\unicode{x2237}}}}
19 \CustomizeMathJax{\newcommand{\colonsim}{\mathrel{\unicode{x2236}\!\sim}}}
```

```
20 \CustomizeMathJax{\newcommand{\coloncolonsim}{\mathrel{\unicode{x2237}\!\sim}}}
21 \CustomizeMathJax{\newcommand{\simcolon}{\mathrel{\sim\!\unicode{x2236}}}}
22 \CustomizeMathJax{\newcommand{\simcoloncolon}{\mathrel{\sim\!\unicode{x2237}}}}
23 \end{warpMathJax}
```

File 99 **`lwarp-color.sty`**

§ 211   Package **color**

color (*Pkg*)   Allowed but ignored. xcolor is then required as well.

color is superceded by xcolor, and lwarp requires several of the features of xcolor. When color is requested, xcolor is loaded as well.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{color}[2016/07/10]
2 \RequirePackage{xcolor}
```

\color@endgroup's \endgraf was conflicting with lwarp's paragraph handling.

```
3 \let\color@endgroup\endgroup
```

File 100   **`lwarp-colortbl.sty`**

§ 212   Package **colortbl**

colortbl (*Pkg*)   colortbl is used as-is for print output, and emulated for HTML.

⚠ row/cell color   Only use \rowcolor and \cellcolor at the start of a row, in that order.

colortbl ignores the overhang arguments.

colored tables   \rowcolors is supported, except that the optional argument is ignored so far.

for HTML output: A placeholder definition is forgotten first:

```
1 \let\rowcolor\relax
2
3 \LWR@ProvidesPackagePass{colortbl}[2022/06/20]
```

The following \LWR@HTML versions are used inside an HTML tabular.

\columncolor   [⟨*model*⟩] {⟨*color*⟩} [⟨*left overhang*⟩] [⟨*right overhang*⟩]

\LWR@getmynexttoken is not used here because \columncolor is not used inside the data area of the tabular.

\columncolor is provided here to satisfy \LWR@formatted's test for the existence of the print-mode macro.

```
4 \ProvideDocumentCommand{\columncolor}{O{named} m o o}{}%
5
6 \NewDocumentCommand{\LWR@HTML@columncolor}{O{named} m o o}{%
7     \convertcolorspec{#1}{#2}{HTML}\LWR@columnHTMLcolor%
8     \LWR@addtabularcellcolor%
9 }
```

```
10
11 \AtBeginDocument{\LWR@formatted{columncolor}}
```

\LWR@getmynexttoken is used for \rowcolor because it is used inside the data area of the tabular.

\rowcolor   [⟨*model*⟩] {⟨*color*⟩} [⟨*left overhang*⟩] [⟨*right overhang*⟩]

```
12 \NewDocumentCommand{\LWR@HTML@rowcolor}{O{named} m o o}{%
13     \convertcolorspec{#1}{#2}{HTML}\LWR@rowHTMLcolor%
14     \LWR@getmynexttoken%
15 }
16
17 \AtBeginDocument{\LWR@expandableformatted{rowcolor}}
```

\cellcolor   [⟨*model*⟩] {⟨*color*⟩} [⟨*left overhang*⟩] [⟨*right overhang*⟩]

```
18 \NewDocumentCommand{\LWR@HTML@cellcolor}{O{named} m o o}{%
19     \convertcolorspec{#1}{#2}{HTML}\LWR@cellHTMLcolor%
20     \LWR@addtabularcellcolor%
21 }
22
23 \AtBeginDocument{\LWR@formatted{cellcolor}}
```

\arrayrulecolor   [⟨*model*⟩] {⟨*color*⟩}

The HTML version for use outside a tabular. Inside a tabular, \LWR@HTML@arrayrulecolornexttoken is used instead.

```
24 \newcommand{\LWR@HTML@arrayrulecolor}[2][named]{%
25     \convertcolorspec{#1}{#2}{HTML}\LWR@ruleHTMLcolor%
26 }
27
28 \AtBeginDocument{\LWR@expandableformatted{arrayrulecolor}}
```

     [⟨*model*⟩] {⟨*color*⟩}

\LWR@arrayrulecolornexttoken   The HTML version for use inside a tabular.

```
29 \newcommand{\LWR@HTML@arrayrulecolornexttoken}[2][named]{%
30     \convertcolorspec{#1}{#2}{HTML}\LWR@ruleHTMLcolor%
31     \LWR@getmynexttoken%
32 }
33
34 \AtBeginDocument{\LWR@expandableformatted{arrayrulecolornexttoken}}
```

\doublerulesepcolor   [⟨*model*⟩] {⟨*color*⟩}

The version for use outside a tabular.

```
35 \newcommand{\LWR@HTML@doublerulesepcolor}[2][named]{}
36
37 \AtBeginDocument{\LWR@expandableformatted{doublerulesepcolor}}
```

     [⟨*model*⟩] {⟨*color*⟩}

\LWR@doublerulesepcolornexttoken The version for use inside a tabular.

```
38 \newcommand{\LWR@HTML@doublerulesepcolornexttoken}[2][named]{\LWR@getmynexttoken}
39
```

```
40 \AtBeginDocument{\LWR@expandableformatted{doublerulesepcolornexttoken}}
```

\rowc@l@rs [⟨*cmds*⟩] {⟨*startrow*⟩} {⟨*odd color*⟩} {⟨*even color*⟩}

```
41 \newcommand*{\LWR@xcolortempcolor}{}
42
43 \VerifyCommand[lwarp][colortbl]{\rowc@l@rs}{A66C3974E0C5BD5C3DDE033367D197A4}
44
45 \def\rowc@l@rs[#1]#2#3#4%
46 {%
```

The lwarp emulation starts at row 1 instead of 0.

```
47 %   \global\rownum=\z@
48    \global\rownum=1%          lwarp
49    \global\@rowcolorstrue%
50    \@ifxempty{#3}%
51     {\def\@oddrowcolor{\@norowcolor}}%
52     {%
53        \convertcolorspec{named}{#3}{HTML}\LWR@xcolortempcolor%     lwarp
54        \edef\@oddrowcolor{%
55           \csdef{LWR@xcolorrowHTMLcolor}{\LWR@xcolortempcolor}%   lwarp
56        }%
57     }%
58    \@ifxempty{#4}%
59     {\def\@evenrowcolor{\@norowcolor}}%
60     {%
61        \convertcolorspec{named}{#4}{HTML}\LWR@xcolortempcolor%     lwarp
62        \edef\@evenrowcolor{%
63           \csdef{LWR@xcolorrowHTMLcolor}{\LWR@xcolortempcolor}%   lwarp
64        }%
65     }%
66    \if@rowcmd
67     \def\@rowcolors
68     {%
69 %        #1%
70        \if@rowcolors
71 %        \noalign{%
72           \relax\ifnum\rownum<#2\@norowcolor\else
73           \ifodd\rownum\@oddrowcolor\else\@evenrowcolor\fi\fi%
74 %        }%
75        \fi%
76     }%
77    \else
78     \def\@rowcolors
79     {%
80        \if@rowcolors
81           \ifnum\rownum<#2%
82 %           \noalign{%
83              \@norowcolor
84 %           }
85           \else
86 %           #1%
87 %           \noalign{%
88              \ifodd\rownum\@oddrowcolor\else\@evenrowcolor\fi%
89 %           }%
90           \fi
91        \fi%
92     }%
93    \fi
```

```
94     \ignorespaces%
95 }
```

\@norowcolor   Turns off color for this row.

```
96 \def\@norowcolor{%
97     \renewcommand{\LWR@xcolorrowHTMLcolor}{}%
98 }
```

\@rowc@lors   Executed at the end of each row.

```
 99 \def\@rowc@lors{%
100 %     \noalign{%
101         \advance\rownum\@ne%
102 %     }%
103     \@rowcolors%
104 }
```

For MATHJAX, use the MATHJAX package. The unused macro options are ignored.

```
105 \begin{warpMathJax}
106
107 \CustomizeMathJax{\require{colortbl}}
108 \CustomizeMathJax{\let\LWRorigcolumncolor\columncolor}
109 \CustomizeMathJax{\renewcommand{\columncolor}[2][named]{%
110     \LWRorigcolumncolor[#1]{#2}%
111     \LWRabsorbtwooptions%
112 }}
113
114 \CustomizeMathJax{\let\LWRorigrowcolor\rowcolor}
115 \CustomizeMathJax{\renewcommand{\rowcolor}[2][named]{%
116     \LWRorigrowcolor[#1]{#2}%
117     \LWRabsorbtwooptions%
118 }}
119
120 \CustomizeMathJax{\let\LWRorigcellcolor\cellcolor}
121 \CustomizeMathJax{\renewcommand{\cellcolor}[2][named]{%
122     \LWRorigcellcolor[#1]{#2}%
123     \LWRabsorbtwooptions%
124 }}
125
126 \end{warpMathJax}
```

File 101   **lwarp-continue.sty**

§ 213   Package   # continue

continue (*Pkg*)   continue is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{continue}}[2018/12/09]`

```
2 \newcommand*{\flagcont}{}
3 \newcommand*{\flagend}{}
4 \newcommand*{\flagword}{}
5 \newcommand*{\preflagword}{}
6 \newcommand*{\postflagword}{}
7 \newlength\contsep
```

8 \newlength\contdrop

File 102   **lwarp-copyrightbox.sty**

§ 214   Package   # copyrightbox

(*Emulates or patches code by* Thomas Fischer, Ives van der Flaas.)

copyrightbox (*Pkg*)   copyrightbox is emulated for use by lwarp.

The entire copyright box is placed inside a <div> of class copyrightbox.

The contents are placed inside a <div> of class copyrightboxcontents.

The copyright notice is placed inside a <div> of class copyrightboxnote.

for HTML output: 1 \LWR@ProvidesPackageDrop{copyrightbox}[2011/11/27]

```
2 \newcommand{\copyrightbox}[3][r]{%
3 \begin{BlockClass}[
4     display: inline-flex;
5     flex-direction: column ;
6 ]{copyrightbox}
7 \begin{BlockClass}{copyrightboxcontents}
8 #2
9 \end{BlockClass}
10 \begin{BlockClass}{copyrightboxnote}
11 #3
12 \end{BlockClass}
13 \end{BlockClass}
14 }
15
16 \newcommand{\CRB@setcopyrightfont}{}
17 \newcommand{\CRB@setcopyrightparagraphstyle}{}
```

File 103   **lwarp-crop.sty**

§ 215   Package   # crop

(*Emulates or patches code by* Melchior Franz.)

crop (*Pkg*)   crop is ignored.

for HTML output: Discard all options for lwarp-crop:

1 \LWR@ProvidesPackageDrop{crop}[2003/05/20]

```
2 \newcommand*{\crop}[1][]{}
3 \newcommand*{\cropdef}[6][]{}
```

File 104 **`lwarp-ctable.sty`**

§ 216 Package **ctable**

(*Emulates or patches code by* WYBO DEKKER.)

ctable (*Pkg*) ctable is patched for use by lwarp.

⚠ Misplaced alignment tab character & Use \StartDefiningTabulars before one or more \ctables, and \StopDefiningTabulars after. These change the meaning of the ampersand & character.

for HTML output: 1 \LWR@ProvidesPackagePass{ctable}[2015/10/17]

The following is in the original:

```
2 \newcommand{\LWR@HTML@ctable}[4][]{%
3   \let\@CTtaborfig  \@dfltCTtaborfig
4   \let\@CTalign     \@dfltCTalign
5   \let\@CTsideways  \@dfltCTsideways
6   \let\@CTcontinued \empty
7   \let\@CTpos       \@dfltCTpos
8   \let\@CTcaption   \empty
9   \let\@CTcap       \undefined
10  \let\@CTlabel     \empty
11  \let\@CTbotcap    \@dfltCTbotcap
12  \let\@CTstarred   \@dfltCTstarred
13  \let\@CTsuper     \@dfltCTsuper
14  \let\@CTnotespar  \@dfltCTnotespar
15  \let\@CTdoinside  \@dfltCTdoinside
16  \let\@CTbgopacity \@dfltCTbgopacity
17  \@CTframerule     \@dfltCTframerule
18  \@CTcaptionskip   \@dfltCTcaptionskip
19  \@CTframesep      \@dfltCTframesep
20  \@CTwidth         \@dfltCTwidth
21  \@CTmaxwidth      \@dfltCTmaxwidth
22  \@CTmincapwidth   \@dfltCTmincapwidth
23  \@CTfooterwidth   \@dfltCTfooterwidth
24  \def\@CTfgactual  {@dfltCTframefg}%
25  \def\@CTbgactual  {@dfltCTframebg}%
26  \def\@CTbeg       {\begin{\@CTsideways\@CTtaborfig\@CTstarred}}%
27  \def\@CTbegin     {\@CTbeg}%
28  \def\@CTend       {\end{\@CTsideways\@CTtaborfig\@CTstarred}}%
29  \setkeys{CT}{#1}%
30  \ifx\@CTcap\undefined\let\@CTcap\@CTcaption\fi
31  \ifx\@CTcap\empty
32    \if@CTcaptionloaded\else
33      \PackageWarningNoLine{lwarp-ctable}{\MessageBreak
34        An empty cap= option prevents lot/loc entry only\MessageBreak
35        if the caption package is loaded!}
36    \fi
37  \fi
38  \if@CTinmemoir\else
39    \ifx\@CTbotcap\undefined
40      \PackageError{lwarp-ctable}{\MessageBreak
41        You can, currently, use the sidecap option only with\MessageBreak
42        memoir documents. Use topcap or botcap only}
```

```
43              {}
44          \fi
45      \fi
46      \ifdim\@CTwidth=0pt\else
47          \ifdim\@CTmaxwidth=0pt\else
48              \PackageError{lwarp-ctable}{\MessageBreak
49              You may not use the width and maxwidth options together\MessageBreak
50                  Use either width or maxwidth}
51                  {}
52          \fi
53      \fi
54      \ifx\@CTpos\empty
55          \ifx\@CTsideways\empty\else
56          \PackageError{lwarp-ctable}{\MessageBreak
57              You may not use the pos and sideways options together\MessageBreak
58              Rotated tables and figures are always typeset on a separate page}
59              {}
60          \fi
61      \fi
62      \ifx\@CTcaption\empty
63          \ifx\@CTlabel\empty\else
64              \PackageError{lwarp-ctable}{\MessageBreak
65                  You may not label a captionless table\MessageBreak
66                  Such a label can't be referenced}
67                  {}
68          \fi
69      \fi
```

Some of the original, regarding computing the width of \CT@t, is removed here.

```
70      \@CTbegin
71          \ifx\@CTcontinued\empty\else\addtocounter{\@CTtaborfig}{-1}\fi
72          \@CTalign
```

lwarp's patches begin here:

```
73      \begin{center}
74          \setlength{\fboxrule}{\@CTframerule}
75          \setlength{\fboxsep}{\@CTframesep}
76          \LWR@forceminwidth{\fboxrule}%  lwarp
77          \convertcolorspec{named}{\@CTbgactual}{HTML}\LWR@tempcolor% lwarp
78          \begin{BlockClass}[%                          lwarp
79              border:
80                  \LWR@printlength{\LWR@atleastonept}
81                  solid
82                  \LWR@colorstyle{named}{\@CTfgactual} ; %
83              padding:\LWR@printlength{\fboxsep} ; %
84              \ifdefstring{\LWR@tempcolor}{FFFFFF}{}{%
85                  background: \LWR@colorstyle{named}{\@CTbgactual} ; %
86              }%
87          ]{fminipage}%      lwarp
88          \ifx\@CTbotcap\@CTfalse\@CTCaption\vskip\@CTcaptionskip\fi
89          \ifx\@CTbotcap\undefined%
90              \begin{sidecaption}[\@CTcap]{\@CTcaption}[\@CTlabel]
91          \fi
92          \@CTdoinside
93          \begin{tabularx}{\linewidth}{#2}%      lwarp
94              #4%
95          \end{tabularx}%                        lwarp
```

```
96              \def\@CTfootnotes{#3}%
97              \ifx#3\empty\else{% append footnotes, if any
98                 \begin{BlockClass}{tnotes}%        lwarp
99                 #3
100                \end{BlockClass}%                  lwarp
101             }
102             \fi
103             \ifx\@CTbotcap\undefined\end{sidecaption}\fi
104             \ifx\@CTbotcap\@CTtrue\vskip\@CTcaptionskip\@CTCaption\fi
105          \end{BlockClass}
106       \end{center}
107    \@CTend
108 }
109 \LWR@formatted{ctable}
```

Required to properly detect the toprule:

```
110 \LetLtxMacro\FL\toprule
```

Table notes are redefined for HTML:

```
111 \newcommand{\LWR@HTML@tmark}[1][a]{%
112    \textsuperscript{\textrm{\textit{#1}}}
113 }
114 \LWR@formatted{tmark}
115
116 \newcommand{\LWR@HTML@tnote}[2][a]{%
117    \tmark[#1]\,#2\par
118 }
119 \LWR@formatted{tnote}
```

File 105   **`lwarp-cuted.sty`**

§ 217   Package   # cuted

(*Emulates or patches code by* Sigitas Tolušis.)

cuted (*Pkg*)   cuted is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{cuted}[2021/10/04]`

```
2 \newenvironment{strip}{}{}
3 \newskip\stripsep
4 \newtoks\preCutedStrip \preCutedStrip{}
5 \newtoks\postCutedStrip \postCutedStrip{}
6 \def\oldcolsbreak#1{}
```

File 106   **`lwarp-cutwin.sty`**

§ 218   Package   # cutwin

(*Emulates or patches code by* Peter Wilson and Alan Hoenig.)

cutwin (*Pkg*)   cutwin is emulated.

for HTML output: Discard all options for lwarp-cutwin:

```
1 \LWR@ProvidesPackageDrop{cutwin}[2010/09/29]


2 \newcommand*{\opencutleft}{}
3 \newcommand*{\opencutright}{}
4 \newcommand*{\opencutcenter}{}
5 \newcommand*{\cutfuzz}{}
6
7 \newenvironment{cutout}[4]
8 {\marginpar{\windowpagestuff}}
9 {}
10
11 \newcommand*{\windowpagestuff}{}
12
13 \newcommand*{\pageinwindow}{%
14 % \begin{minipage}{.3\linewidth}
15 \windowpagestuff
16 % \end{minipage}
17 }
18
19 \newenvironment{shapedcutout}[3]
20 {\marginpar{\picinwindow}}
21 {}
22
23 \newcommand*{\putstuffinpic}{}
24
25 \newcommand*{\picinwindow}{%
26 \begin{picture}(0,0)
27 \putstuffinpic
28 \end{picture}}
```

File 107   `lwarp-dblfloatfix.sty`

§ 219   Package   # dblfloatfix

dblfloatfix (*Pkg*)   dblfloatfix is ignored.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{dblfloatfix}[2012/12/31]
```

File 108   `lwarp-dblfnote.sty`

§ 220   Package   # dblfnote

(*Emulates or patches code by* Hiroshi Nakashima.)

dblfnote (*Pkg*)   dblfnote is ignored.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{dblfnote}[1999/07/14]


2 \newcounter{DFNsloppiness}
3 \newdimen\DFNcolumnsep
4 \newdimen\DFNcolumnwidth
5 \def\DFNallowcbreak{}
```

```
6 \def\DFNinhibitcbreak{}
7 \def\DFNtrysingle{}
8 \def\DFNalwaysdouble{}
9 \def\DFNruleboth{}
10 \def\DFNruleleft{}
```

File 109   **`lwarp-dcolumn.sty`**

§ 221   Package   # dcolumn

dcolumn (*Pkg*)   dcolumn is used as-is in a lateximage, and is emulated by the lwarp core.

dcolumn used to be \LWR@ProvidesPackageDrop in prior versions of lwarp, but is now supported for print mode.

```
1 \LWR@ProvidesPackagePass{dcolumn}[2014/10/28]
```

Due to how the D column is created, cannot use \HTMLnewcolumntype here. An HTML version neutralizes the lower-level macros, leaving a c column type.

```
2 \newcommand*{\LWR@HTML@DC@}[3]{}
3 \LWR@formatted{DC@}
4
5 \providecommand*{\DC@end}{}
6
7 \newcommand*{\LWR@HTML@DC@end}{}
8 \LWR@formatted{DC@end}
```

File 110   **`lwarp-decimal.sty`**

§ 222   Package   # decimal

(*Emulates or patches code by* A. Syropoulos and R. W. D. Nickalls.)

decimal (*Pkg*)   decimal works as-is for svg math, and is emulated for MathJax.

for HTML output: 1 `\LWR@ProvidesPackagePass{decimal}[2011/06/03]`

```
2 \begin{warpMathJax}
3 \CustomizeMathJax{\def\.{\mbox{.}}}
4 \end{warpMathJax}
```

File 111   **`lwarp-decorule.sty`**

§ 223   Package   # decorule

(*Emulates or patches code by* Peter Flynn.)

decorule (*Pkg*)   decorule is patched for use by lwarp.

for HTML output: 1 `\LWR@ProvidesPackagePass{decorule}[2020/04/01]`

```
2 \xpretocmd{\decorule}
3    {\begin{lateximage}*[decorule]}
4    {}
5    {\LWR@patcherror{decorule}{decorule A}}
6
7 \xapptocmd{\decorule}
8    {\end{lateximage}}
9    {}
10   {\LWR@patcherror{decorule}{decorule B}}
```

File 112   **`lwarp-diagbox.sty`**

§ 224   Package   # diagbox

(*Emulates or patches code by* Leo Liu.)

diagbox (*Pkg*)   diagbox is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{diagbox}[2016/12/28]

To restore print-mode inside a lateximage:

```
2 \LetLtxMacro\LWR@origdiagbox@double\diagbox@double
3 \LetLtxMacro\LWR@origdiagbox@triple\diagbox@triple
4
5 \appto\LWR@restoreorigformatting{%
6 \LetLtxMacro\diagbox@double\LWR@origdiagbox@double%
7 \LetLtxMacro\diagbox@triple\LWR@origdiagbox@triple%
8 }
```

\LWR@diagbox@AB $\{\langle E/W\rangle\}\,\{\langle A\rangle\}\,\{\langle E/W\rangle\}\,\{\langle B\rangle\}$

```
9  \newcommand{\LWR@diagbox@AB}[4]{
10 \begingroup%
11 \LetLtxMacro\\\newline%
12 \BlockClassSingle{diagbox#1}{#2}%
13 \BlockClassSingle{diagbox#3}{#4}%
14 \endgroup%
15 \LWR@stoppars%
16 }
```

\LWR@diagboxNW $\{\langle A\rangle\}\,\{\langle B\rangle\}$

```
17 \newcommand{\LWR@diagboxNW}[2]{%
18 \LWR@diagbox@AB{E}{#2}{W}{#1}%
19 }
```

Likewise for NE, SW, SE:

```
20 \newcommand{\LWR@diagboxNE}[2]{%
21 \LWR@diagbox@AB{W}{#1}{E}{#2}%
22 }
23
24 \let\LWR@diagboxSW\LWR@diagboxNE
25 \let\LWR@diagboxSE\LWR@diagboxNW
```

\diagbox@double $\{\langle keys\rangle\}\ \{\langle A\rangle\}\ \{\langle B\rangle\}$

26 \def\diagbox@double#1#2#3{%
27 \setkeys{diagbox}{dir=NW,#1}%
28 \@nameuse{LWR@diagbox\diagbox@dir}{#2}{#3}%
29 }

\LWR@diagboxTNW $\{\langle title\rangle\}\ \{\langle A\rangle\}\ \{\langle B\rangle\}$

30 \newcommand{\LWR@diagboxTNW}[3]{%
31 \BlockClassSingle{diagboxtitleN}{#1}
32 \LWR@diagboxNW{#2}{#3}
33 }

Likewise for NE, SW, SE:

34 \newcommand{\LWR@diagboxTNE}[3]{%
35 \BlockClassSingle{diagboxtitleN}{#1}
36 \LWR@diagboxNE{#2}{#3}
37 }
38
39 \newcommand{\LWR@diagboxTSW}[3]{%
40 \LWR@diagboxSW{#2}{#3}
41 \BlockClassSingle{diagboxtitleS}{#1}
42 \LWR@stoppars%
43 }
44
45 \newcommand{\LWR@diagboxTSE}[3]{%
46 \LWR@diagboxSE{#2}{#3}
47 \BlockClassSingle{diagboxtitleS}{#1}
48 \LWR@stoppars%
49 }

\diagbox@triple $\{\langle keys\rangle\}\ \{\langle A\rangle\}\ \{\langle T\rangle\}\ \{\langle B\rangle\}$

50 \def\diagbox@triple#1#2#3#4{%
51 \setkeys{diagbox}{dir=NW,#1}%
52 \@nameuse{LWR@diagboxT\diagbox@dir}{#3}{#2}{#4}%
53 }

File 113   **lwarp-dingbat.sty**

§ 225   Package   # dingbat

(*Emulates or patches code by* SCOTT PAKIN.)

dingbat (*Pkg*)   dingbat is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{dingbat}[2001/04/27]

2 \newcommand*{\LWR@dingbatsymbol}[1]{\HTMLunicode{#1}}
3
4 \newcommand{\LWR@HTML@rightpointright}{\LWR@dingbatsymbol{261E}}
5 \newcommand{\LWR@HTML@leftpointright}{\LWR@dingbatsymbol{261E}}
6 \newcommand{\LWR@HTML@leftthumbsdown}{\LWR@dingbatsymbol{1F44E}}
7 \newcommand{\LWR@HTML@leftthumbsup}{\LWR@dingbatsymbol{1F44D}}

8 \newcommand{\LWR@HTML@rightpointleft}{\LWR@dingbatsymbol{261C}}
9 \newcommand{\LWR@HTML@rightthumbsdown}{\LWR@dingbatsymbol{1F44E}}
10 \newcommand{\LWR@HTML@rightthumbsup}{\LWR@dingbatsymbol{1F44D}}
11 \newcommand{\LWR@HTML@squarewithdots}{\LWR@dingbatsymbol{25C7}}
12 \newcommand{\LWR@HTML@filledsquarewithdots}{\LWR@dingbatsymbol{25C6}}
13 \newcommand{\LWR@HTML@Sborder}{\LWR@dingbatsymbol{271A}}
14 \newcommand{\LWR@HTML@Zborder}{\LWR@dingbatsymbol{274B}}
15 \newcommand{\LWR@HTML@largepencil}{\LWR@dingbatsymbol{270E}}
16 \newcommand{\LWR@HTML@anchor}{\LWR@dingbatsymbol{2693}}
17 \newcommand{\LWR@HTML@carriagereturn}{\LWR@dingbatsymbol{23CE}}
18 \newcommand{\LWR@HTML@checkmark}{\LWR@dingbatsymbol{2713}}
19 \newcommand{\LWR@HTML@eye}{\LWR@dingbatsymbol{1F441}}
20 \newcommand{\LWR@HTML@satellitedish}{\LWR@dingbatsymbol{1F4E1}}
21 \newcommand{\LWR@HTML@smallpencil}{\LWR@dingbatsymbol{270E}}
22
23 \LWR@formatted{rightpointright}
24 \LWR@formatted{leftpointright}
25 \LWR@formatted{leftthumbsdown}
26 \LWR@formatted{leftthumbsup}
27 \LWR@formatted{rightpointleft}
28 \LWR@formatted{rightthumbsdown}
29 \LWR@formatted{rightthumbsup}
30 \LWR@formatted{squarewithdots}
31 \LWR@formatted{filledsquarewithdots}
32 \LWR@formatted{Sborder}
33 \LWR@formatted{Zborder}
34 \LWR@formatted{largepencil}
35 \LWR@formatted{anchor}
36 \LWR@formatted{carriagereturn}
37 \LWR@formatted{checkmark}
38 \LWR@formatted{eye}
39 \LWR@formatted{satellitedish}
40 \LWR@formatted{smallpencil}

File 114 **`lwarp-doipubmed.sty`**

§ 226   Package   # doipubmed

(*Emulates or patches code by* Nicola Talbot.)

doipubmed (*Pkg*)   doipubmed is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{doipubmed}[2007/08/20]

2 \VerifyCommand[lwarp][doipubmed]{\doi}{13FFCBAF4F1414B838B6C3AD344117A8}
3
4 \renewcommand*{\doi}[1]{%
5 \def\@doi@code{}%
6 \@doi@linksubs#1\#\@@\@doi@code
7 \@onelevel@sanitize{\@doi@code}%
8 \def\@doi@text{}%
9 \@doi@textsubs#1\@nil\@@\@doi@text%
10 \xpretocmd{\@doi@code}{http://dx.doi.org/}{}{}%
11 \expandafter\href\expandafter{\@doi@code}{\doitext{\@doi@text}}}}

Must not modify catcodes before using \url:

```
12 \DeclareDocumentCommand{\LWR@citeurlb}{m}{%
13     \LWR@ensuredoingapar%
14     \textless%
15     \LWR@href@sanitized{#1}{#1}%
16     \textgreater%
17     \endgroup%
18 }
19
20 \renewrobustcmd*{\citeurl}{%
21     \begingroup%
22     \LWR@linkcatcodes%
23     \LWR@citeurlb%
24 }
```

File 115   **lwarp-DotArrow.sty**

§ 227   Package   # DotArrow

(*Emulates or patches code by* Sven Schneider.)

DotArrow (*Pkg*)   **DotArrow** is patched for use by lwarp, and emulated for MathJax.

for HTML output: 1 \LWR@ProvidesPackagePass{DotArrow}[2007/02/12]

The width must be recomputed each time, depending on print or HTML output.

```
2 \xpretocmd{\dotarrow}{\settowidth{\oneWidth}{\onePartX}}{}{}
3
4 \begin{warpMathJax}
5 \CustomizeMathJax{\newcommand{\dotarrow}[1]{\stackrel{#1}{\unicode{x21E2}}}}
6 \end{warpMathJax}
```

File 116   **lwarp-dotlessi.sty**

§ 228   Package   # dotlessi

(*Emulates or patches code by* Javier Bezos.)

dotlessi (*Pkg*)   **dotlessi** is used as-is for svg math, and is emulated for MathJax.

⚠ HTML \dotlessj   Use \usepackage{cmap} if \dotlessj does not appear in HTML in text mode. See section 7.4.

⚠ not bold   For MathJax, use \boldsymbol instead of \mathbf.

for HTML output: 1 \LWR@ProvidesPackagePass{dotlessi}[1999/10/12]

For MathJax:

```
2 \begin{warpMathJax}
3 \CustomizeMathJax{\let\dotlessi\imath}
4 \CustomizeMathJax{\let\dotlessj\jmath}
5 \end{warpMathJax}
```

File 117 **`lwarp-dprogress.sty`**

§ 229 Package **dprogress**

dprogress (*Pkg*) dprogress is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{dprogress}[2008/02/21]

File 118 **`lwarp-draftcopy.sty`**

§ 230 Package **draftcopy**

draftcopy (*Pkg*) draftcopy is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{draftcopy}[2002/02/25]

2 \newcommand{\draftcopyVersion}[1]{}
3 \newcommand{\draftcopySetGrey}[1]{}
4 \newcommand{\draftcopySetScale}[1]{}
5 \newcommand{\draftcopySetScaleFactor}[1]{}
6 \newcommand{\draftcopyFirstPage}[1]{}
7 \newcommand{\draftcopyLastPage}[1]{}
8 \newcommand{\draftcopyName}[2]{}
9 \newcommand{\draftcopyPageTransform}[1]{}
10 \newcommand{\draftcopyBottomTransform}[1]{}
11 \newcommand{\draftcopyPageX}[1]{}
12 \newcommand{\draftcopyPageY}[1]{}
13 \newcommand{\draftcopyBottomX}[1]{}
14 \newcommand{\draftcopyBottomY}[1]{}

File 119 **`lwarp-draftfigure.sty`**

§ 231 Package **draftfigure**

draftfigure (*Pkg*) draftfigure is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{draftfigure}[2017/07/19]
2 \RequirePackage{xkeyval}

3 \define@key{draftfigure}{code}{}
4 \define@key{draftfigure}{noframe}[true]{}
5 \define@key{draftfigure}{filename}[true]{}
6 \define@key{draftfigure}{content}[]{}
7 \define@key{draftfigure}{style}[normal]{}
8 \define@key{draftfigure}{position}[left]{}
9 \define@key{draftfigure}{size}[normal]{}
10 \newcommand\setdf[1]{\setkeys{draftfigure}{#1}}

File 120   **`lwarp-draftwatermark.sty`**

§ 232   Package   # draftwatermark

(*Emulates or patches code by* Sergio Callegari.)

draftwatermark (*Pkg*)   draftwatermark is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{draftwatermark}[2020/03/14]

2 \newcommand{\DraftwatermarkOptions}[1]{}
3 \newcommand{\DraftwatermarkStdMark}{}
4 \newcommand{\SetWatermarkAngle}[1]{}
5 \newcommand{\SetWatermarkColor}[1]{}
6 \newcommand{\SetWatermarkLightness}[1]{}
7 \newcommand{\SetWatermarkFontSize}[1]{}
8 \newcommand{\SetWatermarkScale}[1]{}
9 \newcommand{\SetWatermarkHorCenter}[1]{}
10 \newcommand{\SetWatermarkVertCenter}[1]{}
11 \newcommand{\SetWatermarkText}[1]{}

File 121   **`lwarp-drftcite.sty`**

§ 233   Package   # drftcite

(*Emulates or patches code by* Donald Arseneau.)

drftcite (*Pkg*)   drftcite is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{drftcite}[1995/01/23]

2 \VerifyCommand[lwarp][drftcite]{\@lbibitem}{43265BD7F1B9C9818D873D651C19485C}
3
4 \def\@lbibitem[#1]#2{\global\@HighCite\z@
5   \item[
6   \textsuperscript{\@nameuse{DCN@#2\@extra@b@citeb}}~%   lwarp
7   \@biblabel{\@ifundefined{DCN@#2\@extra@b@citeb}{\@warning
8   {Reference `#2' on page \thepage\space was never cited}}{}%
9 %   \DC@llap{$^{\@nameuse{DCN@#2\@extra@b@citeb}}$\ \ }%%o
10   \@citeverb{#2}}\hfil]\if@filesw{\def\protect##1{\string ##1\space}%
11   \immediate\write\@auxout{\string\bibcite{#2}{#1}}}\fi\ignorespaces}

File 122   **`lwarp-easy-todo.sty`**

§ 234   Package   # easy-todo

(*Emulates or patches code by* Juan Rada-Vilela.)

easy-todo (*Pkg*)   easy-todo is patched for use by lwarp.

To remove the "P." heading for HTML:

```
\warpHTMLonly{\renewcommand{\todoindexpagetitle}{}}
```

for HTML output: 1 `\LWR@ProvidesPackagePass{easy-todo}[2014/01/01]`

`\listoftodos`   Modified to correct buggy use of `\flushright`.

```
2 \let\LWR@easytodo@origlistoftodos\listoftodos
3
4 \renewcommand{\listoftodos}{%
5 \begingroup
6 \renewcommand{\flushright}{}
7 \LWR@easytodo@origlistoftodos
8 \endgroup
9 }
```

`\todoii`   Modified to use `\textcolor` instead of `\color`.

```
10 \VerifyCommand[lwarp][easy-todo]{\todoii}{04C63A894C30C706AC60DD6B58FDEDA2}
11
12 \renewcommand{\todoii}[2]{%
13 \ifthenelse{\equal{\@todoobeyfinal}{true}}%
14     {%
15         \ifoptionfinal{\todoenable{false}}{\todoenable{true}}%
16     }%
17     {}%
18 \ifthenelse{\equal{\@todoenable}{true}}%
19     {%
20         \refstepcounter{todos}%
21         \noindent{%
22             \todocolor%
23             \LWR@textcurrentcolor{%
24                 \normalfont\scriptsize{\bfseries{\thetodos.#1}}%
25             }%
26         }%
27         \addcontentsline{lod}{todos}{\protect{\thetodos. }\LWR@isolate{#2}}%
28     }%
29     {}%
30 }
```

File 123   **lwarp-ebook.sty**

§ 235   Package   # ebook

(*Emulates or patches code by* JØRGEN STEENSGAARD.)

ebook (*Pkg*)   ebook is ignored.

for HTML output: 1 `\LWR@ProvidesPackageDrop{ebook}`

```
2 \setcounter{secnumdepth}{0}
3 \setcounter{tocdepth}{2}
4
5 \providecommand{\pagefill}[1][0.001mm]{\noindent}
6
7 \providecommand{\ebook}{
8 \setcounter{secnumdepth}{0}
```

9 \setcounter{tocdepth}{2}
10 }

File 124 **`lwarp-econometrics.sty`**

§ 236   Package   # econometrics

(*Emulates or patches code by* ERIK KOLE.)

econometrics (*Pkg*)   econometrics is used as-is for SVG math, and is emulated for MATHJAX.

for HTML output: 1 \LWR@ProvidesPackagePass{econometrics}% no date specified in the original

2 \LWR@origRequirePackage{lwarp-common-mathjax-letters}
3
4 \begin{warpMathJax}
5 \LWR@infoprocessingmathjax{econometrics}
6
7 \CustomizeMathJax{\newcommand{\SC}{\mathbb{C}}}
8 \CustomizeMathJax{\newcommand{\SN}{\mathbb{N}}}
9 \CustomizeMathJax{\newcommand{\SQ}{\mathbb{Q}}}
10 \CustomizeMathJax{\newcommand{\SR}{\mathbb{R}}}
11 \CustomizeMathJax{\newcommand{\SZ}{\mathbb{Z}}}
12
13 \CustomizeMathJax{\newcommand{\calA}{\mathcal{A}}}
14 \CustomizeMathJax{\newcommand{\calB}{\mathcal{B}}}
15 \CustomizeMathJax{\newcommand{\calC}{\mathcal{C}}}
16 \CustomizeMathJax{\newcommand{\calD}{\mathcal{D}}}
17 \CustomizeMathJax{\newcommand{\calE}{\mathcal{E}}}
18 \CustomizeMathJax{\newcommand{\calF}{\mathcal{F}}}
19 \CustomizeMathJax{\newcommand{\calG}{\mathcal{G}}}
20 \CustomizeMathJax{\newcommand{\calH}{\mathcal{H}}}
21 \CustomizeMathJax{\newcommand{\calI}{\mathcal{I}}}
22 \CustomizeMathJax{\newcommand{\calJ}{\mathcal{J}}}
23 \CustomizeMathJax{\newcommand{\calK}{\mathcal{K}}}
24 \CustomizeMathJax{\newcommand{\calL}{\mathcal{L}}}
25 \CustomizeMathJax{\newcommand{\calM}{\mathcal{M}}}
26 \CustomizeMathJax{\newcommand{\calN}{\mathcal{N}}}
27 \CustomizeMathJax{\newcommand{\calO}{\mathcal{O}}}
28 \CustomizeMathJax{\newcommand{\calP}{\mathcal{P}}}
29 \CustomizeMathJax{\newcommand{\calQ}{\mathcal{Q}}}
30 \CustomizeMathJax{\newcommand{\calR}{\mathcal{R}}}
31 \CustomizeMathJax{\newcommand{\calS}{\mathcal{S}}}
32 \CustomizeMathJax{\newcommand{\calT}{\mathcal{T}}}
33 \CustomizeMathJax{\newcommand{\calU}{\mathcal{U}}}
34 \CustomizeMathJax{\newcommand{\calV}{\mathcal{V}}}
35 \CustomizeMathJax{\newcommand{\calW}{\mathcal{W}}}
36 \CustomizeMathJax{\newcommand{\calX}{\mathcal{X}}}
37 \CustomizeMathJax{\newcommand{\calY}{\mathcal{Y}}}
38 \CustomizeMathJax{\newcommand{\calZ}{\mathcal{Z}}}
39
40 \LWR@mathjax@addlatin@u@bfit{m}% uppercase Latin, bold italic
41 \LWR@mathjax@addlatin@l@bfit{v}% lowercase Latin, bold italic
42
43 \LWR@mathjax@addgreek@l@bfit{v}{}% lowercase Greek bold italic
44 \LWR@mathjax@addgreek@u@bfit*{m}{}% uppercase Greek bold italic, capitalized macro names
45

```
46 \CustomizeMathJax{\newcommand{\rb}{\mathrm{b}}}
47 \CustomizeMathJax{\newcommand{\rB}{\mathrm{B}}}
48 \CustomizeMathJax{\newcommand{\rC}{\mathrm{C}}}
49 \CustomizeMathJax{\newcommand{\rD}{\mathrm{D}}}
50 \CustomizeMathJax{\newcommand{\rf}{\mathrm{f}}}
51 \CustomizeMathJax{\newcommand{\rF}{\mathrm{F}}}
52 \CustomizeMathJax{\newcommand{\rH}{\mathrm{H}}}
53 \CustomizeMathJax{\newcommand{\rL}{\mathrm{L}}}
54 \CustomizeMathJax{\newcommand{\rN}{\mathrm{N}}}
55 \CustomizeMathJax{\newcommand{\rt}{\mathrm{t}}}
56 \CustomizeMathJax{\newcommand{\rU}{\mathrm{U}}}
57 \CustomizeMathJax{\newcommand{\rGam}{\mathrm{Gam}}}
58 \CustomizeMathJax{\newcommand{\rBeta}{\mathrm{Beta}}}
59
60 \CustomizeMathJax{\newcommand{\Bin}{\mathrm{Bin}}}
61 \CustomizeMathJax{\newcommand{\eu}{\mathrm{e}}}
62 \CustomizeMathJax{\newcommand{\iu}{\mathrm{i}}}
63 \CustomizeMathJax{\newcommand{\LN}{\mathrm{LN}}}
64 \CustomizeMathJax{\newcommand{\IN}{\mathrm{IN}}}
65
66 \CustomizeMathJax{\newcommand{\Poi}{\mathrm{Poi}}}
67
68 \CustomizeMathJax{\newcommand{\ped}[1]{_\mathrm{#1}}}
69 \CustomizeMathJax{\newcommand{\ap}[1]{^\mathrm{#1}}}
70 \CustomizeMathJax{\renewcommand{\Re}{\mathrm{Re}}\nolimits}}
71 \CustomizeMathJax{\renewcommand{\Im}{\mathrm{Im}}\nolimits}}
72
73 \CustomizeMathJax{\newcommand{\deriv}[3][]{%
74     \frac{\mathrm{d}^{#1}#2}{\mathrm{d}\,#3^{#1}}%
75 }}
76 \CustomizeMathJax{\newcommand{\pderiv}[3][]{%
77     \frac{\partial^{#1}#2}{\partial #3^{#1}}%
78 }}
79
80 \CustomizeMathJax{\newcommand{\bias}{\operatorname{bias}}}
81 \CustomizeMathJax{\newcommand{\col}{\operatorname{col}}}
82 \CustomizeMathJax{\newcommand{\corr}{\operatorname{corr}}}
83 \CustomizeMathJax{\newcommand{\cov}{\operatorname{cov}}}
84 \CustomizeMathJax{\newcommand{\dg}{\operatorname{dg}}}
85 \CustomizeMathJax{\newcommand{\diag}{\operatorname{diag}}}
86 \CustomizeMathJax{\newcommand{\E}{\operatorname{E}}}
87 \CustomizeMathJax{\newcommand{\etr}{\operatorname{etr}}}
88 \CustomizeMathJax{\newcommand{\ip}{\mathrm{int}}\nolimits}}
89 \CustomizeMathJax{\newcommand{\kur}{\operatorname{kur}}}
90 \CustomizeMathJax{\newcommand{\MSE}{\operatorname{MSE}}}
91 \CustomizeMathJax{\newcommand{\MSFE}{\operatorname{MSFE}}}
92 \CustomizeMathJax{\newcommand{\OLS}{\operatorname{OLS}}}
93 \CustomizeMathJax{\newcommand{\plim}{\operatorname{plim}}}
94 \CustomizeMathJax{\newcommand{\resid}{\operatorname{resid}}}
95 \CustomizeMathJax{\newcommand{\rk}{\operatorname{rk}}}
96 \CustomizeMathJax{\newcommand{\SE}{\operatorname{SE}}}
97 \CustomizeMathJax{\newcommand{\sgn}{\operatorname{sgn}}}
98 \CustomizeMathJax{\newcommand{\tr}{\operatorname{tr}}}
99 \CustomizeMathJax{\newcommand{\var}{\operatorname{var}}}
100 \CustomizeMathJax{\renewcommand{\vec}{\operatorname{vec}}}
101 \CustomizeMathJax{\newcommand{\vech}{\operatorname{vech}}}
102
103 \CustomizeMathJax{\newcommand{\distr}{\sim}}
104 \CustomizeMathJax{\newcommand{\adistr}{\stackrel{a}{\distr}}}
105 \CustomizeMathJax{\newcommand{\diff}{\Delta}}
```

```
106 \CustomizeMathJax{\newcommand{\fdiff}{\diff_{\rf}}}
107 \CustomizeMathJax{\newcommand{\bdiff}{\diff_{\rb}}}
108
109 \CustomizeMathJax{\newcommand{\eps}{\epsilon}}
110 \CustomizeMathJax{\newcommand{\epsi}{\varepsilon}}
111
112 \CustomizeMathJax{\newcommand{\longto}{\longrightarrow}}
113 \CustomizeMathJax{\newcommand{\pto}{\stackrel{p}{\longrightarrow}}}
114 \CustomizeMathJax{\newcommand{\dto}{\stackrel{d}{\longrightarrow}}}
115 \CustomizeMathJax{\newcommand{\wto}{\stackrel{w}{\longrightarrow}}}
116
117 \CustomizeMathJax{\newcommand{\Infmat}{\bm\calI}}
118 \CustomizeMathJax{\newcommand{\Hesmat}{\bm\calH}}
119 \CustomizeMathJax{\newcommand{\bcdot}{\bullet}}
120
121 \CustomizeMathJax{\newcommand{\vones}{\bm\imath}}
122 \CustomizeMathJax{\newcommand{\vzeros}{\boldsymbol{0}}}
123 \CustomizeMathJax{\newcommand{\mZeros}{\mathbf{O}}}
124
125 \CustomizeMathJax{\newcommand{\e}{\eu}}
126 \CustomizeMathJax{\newcommand{\mply}{\cdot}}
127 \CustomizeMathJax{\newcommand{\rW}{\ensuremath{\mathrm{W}}}}
128 \end{warpMathJax}
```

File 125   **`lwarp-ed.sty`**

§ 237   Package   **ed**

(*Emulates or patches code by* MICHAEL KOHLHASE.)

ed (*Pkg*)   ed is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{ed}[2012/01/29]

Bugs:

1. todolist fails with the hide option, as does \edexplanation.

2. \edstubURI is actually \edstuURI.

```
2 \RequirePackage{xcolor}
3
4 \renewenvironment{edstub}[2][The following blue text]
5 {%
6     \def\@test{#1}%
7     \begin{center}%
8         \huge%
9         \textcolor{red}{%
10            #1 is only a provisional stub\\\Large
11            the Office document
12            \ifx\ed@stubURI\@empty{#2}\else\LWR@href{\ed@stubURI}{#2}\fi\
13            contains more text\\which will be merged for the final document%
14        }%
15    \end{center}%
16    \BlockClass[color:blue]{edstub}%
17 }
18 {\endBlockClass}
```

File 126   **`lwarp-ellipsis.sty`**

§ 238   Package   # ellipsis

(*Emulates or patches code by* PETER J. HESLIN.)

ellipsis (*Pkg*)   ellipsis is emulated.

```
1 \LWR@ProvidesPackageDrop{ellipsis}[2004/09/28]
2
3 \newcommand{\ellipsisgap}{0.1em}
4
5 \newcommand*{\midwordellipsis}{\,\textellipsis\,}
```

File 127   **`lwarp-embrac.sty`**

§ 239   Package   # embrac

(*Emulates or patches code by* CLEMENS NIEDERBERGER.)

embrac (*Pkg*)   embrac is patched for HTML and used as-is for print.

for HTML output:
```
1 \LWR@ProvidesPackagePass{embrac}[2021/02/20]
```

```
2 \ExplSyntaxOn
3 \RenewDocumentCommand{\embrac_kern:n}{m}{}
4 \ExplSyntaxOff
```

```
5 \LetLtxMacro\LWR@orig@HTML@emph\LWR@HTML@emph
6 \RenewDocumentCommand{\LWR@HTML@emph}{s m}{\LWR@orig@HTML@emph{#2}}
7
8 \LetLtxMacro\LWR@orig@HTML@textit\LWR@HTML@textit
9 \RenewDocumentCommand{\LWR@HTML@textit}{s m}{\LWR@orig@HTML@textit{#2}}
10
11 \LetLtxMacro\LWR@orig@HTML@textsl\LWR@HTML@textsl
12 \RenewDocumentCommand{\LWR@HTML@textsl}{s m}{\LWR@orig@HTML@textsl{#2}}
13
14 \AtBeginDocument{
15     \LWR@formatted{emph}
16     \LWR@formatted{textit}
17     \LWR@formatted{textsl}
18     \ifdef{\textsi}
19     {
20         \LetLtxMacro\LWR@orig@HTML@textsi\LWR@HTML@textsi
21         \RenewDocumentCommand{\LWR@HTML@textsi}{s m}{%
22             \LWR@orig@HTML@textsi{#2}%
23         }
24         \LWR@formatted{textsi}
25     }{}
26 }
27
28 \newcommand{\LWR@HTML@EmbracOff}{}
```

```
29 \LWR@formatted{EmbracOff}
30
31 \newcommand{\LWR@HTML@EmbracOn}{}
32 \LWR@formatted{EmbracOn}
```

File 128   **`lwarp-emptypage.sty`**

§ 240   Package   # emptypage

emptypage (*Pkg*)   emptypage is ignored.

for HTML output: Discard all options for lwarp-emptypage:

```
1 \LWR@ProvidesPackageDrop{emptypage}[2010/05/30]
```

File 129   **`lwarp-endfloat.sty`**

§ 241   Package   # endfloat

endfloat (*Pkg*)   endfloat is ignored.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{endfloat}[2019/04/15]

2 \newcommand\figureplace{}
3 \newcommand\tableplace{}
4 \newcommand\floatplace[1]{}
5 \newcounter{posttable}
6 \newcounter{postfigure}
7 \newcommand*{\theposttbl}{}
8 \newcommand*{\thepostfig}{}
9 \newcommand{\AtBeginFigures}[1]{}
10 \newcommand{\AtBeginTables}[1]{}
11 \newcommand{\AtBeginDelayedFloats}[1]{}
12 \newcommand*{\processdelayedfloats}{}
13 \newcommand*{\efloatseparator}{}
14 \def\efloattype{}
15 \providecommand\efloatheading[1]{}
16 \providecommand\efloatpreamble{}
17 \providecommand\efloatpostamble{}
18 \NewDocumentCommand{\addtodelayedfloat}{s m m}{}
19 \providecommand{\efloatbegin}{}
20 \providecommand{\efloatend}{}
21 \providecommand{\efloatbeginlist}{}
22 \providecommand{\efloatendlist}{}
```

File 130   **`lwarp-endheads.sty`**

§ 242   Package   # endheads

endheads (*Pkg*)   endheads is ignored.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{endheads}[2017/04/06]
```

```
2 \newcommand{\changesinglepageabbrev}[1]{}
3 \newcommand{\changemultiplepageabbrev}[1]{}
4 \newcommand{\changenotesname}[1]{}
5 \newcommand{\changenotesheader}[1]{}
6 \newcommand{\changenotescontentsname}[1]{}
7 \newcommand{\changechapternotesline}[1]{}
8 \newcommand{\checknoteheaders}{}
9 \newif\ifnotesincontentson \notesincontentsonfalse
10 \newcommand{\notesincontents}{\notesincontentsontrue}
11 \newif\ifendnoteheaderson \endnoteheadersonfalse
12 \newcommand{\setupendnoteheaders}{%
13    \endnoteheadersontrue%
14 }
15 \newif\iftitleinnotes \titleinnotestrue
16 \newcommand{\styleforchapternotebegin}{}
17 \newcommand{\styleforchapternoteend}{}
18 \newcommand{\setstyleforchapternotebegin}[1]{%
19    \renewcommand{\styleforchapternotebegin}{#1}%
20 }
21 \newcommand{\setstyleforchapternoteend}[1]{%
22    \renewcommand{\styleforchapternoteend}{#1}%
23 }
24 \newcommand{\resetendnotes}{}
25 \newif\ifnotesbychapteron \notesbychapteronfalse
26 \newcommand{\notesbychapter}{\notesbychapterontrue}
```

File 131   **lwarp-endnotes.sty**

§ 243   Package   # endnotes

(*Emulates or patches code by* JOHN LAVAGNINO.)

endnotes (*Pkg*)   Patched for HTML.

If using cleveref, endnotes displays as a link to an endnote, rather than a section. A comma-separated list of end notes does not work with \cref and related. (In print mode, such as list simply displays a link to the section.)

table of contents   To place the endnotes in the TOC, use:

>   \usepackage{endnotes}
>   \appto\enoteheading{\addcontentsline{toc}{section}{\notesname}}
>   \renewcommand*{\notesname}{Endnotes} % optional

HTML page   To additionally have the endnotes on their own HTML page, if FileDepth allows:

>   \ForceHTMLPage
>   \theendnotes

⚠ \endnotemark numbering   If using MATHJAX, see section 8.5.4 regarding the use of \endnotemark and \endnotetext.

for HTML output:
```
1 \LWR@ProvidesPackagePass{endnotes}[2020-01-02]
```

```
2 \def\enoteformat{%
3 % \rightskip\z@ \leftskip\z@ \parindent=1.8em
```

```
4 \leavevmode
5 % \llap{
6 \makeenmark
7 % }
8 }
```

```
9 \def\LWR@HTML@@makeenmark{\hbox{\LWR@htmlspan{sup}{\normalfont\theenmark}}}
10 \LWR@formatted{@makeenmark}
11
12 \def\makeenmark{\@makeenmark}
```

To nullify the endnotes:

```
13 \apptocmd{\LWR@nullifyfootnotes}{%
14     \renewcommand{\endnote}[2][]{}%
15     \renewcommand{\endnotemark}[1]{}%
16 }{}{}
```

Modified for updated LaTeX labels.

```
17 \def\theendnotes{\immediate\closeout\@enotes \global\@enotesopenfalse
18   \begingroup
19     \makeatletter
20     %
21     % The machinery with \@ResetGT and > here ensures that
22     % \@doanenote works properly even if > is an active character
23     % at the point where \theendnotes is invoked. > needs to have
24     % catcode 12 when the arguments of \@doanenote are scanned, so
25     % that the > in the string "macro:->" is matched.  The actual
26     % footnote text is not an argument to \@doanenote, but just
27     % follows it in the .ent file; so \@ResetGT can reset the
28     % category code for > that should be used when processing
29     % that text.  That resetting takes place within a
30     % \begingroup-\endgroup block set up by \@doanenote and
31     % \@endanenote, so the catcode for > is back to 12 for the
32     % next note.
33     %
34     \edef\@tempa{`\string >}%
35     \ifnum\catcode\@tempa=12%
36       \let\@ResetGT\relax
37     \else
38       \edef\@ResetGT{\noexpand\catcode\@tempa=\the\catcode\@tempa}%
39       \@makeother\>%
40     \fi
41     \def\@doanenote##1##2>{\def\@theenmark{##1}\par\begingroup
42         \@ResetGT

43         \def\@currentcounter{endnote}%

44         \edef\@currentlabel{\csname p@endnote\endcsname\@theenmark}%
45         \enoteformat}
46     \def\@endanenote{\par\endgroup}%
47     \enoteheading
48     \enotesize
49     \InputIfFileExists{\jobname.ent}{}{%
50      \PackageWarning{endnotes}{No endnotes found (file \jobname.ent does not exist)\MessageBreak}
51     }%
52   \endgroup}
```

For MATHJAX:

```
53 \begin{warpMathJax}
54 \def\endnotename{endnote}
55 \appto\LWR@syncnotenumbers{\LWR@synconenotenumber{LWRendnote}{\theendnote}}
56 \appto\LWR@syncnotenames{\LWR@synconenotename{LWRendnote}{\endnotename}}
57 \CustomizeMathJax{\def\LWRendnote{1}}
58 \CustomizeMathJax{\newcommand{\endnote}[2][\LWRendnote]{{}^{\mathrm{#1}}}}
59 \CustomizeMathJax{\newcommand{\endnotemark}[1][\LWRendnote]{{}^{\mathrm{#1}}}}
60 \end{warpMathJax}
```

File 132   **`lwarp-engtlc.sty`**

§ 244   Package   # engtlc

(*Emulates or patches code by* CLAUDIO FIANDRINO.)

engtlc (*Pkg*)   engtlc is patched for use by lwarp. MATHJAX is emulated.

⚠   For MATHJAX, \signt, \signf, \signn, and \signz do not force letter case as they do in SVG math.

for HTML output: 1 `\LWR@ProvidesPackagePass{engtlc}[2012/12/18]`

```
2 \newcommand{\LWR@HTML@finees}{%
3     \begin{BlockClass}[text-align:right]{exerend}%
4     \HTMLunicode{220E}%
5     \end{BlockClass}%
6 }
7 \LWR@formatted{finees}
8
9 \newcommand{\LWR@HTML@exerend}{\finees}
10 \LWR@formatted{exerend}
11
12 \begin{warpMathJax}
13 \LWR@infoprocessingmathjax{engtlc}
14
15 \CustomizeMathJax{\newcommand{\unit}[1]{\,\mathrm{#1}}}
16 \CustomizeMathJax{\newcommand{\micro}{\mathrm{\unicode{x00B5}}}}
17 %
18 \CustomizeMathJax{\newcommand{\ho}{\unit{h}}}
19 \CustomizeMathJax{\newcommand{\s}{\unit{s}}}
20 \CustomizeMathJax{\newcommand{\ms}{\unit{ms}}}
21 \CustomizeMathJax{\newcommand{\us}{\unit{\micro s}}}
22 \CustomizeMathJax{\newcommand{\ns}{\unit{ns}}}
23 \CustomizeMathJax{\newcommand{\ps}{\unit{ps}}}
24 %
25 \CustomizeMathJax{\newcommand{\um}{\unit{\micro m}}}
26 \CustomizeMathJax{\newcommand{\mm}{\unit{mm}}}
27 \CustomizeMathJax{\newcommand{\cm}{\unit{cm}}}
28 \CustomizeMathJax{\newcommand{\dm}{\unit{dm}}}
29 \CustomizeMathJax{\newcommand{\m}{\unit{m}}}
30 \CustomizeMathJax{\newcommand{\km}{\unit{km}}}
31 %
32 \CustomizeMathJax{\newcommand{\MA}{\unit{MA}}}
33 \CustomizeMathJax{\newcommand{\kA}{\unit{kA}}}
34 \CustomizeMathJax{\newcommand{\A}{\unit{A}}}
```

```
35 \CustomizeMathJax{\newcommand{\mA}{\unit{mA}}}
36 \CustomizeMathJax{\newcommand{\uA}{\unit{\micro A}}}
37 \CustomizeMathJax{\newcommand{\nA}{\unit{nA}}}
38 %
39 \CustomizeMathJax{\newcommand{\MV}{\unit{MV}}}
40 \CustomizeMathJax{\newcommand{\kV}{\unit{kV }}}
41 \CustomizeMathJax{\newcommand{\V}{\unit{V}}}
42 \CustomizeMathJax{\newcommand{\mV}{\unit{mV}}}
43 \CustomizeMathJax{\newcommand{\uV}{\unit{\micro V}}}
44 %
45 \CustomizeMathJax{\newcommand{\mohm}{\unit{m\Omega}}}
46 \CustomizeMathJax{\newcommand{\ohm}{\unit{\Omega}}}
47 \CustomizeMathJax{\newcommand{\kohm}{\unit{k\Omega}}}
48 \CustomizeMathJax{\newcommand{\Mohm}{\unit{M\Omega}}}
49 %
50 \CustomizeMathJax{\newcommand{\pSi}{\unit{pS}}}
51 \CustomizeMathJax{\newcommand{\nSi}{\unit{nS}}}
52 \CustomizeMathJax{\newcommand{\uSi}{\unit{\micro S}}}
53 \CustomizeMathJax{\newcommand{\mSi}{\unit{mS}}}
54 \CustomizeMathJax{\newcommand{\Si}{\unit{S}}}
55 \CustomizeMathJax{\newcommand{\kSi}{\unit{kS}}}
56 \CustomizeMathJax{\newcommand{\MSi}{\unit{MS}}}
57 %
58 \CustomizeMathJax{\newcommand{\fFa}{\unit{fF}}}
59 \CustomizeMathJax{\newcommand{\pFa}{\unit{pF}}}
60 \CustomizeMathJax{\newcommand{\nFa}{\unit{nF}}}
61 \CustomizeMathJax{\newcommand{\uFa}{\unit{\micro F}}}
62 \CustomizeMathJax{\newcommand{\mFa}{\unit{mF}}}
63 \CustomizeMathJax{\newcommand{\Fa}{\unit{F}}}
64 %
65 \CustomizeMathJax{\newcommand{\fHe}{\unit{fH}}}
66 \CustomizeMathJax{\newcommand{\pHe}{\unit{pH}}}
67 \CustomizeMathJax{\newcommand{\nHe}{\unit{nH}}}
68 \CustomizeMathJax{\newcommand{\uHe}{\unit{\micro H}}}
69 \CustomizeMathJax{\newcommand{\mHe}{\unit{mH}}}
70 \CustomizeMathJax{\newcommand{\He}{\unit{H}}}
71 %
72 \CustomizeMathJax{\newcommand{\dB}{\unit{dB}}}
73 \CustomizeMathJax{\newcommand{\dBm}{\unit{dBm}}}
74 %
75 \CustomizeMathJax{\newcommand{\uW}{\unit{\micro W}}}
76 \CustomizeMathJax{\newcommand{\mW}{\unit{mW}}}
77 \CustomizeMathJax{\newcommand{\W}{\unit{W}}}
78 \CustomizeMathJax{\newcommand{\kW}{\unit{kW}}}
79 \CustomizeMathJax{\newcommand{\MW}{\unit{MW}}}
80 %
81 \CustomizeMathJax{\newcommand{\Hz}{\unit{Hz}}}
82 \CustomizeMathJax{\newcommand{\kHz}{\unit{kHz}}}
83 \CustomizeMathJax{\newcommand{\MHz}{\unit{MHz}}}
84 \CustomizeMathJax{\newcommand{\GHz}{\unit{GHz}}}
85 \CustomizeMathJax{\newcommand{\THz}{\unit{THz}}}
86 %
87 \CustomizeMathJax{\newcommand{\bit}{\unit{bit}}}
88 \CustomizeMathJax{\newcommand{\kbit}{\unit{Kib}}}
89 \CustomizeMathJax{\newcommand{\Mbit}{\unit{Mib}}}
90 \CustomizeMathJax{\newcommand{\Byte}{\unit{B}}}
91 \CustomizeMathJax{\newcommand{\kByte}{\unit{KiB}}}
92 \CustomizeMathJax{\newcommand{\MByte}{\unit{Mib}}}
93 \CustomizeMathJax{\newcommand{\GByte}{\unit{GiB}}}
94 \CustomizeMathJax{\newcommand{\TByte}{\unit{TiB}}}
```

```
 95 \CustomizeMathJax{\newcommand{\bits}{\unit{bit/s}}}
 96 \CustomizeMathJax{\newcommand{\kbits}{\unit{Kib/s}}}
 97 \CustomizeMathJax{\newcommand{\Mbits}{\unit{Mib/s}}}
 98 \CustomizeMathJax{\newcommand{\Bytes}{\unit{B/s}}}
 99 \CustomizeMathJax{\newcommand{\kBytes}{\unit{KiB/s}}}
100 \CustomizeMathJax{\newcommand{\MBytes}{\unit{MiB/s}}}
101 \CustomizeMathJax{\newcommand{\GBytes}{\unit{GiB/s}}}
102 \CustomizeMathJax{\newcommand{\TBytes}{\unit{TiB/s}}}
103 \CustomizeMathJax{\newcommand{\chips}{\unit{chip/s}}}
104 \CustomizeMathJax{\newcommand{\kchips}{\unit{Ki\mkern2mu chip/s}}}
105 \CustomizeMathJax{\newcommand{\Mchips}{\unit{Mi\mkern2mu chip/s}}}
106 \CustomizeMathJax{\newcommand{\chipsubit}{\unit{chip/bit}}}
107 %
108 \CustomizeMathJax{\newcommand{\frecciadex}[1][0.5]{%
109     \hspace{.25cm}\Longrightarrow \hspace{.25cm}}%
110 }
111 \CustomizeMathJax{\newcommand{\varianzarumore}{\frac{N_0}{2}}}
112 %
113 \CustomizeMathJax{\newcommand{\etsymbolbracearg}[2]{%
114     #1\mathopen{}\left\lbrace#2\right\rbrace\mathclose{}}%
115 }
116 \CustomizeMathJax{\newcommand{\fourier}[1]{\etsymbolbracearg{\mathcal{F}}{#1}}}
117 \CustomizeMathJax{\newcommand{\invfourier}[1]{\etsymbolbracearg{\mathcal{F}^{-1}}{#1}}}
118 \CustomizeMathJax{\newcommand{\partereale}[1]{\etsymbolbracearg{\textbf{Re}}{#1}}}
119 \CustomizeMathJax{\newcommand{\parteimm}[1]{\etsymbolbracearg{\textbf{Im}}{#1}}}
120 \CustomizeMathJax{\newcommand{\Info}[1]{I\left(#1\right)}}
121 \CustomizeMathJax{\newcommand{\versore}[1]{\hat{#1}}}
122 \CustomizeMathJax{\newcommand{\vettore}[1]{\overrightarrow{#1}}}
123 \CustomizeMathJax{\newcommand{\coseno}[1]{\cos\left(2\pi#1t\right)}}
124 \CustomizeMathJax{\newcommand{\seno}[1]{\sin\left(2\pi#1t\right)}}
125 \CustomizeMathJax{\newcommand{\energia}[1]{\mathcal{E}_{#1}}}
126 \CustomizeMathJax{\newcommand{\moduloexp}[2]{\left\vert#1\right\vert^{#2}}}
127 \CustomizeMathJax{\newcommand{\modulo}[1]{\left\vert#1\right\vert}}
128 \CustomizeMathJax{\newcommand{\indB}[1]{%
129     \mathopen{}\left.#1\right\vert_{\mathrm{dB}}\mathclose{}}%
130 \CustomizeMathJax{\newcommand{\for}[2]{\left. #1 \right\vert_{#2}}}
131 \CustomizeMathJax{\newcommand{\massimo}[1]{\etsymbolbracearg{\max}{#1}}}
132 \CustomizeMathJax{\newcommand{\minimo}[1]{\etsymbolbracearg{\min}{#1}}}
133 \CustomizeMathJax{\newcommand{\valc}{3\cdot 10^8}}
134 \CustomizeMathJax{\newcommand{\loga}[2]{\log_{#1}#2}}
135 \CustomizeMathJax{\newcommand{\analitic}[1]{\mathring{#1}}}
136 \CustomizeMathJax{\newcommand{\diff}{\mathop{}\mathopen{\mathrm{d}}}}
137 \CustomizeMathJax{\newcommand{\intinf}[1]{\int_{-\infty}^{+\infty}{#1}}}
138 \CustomizeMathJax{\newcommand{\deltain}[1]{\delta\left(#1\right)}}
139 \CustomizeMathJax{\newcommand{\iu}{\mathrm{j}}}
140 \CustomizeMathJax{\newcommand{\ex}[1]{\mathrm{e}^{#1}}}
141 %
142 \CustomizeMathJax{\newcommand{\gammatens}{{}^{\mathrm{V}}\Gamma}}
143 \CustomizeMathJax{\newcommand{\gammacorr}{{}^{\mathrm{I}}\Gamma}}
144 \CustomizeMathJax{\newcommand{\gammatensin}[1]{{}^{\mathrm{V}}\Gamma_{\mathrm{#1}}}}
145 \CustomizeMathJax{\newcommand{\gammacorrin}[1]{{}^{\mathrm{I}}\Gamma_{\mathrm{#1}}}}
146 \CustomizeMathJax{\newcommand{\gammain}[1]{\Gamma_{\mathrm{#1}}}}
147 \CustomizeMathJax{\newcommand{\gammak}{{}^{\mathrm{k}}\Gamma}}
148 %
149 \CustomizeMathJax{\newcommand{\lbvt}{\lambda_0}}
150 \CustomizeMathJax{\newcommand{\lbg}{\lambda_g}}
151 \CustomizeMathJax{\newcommand{\lbgvt}{\lambda_{g_0}}}
152 %
153 \CustomizeMathJax{\newcommand{\potin}[1]{P_{\mathrm{#1}}}}
154 \CustomizeMathJax{\newcommand{\potdisp}[1][]{P_{\mathrm{disp}}^{#1}}}
```

```
155 \CustomizeMathJax{\newcommand{\potDC}[1][]{P_{\mathrm{DC}}^{#1}}}
156 \CustomizeMathJax{\newcommand{\potCC}[1][]{P_{\mathrm{CC}}^{#1}}}
157 \CustomizeMathJax{\newcommand{\potirr}[1][]{P_{\mathrm{irr}}^{#1}}}
158 \CustomizeMathJax{\newcommand{\potdiss}[1][]{P_{\mathrm{diss}}^{#1}}}
159 \CustomizeMathJax{\newcommand{\potinc}[1][]{P_{\mathrm{inc}}^{#1}}}
160 %
161 \CustomizeMathJax{\newcommand{\z}[1]{Z_{\mathrm{#1}}}}
162 \CustomizeMathJax{\newcommand{\znorm}[1]{z_{\mathrm{#1}}}}
163 \CustomizeMathJax{\newcommand{\y}[1]{Y_{\mathrm{#1}}}}
164 \CustomizeMathJax{\newcommand{\ynorm}[1]{y_{\mathrm{#1}}}}
165 \CustomizeMathJax{\newcommand{\zinf}[1][]{Z_{\infty#1}}}
166 \CustomizeMathJax{\newcommand{\zinfn}[1]{\zinf[#1]}}
167 \CustomizeMathJax{\newcommand{\yinf}[1][]{Y_{\infty#1}}}
168 \CustomizeMathJax{\newcommand{\yinfn}[1]{\yinf[#1]}}
169 \CustomizeMathJax{\newcommand{\zvt}{Z_0}}
170 \CustomizeMathJax{\newcommand{\yvt}{Y_0}}
171 %
172 \CustomizeMathJax{\newcommand{\campoe}{\underline{\mathcal{E}}(\underline{r},t)}}
173 \CustomizeMathJax{\newcommand{\campoefas}{\underline{E}(\underline{r})}}
174 \CustomizeMathJax{\newcommand{\campoh}{\underline{\mathcal{H}}(\underline{r},t)}}
175 \CustomizeMathJax{\newcommand{\campohfas}{\underline{H}(\underline{r})}}
176 %
177 \CustomizeMathJax{\newcommand{\signt}[1]{{#1}(t)}}
178 \CustomizeMathJax{\newcommand{\signf}[1]{{#1}(f)}}
179 \CustomizeMathJax{\newcommand{\signn}[1]{{#1}(n)}}
180 \CustomizeMathJax{\newcommand{\signz}[1]{{#1}(z)}}
181 %
182 \CustomizeMathJax{\newcommand{\prob}[1]{\mathcal{P}\left(#1\right)}}
183 \CustomizeMathJax{\newcommand{\valatt}[1]{\mathbb{E}\left[#1\right]}}
184 \CustomizeMathJax{\newcommand{\var}[1]{\mathrm{Var}\left[#1\right]}}
185 \CustomizeMathJax{\newcommand{\comma}{\, , \,}}
186 \CustomizeMathJax{\newcommand{\dato}{\,|\,}}
187 %
188 \CustomizeMathJax{\let\bfRe\partereale}
189 \CustomizeMathJax{\let\bfIm\parteimm}
190 \CustomizeMathJax{\let\noisevar\varianzarumore}
191 % \CustomizeMathJax{\let\exerend\finees}
192 \CustomizeMathJax{\let\Spimplies\frecciadex}
193 \CustomizeMathJax{\let\Downimplies\frecciadown}
194 \CustomizeMathJax{\let\unitvec\versore}
195 \CustomizeMathJax{\let\vector\vettore}
196 \CustomizeMathJax{\let\cosine\coseno}
197 \CustomizeMathJax{\let\sine\seno}
198 \CustomizeMathJax{\let\energy\energia}
199 \CustomizeMathJax{\let\Abs\modulo}
200 \CustomizeMathJax{\let\AbsPow\moduloexp}
201 \CustomizeMathJax{\let\Max\massimo}
202 \CustomizeMathJax{\let\Min\minimo}
203 \CustomizeMathJax{\let\clight\valc}
204 \CustomizeMathJax{\let\Log\loga}
205 \CustomizeMathJax{\let\analytic\analitic}
206 \CustomizeMathJax{\let\infint\intinf}
207 \CustomizeMathJax{\let\deltaimp\deltain}
208 \CustomizeMathJax{\let\Vgamma\gammatens}
209 \CustomizeMathJax{\let\Cgamma\gammacorr}
210 \CustomizeMathJax{\let\Vgammain\gammatensin}
211 \CustomizeMathJax{\let\Cgammain\gammacorrin}
212 \CustomizeMathJax{\let\Kgamma\gammak}
213 \CustomizeMathJax{\let\powerin\potin}
214 \CustomizeMathJax{\let\availpow\potdisp}
```

```
215 \CustomizeMathJax{\let\irrpow\potirr}
216 \CustomizeMathJax{\let\disspow\potdiss}
217 \CustomizeMathJax{\let\incpow\potinc}
218 \CustomizeMathJax{\let\potalim\potCC}
219 \CustomizeMathJax{\let\potDC\potCC}
220 \CustomizeMathJax{\let\Efield\campoe}
221 \CustomizeMathJax{\let\Hfield\campoh}
222 \CustomizeMathJax{\let\phasorEfield\campoefas}
223 \CustomizeMathJax{\let\phasorHfiled\campohfas}
224 \CustomizeMathJax{\let\given\dato}
225 \CustomizeMathJax{\let\expval\valatt}
226 \CustomizeMathJax{\let\rmexp\ex}
227 \end{warpMathJax}
```

File 133   `lwarp-enotez.sty`

§ 245   Package   **enotez**

(*Emulates or patches code by* Clemens Niederberger.)

enotez (*Pkg*)   enotez is patched for use by lwarp.

for HTML output: 1 `\LWR@ProvidesPackagePass{enotez}[2020/12/13]`

Hyperref is emulated by lwarp, so it is forced on for enotez:

```
2 \ExplSyntaxOn
3 \AtBeginDocument{
4     \bool_set_true:N \l__enotez_hyperref_bool
5     \bool_set_true:N \l__enotez_hyperfootnotes_bool
6 }
```

Do not move or \hbox the \hypertarget:

```
7 % typeset the actual mark:
8 % #1: id
9 % #2: mark
10 \VerifyCommand[lwarp][enotez]{\enotez_write_mark:nn}{61DA2A7B03A7D9F55E3E2E2D2498FB32}
11
12 \cs_gset_protected:Npn \enotez_write_mark:nn #1#2
13   {
14     \bool_if:NTF \l__enotez_hyperfootnotes_bool
15       {
16         \enotezwritemark { \hyperlink {enz.#1} { \enmarkstyle #2 } }
17         \bool_if:NT \l__enotez_hyperbackref_bool
18           {
19 %             \box_move_up:nn {1em} {
20 %                 \hbox:n {
21                     \hypertarget {enz.#1.backref} { }
22 %                 }
23 %             }
24           }
25       }
26     { \enotezwritemark { \enmarkstyle #2 } }
27   }
28 \cs_generate_variant:Nn \enotez_write_mark:nn {x}
```

Do not move or \hbox the \hypertarget:

```
29 \VerifyCommand[lwarp][enotez]{\enotez_write_list_number:n}{9793BEC2766E17864C6391209599DD84}
30
31 \cs_gset_protected:Npn \enotez_write_list_number:n #1
32   {
33     \bool_if:NT \l__enotez_hyperfootnotes_bool
34       {
35 %         \box_move_up:nn {1em} { \hbox:n {
36           \hypertarget {enz.#1} { }
37 %         } }
38       }
39     \tl_use:N \l__enotez_list_number_format_tl
40     \tl_if_eq:nxTF {a} { \prop_item:Nn \g__enotez_endnote_man_prop {#1} }
41       {
42         \bool_if:nTF
43           { \l__enotez_hyperfootnotes_bool && \l__enotez_hyperbackref_bool }
44           {
45             \exp_args:Nnx
46             \hyperlink {enz.#1.backref}
47               { \exp_not:V \l__enotez_endnote_mark_tl }
48           }
49           { \prop_item:Nn \g__enotez_endnote_mark_prop {#1} }
50       }
51       {
52         \bool_if:nTF
53           { \l__enotez_hyperfootnotes_bool && \l__enotez_hyperbackref_bool }
54           {
55             \exp_args:Nnx
56             \hyperlink {enz.#1.backref}
57               { \exp_not:V \l__enotez_endnote_mark_tl }
58           }
59           { \tl_use:N \l__enotez_endnote_mark_tl }
60       }
61   }
```

Do not move the label to the left:

```
62 \DeclareTemplateCode {enotez-list} {paragraph} {1}
63   {
64     heading       = \enotez_list_heading:n            ,
65     format        = \l__enotez_list_format_tl         ,
66     number        = \enotez_list_number:n             ,
67     number-format = \l__enotez_list_number_format_tl  ,
68     notes-sep     = \l__enotez_list_notes_sep_dim
69   }
70   {
71     \AssignTemplateKeys
72     \enotez_set_totoc:
73     \enotez_list_heading:n { \l__enotez_list_name_tl }
74     \enotez_list_preamble:
75     \enotez_build_print_list:nnnn {#1}
76       {}
77       {
78         \par\noindent
79         \group_begin:
80           \tl_use:N \l__enotez_list_format_tl
81 %           \hbox_overlap_left:n
82 %             {
83               \enotez_list_number:n
```

```
84                { \enotez_write_list_number:n {##1} }
85              \tl_use:N \c_space_tl
86 %            }
87       % \cs_set:cpn {@currentlabel}
88       %   { \p@endnote \l__enotez_endnote_mark_tl }
89         \tl_use:N \g__enotez_endnote_text_tl
90         \par
91         \dim_compare:nT { \l__enotez_list_notes_sep_dim != 0pt }
92           { \addvspace { \l__enotez_list_notes_sep_dim } }
93       \group_end:
94     }
95     {}
96   \enotez_list_postamble:
97 }
98
99 \ExplSyntaxOff
```

For MATHJAX:

```
100 \begin{warpMathJax}
101 \def\endnotename{endnote}
102 \appto\LWR@syncnotenumbers{\LWR@synconenotenumber{LWRendnote}{\theendnote}}
103 \appto\LWR@syncnotenames{\LWR@synconenotename{LWRendnote}{\endnotename}}
104 \CustomizeMathJax{\def\LWRendnote{1}}
105 \CustomizeMathJax{\newcommand{\endnote}[2][\LWRendnote]{{}^{\mathrm{#1}}}}
106 \CustomizeMathJax{\newcommand{\endnotemark}[1][\LWRendnote]{{}^{\mathrm{#1}}}}
107 \end{warpMathJax}
```

File 134 **`lwarp-enumerate.sty`**

§ 246   Package   # enumerate

enumerate (*Pkg*)   enumerate is supported with no changes.

This package is only required because it was used in the past to drop and then emulate the package. It cannot be removed because an older version which dropped the package may still remain, for example in a local vs. distribution directory, but it is now supported directly by lwarp and thus must no longer be dropped.

for HTML output: 1 \LWR@ProvidesPackagePass{enumerate}[2015/07/23]

File 135   **`lwarp-enumitem.sty`**

§ 247   Package   # enumitem

(*Emulates or patches code by* JAVIER BEZOS.)

enumitem (*Pkg*)   enumitem is supported with minor adjustments.

for HTML output: 1 \LWR@ProvidesPackagePass{enumitem}[2018/11/30]

\newlist {⟨*name*⟩} {⟨*type*⟩} {⟨*maxdepth*⟩}
\renewlist {⟨*name*⟩} {⟨*type*⟩} {⟨*maxdepth*⟩}

For enumitem lists, new lists must have the start and end actions assigned to the new environment. Renewed lists already have their actions assigned, and thus need no changes.

```
2 \let\LWR@enumitem@orignewlist\newlist
3
4 \renewcommand*{\newlist}[3]{%
5 \LWR@enumitem@orignewlist{#1}{#2}{#3}%
6 \AtBeginEnvironment{#1}{\@nameuse{LWR@#2start}}%
7 \AtEndEnvironment{#1}{\@nameuse{LWR@#2end}}%
8 }
9
10 \def\DrawEnumitemLabel{}
```

File 136 **`lwarp-epigraph.sty`**

§ 248   Package   # epigraph

(*Emulates or patches code by* PETER WILSON.)

epigraph (*Pkg*)   epigraph is emulated for HTML, and used as-is for print output.

Use css to format epigraphs.

for HTML output: `1 \LWR@ProvidesPackagePass{epigraph}[2020/01/02]`

```
2 \DeclareDocumentCommand{\LWR@HTML@qitem}{m m}
3 {%
4     \begin{BlockClass}{qitem}%
5     #1%
6     \LWR@stoppars%
7     \ifbool{FormatWP}%
8         {\begin{BlockClass}[border-top:1px solid gray]{epigraphsource}}%
9         {\begin{BlockClass}{epigraphsource}}%
10    #2%
11    \end{BlockClass}%
12    \end{BlockClass}%
13 }
14 \LWR@formatted{qitem}
```

epigraph: Added ARIA role.

```
15 \DeclareDocumentCommand{\LWR@HTML@epigraph}{m m}
16 {%
17    \begin{LWR@BlockClassWP}{\LWR@print@mbox{text-align:right}}{}(note){epigraph}%
18    \qitem{#1}{#2}%
19    \end{LWR@BlockClassWP}%
20 }
21 \LWR@formatted{epigraph}
22
23 \DeclareDocumentEnvironment{LWR@HTML@epigraphs}{}
24    {\LWR@BlockClassWP{\LWR@print@mbox{text-align:right}}{}(note){epigraph}}%
25    {\endLWR@BlockClassWP}
26 \LWR@formattedenv{epigraphs}
```

The following cannot be used in print mode while generating HTML:

```
27 \renewcommand{\epigraphhead}[2][0]{#2}
28 \renewcommand{\dropchapter}[1]{}
29 \renewcommand*{\undodrop}{}
```

File 137   `lwarp-epsf.sty`

§ 249   Package   **epsf**

(*Emulates or patches code by* Том Rокіскі.)

epsf (*Pkg*)   epsf is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{epsf}% not date given

```
2 \xpretocmd{\epsfsetgraph}
3     {\begin{lateximage}}
4     {}
5     {\LWR@patcherror{lwarp-epsf}{epsfsetgraph-begin}}
6
7 \xapptocmd{\epsfsetgraph}
8     {\end{lateximage}}
9     {}
10     {\LWR@patcherror{lwarp-epsf}{epsfsetgraph-end}}
```

File 138   `lwarp-epsfig.sty`

§ 250   Package   **epsfig**

epsfig (*Pkg*)   epsfig is emulated for use by lwarp.

⚠   Only the LATEX2e syntax is emulated.

for HTML output: 1 \LWR@ProvidesPackagePass{epsfig}[2017/06/25]

A few additional keys to capture the filename:

```
2 \RequirePackage{graphics}
3
4 \define@key{igraph}{file}{%
5     \xdef\LWR@epsfig@filename{#1}%
6 }
7
8 \define@key{igraph}{figure}{%
9     \xdef\LWR@epsfig@filename{#1}%
10 }
11
12 \define@key{igraph}{prolog}{}
13
14 \define@key{igraph}{silent}[]{}
```

The captured filename is used as the argument to \includegraphics:

```
15 \newcommand{\LWR@HTML@epsfig}[1]{\includegraphics[#1]{\LWR@epsfig@filename}}
16 \LWR@formatted{epsfig}
```

```
17
18 \newcommand{\LWR@HTML@psfig}[1]{\includegraphics[#1]{\LWR@epsfig@filename}}
19 \LWR@formatted{psfig}
```

File 139   **lwarp-epstopdf.sty**

§ 251   Package   # epstopdf

epstopdf (*Pkg*)   Previous versions of lwarp had a nullfied version, but now epstopdf-base is supported. lwarp-epstopdf becomes a placeholder to overwrite previous versions.

See package epstopdf-base for details.

for HTML output: `1 \LWR@ProvidesPackagePass{epstopdf}[2020-01-24]`

File 140   **lwarp-epstopdf-base.sty**

§ 252   Package   # epstopdf-base

epstopdf-base (*Pkg*)

⚠   convert to .svg   Images with an .eps extension will be converted to .pdf. The HTML output uses the .svg version, so use

Enter ⇒   **lwarpmk pdftosvg <listofPDFfiles>**

to generate .svg versions.

for HTML output: `1 \LWR@ProvidesPackagePass{epstopdf-base}[2020-01-24]`

Redefine to remember the image filename, replacing .pdf with .svg. Use the epstopdf print version inside a lateximage.

```
2 \newcommand*{\LWR@HTML@ETE@OrgGin@setfile}[3]{%
3     \edef\LWR@tempone{#3}%
4     \StrSubstitute{\LWR@tempone}{.pdf}{.svg}[\LWR@tempone]%
5     \StrSubstitute{\LWR@tempone}{.PDF}{.SVG}[\LWR@tempone]%
6     \xdef\LWR@parsedfilename{\LWR@tempone}%
7 }
8
9 \LWR@formatted{ETE@OrgGin@setfile}
```

\includegraphics in HTML mode redefines \Gin@setfile to be \LWR@HTML@Gin@setfile, which is now redirected to epstopdf's version:

```
10 \renewcommand*{\LWR@HTML@Gin@setfile}[3]{%
11     \ETE@Gin@setfile{#1}{#2}{#3}%
12 }
```

Allow .eps images to be found if a suffix is not provided:

```
13 \AtBeginDocument{
14 \DeclareGraphicsExtensions{%
15     .eps,.EPS,.svg,.SVG,.gif,.GIF,.png,.PNG,.jpg,.JPG,.jpeg,.JPEG%
```

```
16 }
17 \DeclareGraphicsRule{.svg}{svg}{.svg}{}
18 \DeclareGraphicsRule{.SVG}{svg}{.SVG}{}
19 }
```

Likewise when inside a lateximage:

```
20 \appto\LWR@restoreorigformatting{%
21 \DeclareGraphicsExtensions{%
22     .eps,.EPS,.pdf,.PDF,.gif,.GIF,.png,.PNG,.jpg,.JPG,.jpeg,.JPEG%
23 }%
24 }
```

File 141   `lwarp-eqlist.sty`

§ 253   Package   # eqlist

eqlist (*Pkg*)   eqlist is emulated.

for HTML output: 1 `\LWR@ProvidesPackageDrop{eqlist}[2002/08/15]`

```
2 \newenvironment{eqlist}[1][]{\description}{\enddescription}
3 \newenvironment{eqlist*}[1][]{\description}{\enddescription}
4 \newenvironment{Eqlist}[2][]{\description}{\enddescription}
5 \newenvironment{Eqlist*}[2][]{\description}{\enddescription}
6 \newcommand*{\longitem}[1][]{\item[#1]}
7 \newcommand*{\eqlistinit}{}
8 \newcommand*{\eqliststarinit}{}
9 \newcommand*{\eqlistinitpar}{}
10 \def\eqlistlabel#1{#1}
11 \newcommand{\eqlistauto}[1]{}
12 \newcommand{\eqlistnoauto}{}
```

File 142   `lwarp-eqparbox.sty`

§ 254   Package   # eqparbox

(*Emulates or patches code by* SCOTT PAKIN.)

eqparbox (*Pkg*)   eqparbox is patched for use by lwarp.

for HTML output: 1 `\LWR@ProvidesPackagePass{eqparbox}[2017/09/03]`

```
2 \NewDocumentCommand{\LWR@HTML@eqparbox}{O{t} O{} O{t} m +m}{%
3     {%
4         \minipagefullwidth%
5         \parbox[#1][#2][#3]{\linewidth}{#5}%
6     }%
7 }
8 \LWR@formatted{eqparbox}
9
10 \NewDocumentCommand{\LWR@HTML@eqmakebox}{o o m}{%
11     \makebox[#2]{#3}%
12 }
```

```
13 \LWR@formatted{eqmakebox}
14
15 \NewDocumentCommand{\LWR@HTML@eqframebox}{o o m}{%
16     \framebox[#2]{#3}%
17 }
18 \LWR@formatted{eqframebox}
19
20 \NewDocumentEnvironment{LWR@HTML@eqminipage}{O{t} O{} O{t} m}
21 {%
22     \begingroup%
23     \minipagefullwidth%
24     \minipage[#1][#2][#3]{\linewidth}%
25 }%
26 {%
27     \endminipage%
28     \endgroup%
29 }
30
31 \newcommand*{\LWR@HTML@eqboxwidth}[1]{.25\linewidth}
32 \LWR@formatted{eqboxwidth}
33
34 \newcommand*{\LWR@HTML@eqsetminwidth}[2]{}
35 \newcommand*{\LWR@HTML@eqsetmaxwidth}[2]{}
36
37 \newcommand*{\LWR@HTML@eqsetminwidthto}[2]{}
38 \newcommand*{\LWR@HTML@eqsetmaxwidthto}[2]{}
```

File 143 **`lwarp-errata.sty`**

§ 255   Package **errata**

(*Emulates or patches code by* MICHAEL KOHLHASE.)

errata (*Pkg*)   errata is patched for use by lwarp.

This is for v0.3 of errata. A newer version of errata with more features is under development, at which time the lwarp version will have to be updated.

for HTML output: Macros are being defined with the math dollar, so enable the HTML version during package loading:

```
1 \StartDefiningMath
```

Now load the package:

```
2 \LWR@ProvidesPackagePass{errata}[2006/11/12]
```

Patches for dynamic inline math:

```
3 \VerifyCommand[lwarp][errata]{\erratumAdd}{777B919444DA9C70140B71E0C9EDEEBF}
4
5 \xpatchcmd{\erratumAdd}
6     {$_a^{\arabic{erratum}}$}
7 %     {\inlinemathother$_a^{\arabic{erratum}}$\inlinemathnormal}
8     {\textsubscript{a}\textsuperscript{\arabic{erratum}}}
9     {}
10     {\LWR@patcherror{erratum}{erratumAdd}}
11 }
```

```
11
12 \VerifyCommand[lwarp][errata]{\erratumDelete}{057CF8E4B6A0DBECF95C009E9DC44FBA}
13
14 \xpatchcmd{\erratumDelete}
15     {$_d^{\arabic{erratum}}$}
16 %     {\inlinemathother$_d^{\arabic{erratum}}$\inlinemathnormal}
17     {\textsubscript{d}\textsuperscript{\arabic{erratum}}}
18     {}
19     {\LWR@patcherror{erratum}{erratumDelete}}
20
21 \VerifyCommand[lwarp][errata]{\erratumReplace}{0E24E5FE5415E6038089ABF21C6933D7}
22
23 \xpatchcmd{\erratumReplace}
24     {$_r^{\arabic{erratum}}$}
25 %     {\inlinemathother$_r^{\arabic{erratum}}$\inlinemathnormal}
26     {\textsubscript{r}\textsuperscript{\arabic{erratum}}}
27     {}
28     {\LWR@patcherror{erratum}{erratumReplace}}
29
30 \VerifyCommand[lwarp][errata]{\erratum}{A430F080689BC6FF47E7C905800D2028}
31
32 \xpatchcmd{\erratum}
33     {$_a$}
34 %     {\inlinemathother$_a$\inlinemathnormal}
35     {\textsubscript{a}}
36     {}
37     {\LWR@patcherror{erratum}{erratumDelete}}
38 \xpatchcmd{\erratum}
39     {$_d^{\@thefnmark}$}
40 %     {\inlinemathother$_d^{\@thefnmark}$\inlinemathnormal}
41     {\textsubscript{d}\textsuperscript{\@thefnmark}}
42     {}
43     {\LWR@patcherror{erratum}{eDelete}}
44 \xpatchcmd{\erratum}
45     {$_r^{\@thefnmark}$}
46 %     {\inlinemathother$_r^{\@thefnmark}$\inlinemathnormal}
47     {\textsubscript{r}\textsuperscript{\@thefnmark}}
48     {}
49     {\LWR@patcherror{erratum}{eReplace}}
```

Finish the current page's errata before closing and reloading the list:

```
50 \preto\PrintErrata{\LWR@maybe@orignewpage}
```

No longer defining math macros with the HTML $:

```
51 \StopDefiningMath
```

File 144 **`lwarp-eso-pic.sty`**

§256   Package **eso-pic**

(*Emulates or patches code by* ROLF NIEPRASCHK.)

eso-pic (*Pkg*)   eso-pic is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{eso-pic}[2018/04/12]

```
 2 \newcommand*{\LenToUnit}{}
 3 \newcommand{\AtPageUpperLeft}[1]{}
 4 \newcommand{\AtPageLowerLeft}[1]{}
 5 \newcommand{\AtPageCenter}[1]{}
 6 \newcommand{\AtStockLowerLeft}[1]{}
 7 \newcommand{\AtStockUpperLeft}[1]{}
 8 \newcommand{\AtStockCenter}[1]{}
 9 \newcommand{\AtTextUpperLeft}[1]{}
10 \newcommand{\AtTextLowerLeft}[1]{}
11 \newcommand{\AtTextCenter}[1]{}
12 \NewDocumentCommand{\AddToShipoutPictureBG}{s +m}{}
```

```
13 \newcommand{\AddToShipoutPicture}{\AddToShipoutPictureBG}
14 \NewDocumentCommand{\AddToShipoutPictureFG}{s +m}{}
15 \newcommand*{\ClearShipoutPictureBG}{}
16 \newcommand*{\ClearShipoutPicture}{}
17 \newcommand*{\ClearShipoutPictureFG}{}
18 \newcommand{\gridSetup}[6][]{}
```

File 145   **`lwarp-esvect.sty`**

§ 257   Package   **esvect**

(*Emulates or patches code by* EDDIE SAUDRAIS.)

esvect (*Pkg*)   esvect is used as-is for SVG math, and emulated for MATHJAX.

for HTML output:
```
1 \LWR@ProvidesPackagePass{esvect}% no date given
```

```
2 \begin{warpMathJax}
3 \CustomizeMathJax{\newcommand{\LWResvectvv}[1]{\overrightarrow{#1}}}
4 \CustomizeMathJax{\newcommand{\LWResvectvvstar}[2]{\overrightarrow{#1}\!_{#2}}}
5 \CustomizeMathJax{\newcommand{\vv}{\ifstar\LWResvectvvstar\LWResvectvv}}
6 \end{warpMathJax}
```

File 146   **`lwarp-etoc.sty`**

§ 258   Package   **etoc**

etoc (*Pkg*)   etoc is ignored. All commands are nullified.

⚠ \tableofcontents with \ref   The etoc package uses a non-standard syntax which looks ahead after a \tableofcontents for a following \ref. These \refs appear in the HTML result unless they are removed. Where a \tableofcontents is followed by \ref, and perhaps also \label as well, enclose all of them inside \warpprintonly:

    \warpprintonly{\tableofcontents   \ref{toc:abc}
    \label{toc:def}}

or place all code related to a local \tableofcontents inside a warpprint environment.

⚠ home page   Be sure to keep the initial \tableofcontents on the home page, perhaps in its own \warpHTMLonly macro or warpHTML environment.

for HTML output: 1 `\LWR@ProvidesPackageDrop{etoc}[2019/11/17]`

2 `\def\etocsetlevel#1#2{}`
3 `\def\etocskipfirstprefix{}`
4 `\let\etocthename    \@empty`
5 `\let\etocthenumber \@empty`
6 `\let\etocthepage    \@empty`
7 `\let\etocthelinkedname    \@empty`
8 `\let\etocthelinkednumber \@empty`
9 `\let\etocthelinkedpage    \@empty`
10 `\let\etocthelink    \@firstofone % prior to 1.08j its was \let to \@empty`
11 `\DeclareRobustCommand*{\etocname}  {}`
12 `\DeclareRobustCommand*{\etocnumber}{}`
13 `\DeclareRobustCommand*{\etocpage}  {}`
14 `\DeclareRobustCommand*{\etoclink}  {\@firstofone}`
15 `\DeclareRobustCommand*{\etocifnumbered}{\@firstoftwo}`
16 `\DeclareRobustCommand*{\etociffirst}{\@firstoftwo}`
17 `\DeclareRobustCommand*\etocifwasempty{\@firstoftwo}`
18 `\let\etocaftertitlehook    \@empty`
19 `\let\etocaftercontentshook \@empty`
20 `\def\etoctableofcontents{}`
21 `\newcommand*\localtableofcontents{}`
22 `\newcommand*\localtableofcontentswithrelativedepth[1]{}`
23 `\newcommand\etocsettocstyle[2]{}`
24 `\long\def\etocsetstyle#1#2#3#4#5{}`
25 `\def\etocfontminustwo {\normalfont \LARGE \bfseries}`
26 `\def\etocfontminusone {\normalfont \large \bfseries}`
27 `\def\etocfontzero     {\normalfont \large \bfseries}`
28 `\def\etocfontone      {\normalfont \normalsize \bfseries}`
29 `\def\etocfonttwo      {\normalfont \normalsize}`
30 `\def\etocfontthree    {\normalfont \footnotesize}`
31 `\def\etocsepminustwo {4ex \@plus .5ex \@minus .5ex}`
32 `\def\etocsepminusone {4ex \@plus .5ex \@minus .5ex}`
33 `\def\etocsepzero     {2.5ex \@plus .4ex \@minus .4ex}`
34 `\def\etocsepone      {1.5ex \@plus .3ex \@minus .3ex}`
35 `\def\etocseptwo      {.5ex \@plus .1ex \@minus .1ex}`
36 `\def\etocsepthree    {.25ex \@plus .05ex \@minus .05ex}`
37 `\def\etocbaselinespreadminustwo {1}`
38 `\def\etocbaselinespreadminusone {1}`
39 `\def\etocbaselinespreadzero     {1}`
40 `\def\etocbaselinespreadone      {1}`
41 `\def\etocbaselinespreadtwo      {1}`
42 `\def\etocbaselinespreadthree    {.9}`
43 `\def\etocminustwoleftmargin  {1.5em plus 0.5fil}`
44 `\def\etocminustworightmargin {1.5em plus -0.5fil}`
45 `\def\etocminusoneleftmargin  {1em}`
46 `\def\etocminusonerightmargin {1em}`
47 `\def\etoctoclineleaders`
48 `      {\hbox{\normalfont\normalsize\hb@xt@2ex {\hss.\hss}}}`
49 `\def\etocabbrevpagename {p.~}`
50 `\def\etocpartname        {Part}% modified 1.08b`
51 `\def\etocbookname        {Book}`
52 `\def\etocdefaultlines{}`
53 `\def\etocabovetocskip{3.5ex \@plus 1ex \@minus .2ex}`
54 `\def\etocbelowtocskip{3.5ex \@plus 1ex \@minus .2ex}`
55 `\def\etoccolumnsep{2em}`
56 `\def\etocmulticolsep{0ex}`
57 `\def\etocmulticolpretolerance{-1}`
58 `\def\etocmulticoltolerance{200}`
59 `\def\etocdefaultnbcol{2}`

```
 60 \def\etocinnertopsep{2ex}
 61 \newcommand\etocmulticolstyle[2][]{}
 62 \def\etocinnerbottomsep{3.5ex}
 63 \def\etocinnerleftsep{2em}
 64 \def\etocinnerrightsep{2em}
 65 \def\etoctoprule{\hrule}
 66 \def\etocleftrule{\vrule}
 67 \def\etocrightrule{\vrule}
 68 \def\etocbottomrule{\hrule}
 69 \def\etoctoprulecolorcmd{\relax}
 70 \def\etocbottomrulecolorcmd{\relax}
 71 \def\etocleftrulecolorcmd{\relax}
 72 \def\etocrightrulecolorcmd{\relax}
 73 \newcommand*\etocruledstyle[2][]{}
 74 \def\etocframedmphook{\relax}
 75 \long\def\etocbkgcolorcmd{\relax}
 76 \newcommand*\etocframedstyle[2][]{}
 77 \def\etocmulticol{}
 78 \def\etocruled{}
 79 \def\etocframed{}
 80 \def\etoclocalmulticol{}
 81 \def\etoclocalruled{}
 82 \def\etoclocalframed{}
 83 \def\etocarticlestyle{}
 84 \def\etocarticlestylenomarks{}
 85 \def\etocbookstyle{}
 86 \def\etocbookstylenomarks{}
 87 \let\etocreportstyle\etocbookstyle
 88 \let\etocreportstylenomarks\etocbookstylenomarks
 89 \def\etocmemoirtoctotocfmt #1#2{}
 90 \def\etocmemoirstyle{}
 91 \def\etocscrartclstyle{}
 92 \let\etocscrbookstyle\etocscrartclstyle
 93 \let\etocscrreprtstyle\etocscrartclstyle
 94 \def\etocstandarddisplaystyle{\etocarticlestyle}
 95 \newcommand*\etocmarkboth[1]{}
 96 \newcommand*\etocmarkbothnouc[1]{}
 97 \newcommand\etoctocstyle[3][section]{}
 98 \newcommand\etoctocstylewithmarks[4][section]{}
 99 \newcommand\etoctocstylewithmarksnouc[4][section]{}
100 \def\etocignoretoctocdepth{}
101 \def\etocsettocdepth[1]{}
102 \def\etocdepthtag    #1#{\Etoc@depthtag }
103 \def\Etoc@depthtag  #1{}
104 \def\etocignoredepthtags {}
105 \def\etocobeydepthtags    {}
106 \def\etocsettagdepth #1#2{}
107 \def\invisibletableofcontents {}
108 \def\invisiblelocaltableofcontents{}
109 \def\etocsetnexttocdepth #1{}
110 \def\etocsetlocaltop #1#{\Etoc@set@localtop}
111 \def\Etoc@set@localtop #1{}
112 \def\etocstandardlines {}
113 \def\etoctoclines       {}
114 \let\etocaftertochook    \@empty
115 \let\etocbeforetitlehook \@empty
116 \appto\tableofcontents{\def\tableofcontents{}}
```

File 147   `lwarp-eurosym.sty`

§ 259   Package   # eurosym

(*Emulates or patches code by* Henrik Theiling.)

eurosym (*Pkg*)   eurosym is patched for use by lwarp.

for HTML output: 1 `\LWR@ProvidesPackagePass{eurosym}[1998/08/06]`

2 `\renewrobustcmd\officialeuro{\HTMLentity{euro}}`
3 `\let\geneuro\officialeuro`
4 `\let\geneuronarrow\officialeuro`
5 `\let\geneurowide\officialeuro`
6 `\let\euro\officialeuro`
7 `\renewrobustcmd\eurobars{}`
8 `\renewrobustcmd\eurobarsnarrow{}`
9 `\renewrobustcmd\eurobarswide{}`

File 148   `lwarp-everypage.sty`

§ 260   Package   # everypage

(*Emulates or patches code by* Sergio Callegari.)

everypage (*Pkg*)   everypage is ignored.

for HTML output: 1 `\LWR@ProvidesPackageDrop{everypage}[2007/06/20]`

2 `\newcommand*{\AddEverypageHook}[1]{}`
3 `\newcommand*{\AddThispageHook}[1]{}`

File 149   `lwarp-everyshi.sty`

§ 261   Package   # everyshi

(*Emulates or patches code by* Martin Schröder.)

everyshi (*Pkg*)   ignored.

for HTML output: Discard all options for lwarp-everyshi:

1 `\LWR@ProvidesPackageDrop{everyshi}[2001/05/15]`

2 `\let\EveryShipout\relax`
3 `\newcommand*{\EveryShipout}[1]{}`
4
5 `\let\AtNextShipout\relax`
6 `\newcommand*{\AtNextShipout}[1]{}`

File 150   `lwarp-extarrows.sty`

§ 262   Package   **extarrows**

(*Emulates or patches code by* HUYNH KY ANH.)

extarrows (*Pkg*)   extarrows is used as-is for SVG math, and emulted for MATHJAX.

for HTML output:
```
1 \LWR@ProvidesPackagePass{extarrows}[2008/05/15]
```

```
2 \begin{warpMathJax}
3 \CustomizeMathJax{\Newextarrow\xLongleftarrow{10,10}{0x21D0}}
4 \CustomizeMathJax{\Newextarrow\xLongrightarrow{10,10}{0x21D2}}
5 \CustomizeMathJax{\Newextarrow\xLongleftrightarrow{10,10}{0x21D4}}
6 \CustomizeMathJax{\Newextarrow\xLeftrightarrow{10,10}{0x21D4}}
7 \CustomizeMathJax{\Newextarrow\xlongleftrightarrow{10,10}{0x2194}}
8 \CustomizeMathJax{\Newextarrow\xleftrightarrow{10,10}{0x2194}}
9 \CustomizeMathJax{\let\xlongleftarrow\xleftarrow}
10 \CustomizeMathJax{\let\xlongrightarrow\xrightarrow}
11 \end{warpMathJax}
```

File 151   `lwarp-extramarks.sty`

§ 263   Package   **extramarks**

(*Emulates or patches code by* PIET VAN OOSTRUM.)

extramarks (*Pkg*)   extramarks is ignored.

for HTML output: Discard all options for lwarp-extramarks:

```
1 \LWR@ProvidesPackageDrop{extramarks}[2025/01/07]
```

```
2 \newcommand*{\extramarks}[2]{}
3 \newcommand\extramarksleft[1]{}
4 \newcommand\extramarksright[1]{}
5 \newcommand*{\firstleftxmark}{}
6 \newcommand*{\lastleftxmark}{}
7 \newcommand*{\firstrightxmark}{}
8 \newcommand*{\lastrightxmark}{}
9 \newcommand*{\firstxmark}{}
10 \newcommand*{\lastxmark}{}
11 \newcommand*{\topxmark}{}
12 \newcommand*{\topleftxmark}{}
13 \newcommand*{\toprightxmark}{}
14 \newcommand*{\firstleftmark}{}
15 \newcommand*{\lastrightmark}{}
16 \newcommand*{\firstrightmark}{}
17 \newcommand*{\lastleftmark}{}
```

File 152 **`lwarp-fancybox.sty`**

§ 264   Package   **fancybox**

(*Emulates or patches code by* TIMOTHY VAN ZANDT.)

fancybox (*Pkg*)   **fancybox** is supported with some patches.

framed equation example   **fancybox**'s documentation has an example FramedEqn environment which combines math, \Sbox, a minipage, and an \fbox. This combination requires that the entire environment be enclosed inside a lateximage, which is done by adding \lateximage at the very start of FramedEqn's beginning code, and \endlateximage at the very end of the ending code. Unfortunately, the HTML alt attribute is not used here.

```
\newenvironmentFramedEqn
{
\lateximage% NEW
\setlength{\fboxsep}{15pt}
. . . }{. . .
\[\fbox{\TheSbox}\]
\endlateximage% NEW
}
```

framing alternatives   \fbox works with **fancybox**. Also see **lwarp**'s \fboxBlock macro and fminipage environment for alternatives to \fbox for framing environments.

framed table example   The **fancybox** documentation's example of a framed table using an \fbox containing a tabular does not work with **lwarp**, but the FramedTable environment does work if \fbox is replaced by \fboxBlock. This method does lose some HTML formatting. A better method is to enclose the table's contents inside a fminipage environment. The caption may be placed either inside or outside the fminipage:

```
\begin{table}
\begin{fminipage}{\linewidth}
\begin{tabular}{lr}
. . .
\end{tabular}
\end{fminipage}
\end{table}
```

⚠   framed verbatim   **lwarp** does not support the verbatim environment inside a span, box, or **fancybox**'s \Sbox, but a verbatim may be placed inside a fminipage. The **fancybox** documentation's example FramedVerb may be defined as:

```
\newenvironment{FramedVerb}[1] % width
{
  \VerbatimEnvironment
  \fminipage{#1}
  \beginVerbatim
}{
  \endVerbatim
  \endfminipage
}
```

framed \VerbBox   fancybox's \VerbBox may be used inside \fbox.

indented alignment   LVerbatim, \LVerbatimInput, and \LUseVerbatim indent with horizontal space which may not line up exactly with what *pdftotext* detects. Some lines may be off slightly in their left edge.

lwarp sanitizes HTML for fancybox verbatims, except for the contents of \VerbBox and any \verb inside.

fancybox, fancyvrb
⚠ \VerbatimFootnotes
⚠ sectioning or
displaymath

If using fancybox or fancyvrb with \VerbatimFootnotes, and using footnotes in a sectioning command or display math, use \footnotemark and \footnotetext:

```
\subsection[Subsection Name]
    {Subsection Name\protect\footnotemark}
\footnotetext{A footnote with \verb+verbtim+.}
```

and likewise for equations or display math.

```
1 \LWR@ProvidesPackagePass{fancybox}[2010/05/15]
```

After the preamble is loaded, after any patches to Verbatim:

```
2 \AfterEndPreamble{
3 \LWR@traceinfo{Patching fancybox.}
```

\VerbatimFootnotes   Patched to use the new version.

```
4 \def\VerbatimFootnotes{%
5 \let\@footnotetext\V@footnotetext%
6 \let\LWR@footnotetext\V@footnotetext% lwarp
7 }
```

\V@@footnotetext   Patches in a subset of lwarp's \LWR@footnotetext to the fancyvrb version of \V@@footnotetext.

```
8 \def\V@@footnotetext{%
9 \LWR@traceinfo{V@footnotetext}%
```

Place an autopage marker so that back references to citations inside a footnote will link closer to the footnote text, if possible.

```
10 \LWR@newautopagelabel{page}%
```

Take the current footnote box, then append:

```
11 \global\setbox\LWR@footnotebox=\vbox\bgroup%
```

Add to any current footnotes:

```
12     \unvbox\LWR@footnotebox%
```

Remember the footnote number for \ref:

```
13     \protected@edef\@currentlabel{%
14         \csname p@footnote\endcsname\@thefnmark%
15     }% @currentlabel
16 %     \ifdef{cref@currentlabel}{% cleveref
17 %%         \protected@edef\cref@currentlabel{%
18 %%             [footnote]%
19 %%             [\arabic{footnote}]%
20 %%             []%
```

```
21 %%              \csname p@footnote\endcsname\@thefnmark}%
22 %          \cref@constructprefix{footnote}{\cref@result}%
23 %          \@ifundefined{cref@footnote@alias}%
24 %            {\def\@tempa{footnote}}%
25 %            {\def\@tempa{\csname cref@footnote@alias\endcsname}}%
26 %          \protected@edef\cref@currentlabel{%
27 %            [\@tempa][\arabic{footnote}][\cref@result]%
28 %            \csname p@footnote\endcsname\csname thefootnote\endcsname}%
29 %    }{}%
```

Use HTML superscripts in the footnote even inside a lateximage:

```
30      \renewrobustcmd{\textsuperscript}[1]{\LWR@htmlspan{sup}{##1}}%
```

Use paragraph tags if in a tabular data cell or a lateximage:

```
31      \LWR@htmltagc{\LWR@tagregularparagraph}\LWR@orignewline%
```

Append the footnote to the list:

```
32      \@makefntext{}%
```

The footnote text will follow after \V@@@footnotetext has completed.

```
33      \bgroup%
34      \aftergroup\V@@@footnotetext%
```

Do not generate autopages inside the footnotes, since they are accumulated at the
moment before finally being used perhaps on a later page.

```
35      \let\LWR@newautopagelabel\LWR@null@newautopagelabel%

36      \ignorespaces%
37 }%
```

\V@@@footnotetext

```
38 \def\V@@@footnotetext{%
39      \LWR@orignobreakspace\LWR@orignewline%
40      \LWR@htmltagc{/\LWR@tagregularparagraph}\LWR@orignewline%
41      \strut\egroup%
42 }

43 }% AfterEndPreamble

44 \renewcommand*{\@shadowbox}[1]{%
45 \ifbool{FormatWP}%
46 {\InlineClass[border:1px solid black]{shadowbox}{#1}}%
47 {\InlineClass{shadowbox}{#1}}%
48 }

49

50 \renewcommand*{\@doublebox}[1]{%
51 \ifbool{FormatWP}%
52 {\InlineClass[border:1px double black]{doublebox}{#1}}%
53 {\InlineClass{doublebox}{#1}}%
54 }

55

56 \renewcommand*{\@ovalbox}[2]{%
57 \ifbool{FormatWP}%
58 {\InlineClass[border:1px solid black; border-radius:1ex]{ovalbox}{#2}}%
59 {%
60      \ifthenelse{\isequivalentto{#1}{\thinlines}}%
61          {\InlineClass{ovalbox}{#2}}%
62          {\InlineClass{Ovalbox}{#2}}%
```

```
63 }%
64 }
```

Convert minipages, parboxes, and lists into linear text using the LWR@nestspan
environment:

```
65 \let\LWR@origSbox\Sbox
66
67 \def\Sbox{\LWR@origSbox\LWR@nestspan}
68
69
70 \let\LWR@origendSbox\endSbox
71
72 \def\endSbox{\endLWR@nestspan\LWR@origendSbox}
```

Beqnarray is adapted for MATHJAX or enclosed inside a lateximage:

```
73 \RenewEnviron{Beqnarray}
74 {\LWR@eqnarrayfactor}
75
76 \csgpreto{Beqnarray*}{\boolfalse{LWR@numbereqnarray}}
```

\GenericCaption is enclosed in an HTML block:

```
77 \renewcommand{\GenericCaption}[1]{%
78     \LWR@figcaption%
79     \LWR@isolate{#1}%
80     \endLWR@figcaption%
81 }
```

Btrivlist is enclosed in an HTML block. This is a tabular, and does not use \item.

\trivlist   {⟨l/c/r⟩} [⟨t/c/b⟩]

```
82 \RenewDocumentEnvironment{Btrivlist}{m o}
83 {%
84     \LWR@stoppars%
85     \begin{BlockClass}{Btrivlist}%
86     \tabular{#1}%
87 }
88 {%
89     \endtabular%
90     \end{BlockClass}%
91     \LWR@startpars%
92 }
```

Btrivlist is also neutralized when used inside a span:

```
93 \AtBeginEnvironment{LWR@nestspan}{%
94     \RenewDocumentEnvironment{Btrivlist}{m o}{}{}%
95 }
```

lwarp's handling of \item is patched to accept fancybox's optional arguments:

```
96 \let\LWRFB@origitemizeitem\LWR@itemizeitem
97 \let\LWRFB@origdescitem\LWR@descitem
98
```

```
 99 \RenewDocumentCommand{\LWR@itemizeitem}{d()o}{%
100     \IfValueTF{#2}{%
101         \LWRFB@origitemizeitem[#2]%
102     }{%
103         \LWRFB@origitemizeitem%
104     }%
105 }
106
107 \RenewDocumentCommand{\LWR@descitem}{d()o}{%
108     \IfValueTF{#2}{%
109         \LWRFB@origdescitem[#2]~%
110     }{%
111         \LWRFB@origdescitem%
112     }%
113 }
114 \RenewDocumentCommand{\LWR@nestspanitem}{d()}{%
115     \if@newlist\else{
116         \LWR@htmltagc{br /}%
117         \LWR@orignewline%
118     }\fi%
119     \LWR@origitem%
120 }
```

The various boxed lists become regular lists:

```
121 \renewenvironment{Bitemize}[1][]
122     {%
123         \LWR@spanwarnformat{Bitemize}%
124         \booltrue{LWR@starting@fancybox}%
125         \begin{itemize}%
126         \boolfalse{LWR@starting@fancybox}%
127     }
128     {\end{itemize}}
129
130 \renewenvironment{Benumerate}[1][]
131     {%
132         \LWR@spanwarnformat{Benumerate}%
133         \booltrue{LWR@starting@fancybox}%
134         \begin{enumerate}%
135         \boolfalse{LWR@starting@fancybox}%
136     }
137     {\end{enumerate}}
138
139 \renewenvironment{Bdescription}[1][]
140     {%
141         \LWR@spanwarnformat{Bdescription}%
142         \booltrue{LWR@starting@fancybox}%
143         \begin{description}%
144         \boolfalse{LWR@starting@fancybox}%
145     }
146     {\end{description}}
```

\boxput simply prints one then the other argument, side-by-side instead of above and behind:

```
147 \RenewDocumentCommand{\boxput}{s d() m m}{%
148     \IfBooleanTF{#1}{#3\quad#4}{#4\quad#3}%
149 }
```

Neutralized commands:

```
150 \RenewDocumentCommand{\fancyput}{s d() m}{}
151 \RenewDocumentCommand{\thisfancyput}{s d() m}{}
152
153 \RenewDocumentCommand{\fancypage}{m m}{}
154 \RenewDocumentCommand{\thisfancypage}{m m}{}
155
156 \def\LandScape#1{}
157 \def\endLandScape{}
158 \def\@Landscape#1#2#3{}
159 \def\endLandscape{}
```

Low-level patches for `UseVerbatim` and friends:

```
160 \VerifyCommand[lwarp][fancybox]{\Verbatim@List}{A85522266594F8C0D846AEB1F72232FF}
161
162 \def\Verbatim@List#1{%
163   \if@minipage\else\vskip\parskip\fi
164   \leftskip\@totalleftmargin
165   \@rightskip\@flushglue \rightskip\@rightskip
166   \parindent\z@
167   \parskip\z@
168   \parfillskip\@flushglue
169   \hfuzz\VerbatimFuzz\relax
170   \@@par
171   \global\@inlabelfalse  %Prevents vspace from being inserted when
172   \Verbatim@Prep         %first line exceeds \hsize.
173 %  #1%
174     \expandafter\def\expandafter\tmpb\expandafter{#1}%  lwarp
175     \LWR@HTMLsanitize@tmpb%                             lwarp
176     \tmpb%                                              lwarp
177   \Verbatim@Par}%
178
179 \VerifyCommand[lwarp][fancybox]{\Verbatim@@Input}{3DCC957D04BC5060FF70DD0FF2928D55}
180
181 \def\Verbatim@@Input{%
182   \let\protect\noexpand
183   \LetLtxMacro\tmpb\The@Verbatim%   lwarp
184   \LWR@HTMLsanitize@tmpb%           lwarp
185   \edef\The@Verbatim{%
186     \noexpand\Every@VerbatimLine
187 %    \The@Verbatim
188     \tmpb%                          lwarp
189     \ifeof\Verbatim@Infile\else\noexpand\Verbatim@Par\fi}%
190   \let\protect\relax
191   \The@Verbatim
192   \ifeof\Verbatim@Infile\else
193     \let\The@Verbatim\The@GVerbatim
194     \def\The@GVerbatim{}\Verbatim@NextLine
195     \expandafter\Verbatim@@Input
196   \fi}%
197
198
199
200 \let\LWRFB@UseVerbatim\UseVerbatim
201 \renewcommand*{\UseVerbatim}[1]{%
202     \LWR@atbeginverbatim{Verbatim}%
203     \LWRFB@UseVerbatim{#1}%
```

```
204     \LWR@afterendverbatim%
205 }
206
207 \let\LWRFB@LUseVerbatim\LUseVerbatim
208
209 \renewcommand*{\LUseVerbatim}[1]{%
210     \LWR@atbeginverbatim{LVerbatim}%
211     \noindent%
212     \LWRFB@LUseVerbatim{#1}%
213     \LWR@afterendverbatim%
214 }
215
216 \def\@BUseVerbatim[#1]#2{%
217     \LWR@atbeginverbatim{BVerbatim}%
218     \LWRFB@UseVerbatim{#2}%
219     \LWR@afterendverbatim%
220 }
```

File 153   **`lwarp-fancyhdr.sty`**

§ 265   Package   # fancyhdr

(*Emulates or patches code by* PIET VAN OOSTRUM.)

fancyhdr (*Pkg*)   fancyhdr is ignored.

for HTML output: Discard all options for lwarp-fancyhdr:

```
1 \LWR@ProvidesPackageDrop{fancyhdr}[2025/01/07]
```

```
2 \newcommand*{\fancyhead}[2][]{}
3 \newcommand*{\fancyfoot}[2][]{}
4 \newcommand*{\fancyhf}[2][]{}
5
6 \newcommand*{\lhead}[2][]{}
7 \newcommand*{\chead}[2][]{}
8 \newcommand*{\rhead}[2][]{}
9 \newcommand*{\lfoot}[2][]{}
10 \newcommand*{\cfoot}[2][]{}
11 \newcommand*{\rfoot}[2][]{}
12 \newcommand*{\headrulewidth}{}
13 \newcommand*{\footrulewidth}{}
14 \providecommand{\headruleskip}{0pt}
15 \providecommand{\footruleskip}{0pt}
16 \newcommand{\plainheadrulewidth}{0pt}
17 \newcommand{\plainfootrulewidth}{0pt}
18 \def\fancyplain#1#2{#1}
19 \newcommand*{\headrule}{}
20 \newcommand*{\footrule}{}
21 \newlength{\headwidth}
22 \newcommand*{\fancycenter}[1][1em]{}
23 \newcommand*{\fancyheadoffset}[2][]{}
24 \newcommand*{\fancyfootoffset}[2][]{}
25 \newcommand*{\fancyhfoffset}[2][]{}
26 \newcommand{\fancyheadinit}[1]{}
27 \newcommand{\fancyfootinit}[1]{}
28 \newcommand{\fancyhfinit}[1]{}
```

```
29 \newcommand*{\iffloatpage}[2]{#2}
30 \newcommand*{\ifftopfloat}[2]{#2}
31 \newcommand*{\iffbotfloat}[2]{#2}
32 \newcommand*{\iffootnote}[2]{#2}
33
34 \NewDocumentCommand{\fancypagestyle}{s m o m}{}
35
36 \newcommand\fancyhdrsettoheight[2]{}
37 \newcommand{\fancyfootalign}[1]{}
38 \NewDocumentCommand{\fancyhdrbox}{ O{cl} o m }{}
39 \newcommand\fancypagestyleassign[2]{}
40 \newcommand{\fancyheadwidth}[2][]{}
41 \newcommand{\fancyfootwidth}[2][]{}
42 \newcommand{\fancyhfwidth}[2][]{}
```

File 154   `lwarp-fancypar.sty`

§ 266   Package   # fancypar

(*Emulates or patches code by* GONZALO MEDINA.)

fancypar (*Pkg*)   fancypar is used as-is for print output, and emulated for HTML.

⚠   css classes   \NotebookPar and related are used as-is inside a lateximage, but for HTML these are emulated as a <div> of class NotebookPar, etc. For HTML, the package options and the macro optional arguments are ignored. The user must provide custom css for each if visual effects are required. See section 7.7.

⚠   custom styles   If using a custom paragraph style, such as \MyStylePar from the documentation, use the following to generate an HTML <div> of class MyStylePar:

```
... (existing definiton of \MyStylePar, print version) ...
\begin{warpHTML}
\AddFancyparClass{MyStyle}
\end{warpHTML}
```

\MyStylePar is then modified to emulate HTML. An optional argument is allowed, which is ignored.

for HTML output: 1 \LWR@ProvidesPackagePass{fancypar}[2019/01/18]

```
2 \begin{warpHTML}
3 \makeatletter
4
5 \newcommand{\LWR@fancypar}[2]{%
6     \begin{BlockClass}{#1Par}
7     #2
8     \end{BlockClass}
9 }
10
11 \newcommand{\LWR@HTML@NotebookPar}[2][]{\LWR@fancypar{Notebook}{#2}}
12 \LWR@formatted{NotebookPar}
13
14 \newcommand{\LWR@HTML@ZebraPar}[2][]{\LWR@fancypar{Zebra}{#2}}
15 \LWR@formatted{ZebraPar}
16
```

```
17 \newcommand{\LWR@HTML@DashedPar}[2][]{\LWR@fancypar{Dashed}{#2}}
18 \LWR@formatted{DashedPar}
19
20 \newcommand{\LWR@HTML@MarkedPar}[2][]{\LWR@fancypar{Marked}{#2}}
21 \LWR@formatted{MarkedPar}
22
23 \newcommand{\LWR@HTML@UnderlinedPar}[2][]{\LWR@fancypar{Underlined}{#2}}
24 \LWR@formatted{UnderlinedPar}
25
26
27 \newcommand{\LWR@HTML@add@fancy@format}{}
28 \LWR@formatted{add@fancy@format}
29
30
31 \newcommand{\AddFancyparClass}[1]{%
32     \expandafter\newcommand\csname LWR@HTML@#1Par\endcsname[2][]{%
33         \LWR@fancypar{#1}{##2}%
34     }
35     \LWR@formatted{#1Par}
36 }
37
38 \makeatother
39 \end{warpHTML}
```

File 155   `lwarp-fancyref.sty`

§ 267   Package   # fancyref

(*Emulates or patches code by* AXEL REICHERT.)

fancyref (*Pkg*)   fancyref is modifed for HTML output.

for HTML output: `1 \LWR@ProvidesPackagePass{fancyref}[1999/02/03]`

\fancyrefhook (*Hook*) [fancyref]   To remove the margin option, if \fancyrefhook is anything other than the paren option, then force it to the default instead. (Comparing to the margin option was not possible since lwarp has revised the meaning of \mbox so the comparison failed.)

```
2 \newcommand*{\LWRfref@parenfancyrefhook}[1]{(#1)}
3
4 \ifdefstrequal{\fancyrefhook}{\LWRfref@parenfancyrefhook}
5 {}{
6     \renewcommand*{\fancyrefhook}[1]{#1}%
7 }
```

File 156   `lwarp-fancytabs.sty`

§ 268   Package   # fancytabs

fancytabs (*Pkg*)   fancytabs is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{fancytabs}[2016/03/29]`

```
2 \newcommand{\fancytab}[3][RIGHT]{}
3 \newcommand{\fancytabsStyle}[1]{}
4 \newcommand{\fancytabsHeight}[1]{}
5 \newcommand{\fancytabsWidth}[1]{}
6 \newcommand{\fancytabsCount}[1]{}
7 \newcommand{\fancytabsLeftColor}[1]{}
8 \newcommand{\fancytabsRightColor}[1]{}
9 \newcommand{\fancytabsTop}[1]{}
10 \newcommand{\fancytabsTextVPos}[1]{}
11 \newcommand{\fancytabsTextHPos}[1]{}
12 \newcommand{\fancytabsGap}[1]{}
13 \newcommand{\fancytabsFloor}[1]{}
14 \newcommand{\fancytabsRotate}[1]{}
```

File 157   **lwarp-fancyvrb.sty**

§ 269   Package   # fancyvrb

(*Emulates or patches code by* TIMOTHY VAN ZANDT.)

fancyvrb (*Pkg*)   fancyvrb is supported with some patches.

HTML classes   The fancy verbatim environment is placed inside a <div> of class fancyvrb. The label is placed inside a <div> of class fancyvrblabel. The verbatim text itself is placed inside a <div> of class verbatim.

For an inline \Verb, the verbatim is placee inside a <span> of class fancyvrb.

fancybox, fancyvrb   If using fancybox or fancyvrb with \VerbatimFootnotes, and using footnotes in a
⚠ \VerbatimFootnotes   sectioning command or display math, use \footnotemark and \footnotetext:
⚠   sectioning or
displaymath

```
    \subsection[Subsection Name]
        {Subsection Name\protect\footnotemark}
\footnotetext{A footnote with \verb+verbtim+.}
```

and likewise for equations or display math.

```
1 \AtBeginDocument{\RequirePackage{xcolor}}% for \convertcolorspec
2
3 \LWR@ProvidesPackagePass{fancyvrb}[2024/01/20]
```

For fvextra, to support background colors.

```
4 \newcommand*{\LWR@currentFVbackstyle}{}
5
6 \newcommand*{\LWR@find@currentFVbackstyle}{%
7    \renewcommand*{\LWR@currentFVbackstyle}{}%
8 }%
```

Initial default patch for fancyvrb:

```
9 \fvset{frame=none}%
```

Gobble does not work with HTML sanitization, so if gobbling is non-zero then turn off HTML sanitization.

```
10 \define@key{FV}{gobble}{%
11   \@tempcnta=#1\relax
12   \ifnum\@tempcnta<\@ne
13     \let\FV@Gobble\relax
14   \else
15     \ifnum\@tempcnta>9
16       \FV@Error{gobble parameter must be less than 10}\FV@eha
17     \else
18       \renewcommand{\FV@@@@Gobble}[\@tempcnta]{}%
19       \let\FV@Gobble\FV@@Gobble%
20       \boolfalse{LWR@HTMLsanitize@tmpb@enable}%      lwarp
21     \fi
22   \fi}
```

\FancyVerbSpace   Force the use of a vibible space instead of an empty box.

```
23 \ifxetexorluatex
24 \def\LWR@HTML@FancyVerbSpace{\textvisiblespace}
25 \else
26 \@ifundefined{verbvisiblespace}%
27   {%
28     \@ifundefined{textvisiblespace}%
29     {\begingroup\catcode`\ =12 \gdef\LWR@HTML@FancyVerbSpace{\tt }\endgroup}
30     {\def\LWR@HTML@FancyVerbSpace{\textvisiblespace}}
31   }%
32   {\def\LWR@HTML@FancyVerbSpace{\verbvisiblespace}}
33 \fi
34 \LWR@formatted{FancyVerbSpace}
```

\FancyVerbTab   Set the visible tab. Unicode 240B (SYMBOL FOR HORIZONTAL TABULATION) and 21E5 (RIGHTWARDS ARROW TO BAR) both gave unindended new lines at each tab, so a simple pipe character is used instead.

For the print mode, the fancyvrb definition does not copy, so a simplified version is used.

```
35 \def\LWR@print@FancyVerbTab{|}
36 \def\LWR@HTML@FancyVerbTab{|}%
37 \LWR@formatted{FancyVerbTab}

38 \fvset{tabsize=8,showtabs=false}
```

\FV@CatCodes

```
39 \VerifyCommand[lwarp][fancyvrb]{\FV@CatCodes}{BF2C1F38D5FEF0658C18B636ACBDA40E}
40
41 \def\FV@CatCodes{%
42   \let\do\@makeother\dospecials  % The usual stuff.
43   \FV@ActiveWhiteSpace           % See below.
44   \FV@FontScanPrep               % See below.
45   \FV@CatCodesHook               % A style hook.
46   \FancyVerbCodes                % A user-defined hook.
47   \catcode`\>=12%                % lwarp
48   \catcode`\<=12%                % lwarp
49 }
```

\FV@GetLine   Added the catcode changes for < and> to avoid the effect of \@noligs for these

characters. They were being made active and thus would not be sensed by the search/replace to sanitize.

⚠ This code is sensitive to the use of %, and for some reason does not work if \expandafter immediately follows the < catcode change.

```
50 \VerifyCommand[lwarp][fancyvrb]{\FV@GetLine}{9B86134119C575F099B5B567A9B65A9F}
51
52 \def\FV@GetLine{%
53     \@noligs%
54     \catcode`\>=12%      lwarp, and the next line as well
55 % for some reason, there must not be a % after the following 12:
56     \catcode`\<=12
57 %
58     \expandafter\FV@CheckScan\FancyVerbGetLine%
59 }
```

Modified to sanitize HTML. \LWR@HTMLsanitize@tmpb is included in \FV@Line, so it will adjust if used inside an lateximage.

```
60 \VerifyCommand[lwarp][fancyvrb]{\FancyVerbGetLine}{498B88BACBD0811BAC0791BDF4F3B335}
61
62 \begingroup
63 \catcode`\^^M=\active%
64 \gdef\FancyVerbGetLine#1^^M{%
65     \@nil%
66     \FV@CheckEnd{#1}%
67     \ifx\@tempa\FV@EnvironName%          % True if end is found
68         \ifx\@tempb\FV@@@CheckEnd\else\FV@BadEndError\fi%
69         \let\next\FV@EndScanning%
70     \else%
71         \ifbool{LWR@HTMLsanitize@tmpb@enable}%                    lwarp
72             {\def\FV@Line{\LWR@HTMLsanitize@use@tmpb{#1}}}%     lwarp
73             {\def\FV@Line{#1}}%                                lwarp
74         \def\next{\FV@PreProcessLine\FV@GetLine}%
75     \fi%
76     \next}%
77 \endgroup
```

\FV@List   {⟨R/L margin 0/1⟩}

Modified to always allow line wrapping because added HTML tags may make run off the end of the line in the PDF output file before conversion to HTML.

```
78 \VerifyCommand[lwarp][fancyvrb]{\FV@List}{8FB649FAF7C9487B257B76AF4FFB27D1}
79
80 \def\FV@List#1{%
81     \begingroup
82     \FV@UseKeyValues
83     \FV@LeaveVMode
84     \if@inlabel\else\setbox\@labels=\box\voidb@x\fi
85     \FV@ListNesting{#1}%
86     \FV@ListParameterHook
87     \FV@ListVSpace
88     \FV@SetLineWidth
89     \FV@InterLinePenalty
90 %   \let\FV@ProcessLine\FV@ListProcessLine@i
91     \let\FV@ProcessLine\FV@ListProcessLine%              lwarp
92     \FV@CatCodes
93     \FV@FormattingPrep
94     \FV@ObeyTabsInit
95     \FV@BeginListFrame}
```

\FV@ListProcessLine   {⟨*text*⟩}

Processes each line, adding optional line numbers. Modified to always allow line wrapping because added HTML tags may make run off the end of the line in the PDF output file before conversion to HTML.

```
 96 \VerifyCommand[lwarp][fancyvrb]{\FV@ListProcessLine}{660F9938234FC1043ACF7B02B3F37372}
 97
 98 \def\FV@ListProcessLine#1{%
 99   \hbox to \hsize{%
100 %     \kern\leftmargin
101     \hbox to \VerbatimHTMLWidth {%
102     \ifcsvoid{FV@LeftListNumber}{}{\kern 2.5em}%
103       \FV@LeftListNumber%
104 %       \FV@LeftListFrame
105     \FancyVerbFormatLine{#1}%
106     \hss%
107 %       \FV@RightListFrame
108     \FV@RightListNumber%
109   }%
110     \hss% required to avoid underfull hboxes
111 }
112 }
```

\FVC@SaveVerb   {⟨*name*⟩} {⟨*character*⟩}

Modified to sanitize HTML when stored. Sanitizing on use would be too late to adjust catcodes. \LWR@HTMLsanitize@tmpb is included in the saved macro, so if inside a lateximage, \LWR@HTMLsanitize@tmp does nothing.

```
113 \VerifyCommand[lwarp][fancyvrb]{\FVC@SaveVerb}{68373ED055890622906844A5611810C8}
114
115 \begingroup
116 \catcode`\^^M=\active%
117 \gdef\FVC@SaveVerb#1#2{%
118   \@namedef{FV@SV@#1}{}%
119   \begingroup%
120     \FV@UseKeyValues%
121     \FV@CatCodes%
122     \outer\def^^M{\FV@EOL}%
123     \global\let\@tempg\FancyVerbAfterSave%
124     \catcode`#2=12%
125     \def\@tempa{\def\FancyVerbGetVerb####1####2}%
126 %     \expandafter\@tempa\string#2{\endgroup\@namedef{FV@SV@#1}{##2}\@tempg}%
127     \expandafter\@tempa\string#2{%
128         \endgroup%
129         \@namedef{FV@SV@#1}{%
130 %           ##2
131             \LWR@HTMLsanitize@use@tmpb{##2}%     lwarp
132       }\@tempg%
133     }%
134     \FancyVerbGetVerb\FV@EOL}%
135 \endgroup
```

\FV@UseVerb   {⟨*macro*⟩}

Adds a <span>.

```
136 \VerifyCommand[lwarp][fancyvrb]{\FV@UseVerb}{A3A9D802CCDBEC8D2FDAB626346B5EDD}
137
```

```
138 \def\FV@UseVerb#1{%
139    \ifbool{LWR@verbtags}%                    lwarp
140     {\LWR@htmltag{span class=\textquotedbl{}fancyvrb\textquotedbl}}% lwarp
141        {}%                                    lwarp
142    \mbox{\FV@UseKeyValues\FV@FormattingPrep#1}%
143    \ifbool{LWR@verbtags}%                    lwarp
144        {\LWR@htmltag{/span}}%                lwarp
145        {}%                                    lwarp
146 }
```

\FVC@Verb   {⟨*character*⟩}

Modified to sanitize HTML.

```
147 \VerifyCommand[lwarp][fancyvrb]{\FVC@Verb}{806B03D5A78CAB39E0514667991695C9}
148
149 \begingroup
150 \catcode`\^^M=\active%
151 \gdef\FVC@Verb#1{%
152   \begingroup%
153     \FV@UseKeyValues%
154     \FV@FormattingPrep%
155     \FV@CatCodes%
156     \outer\def^^M{}%
157     \catcode`#1=12%
158     \def\@tempa{\def\FancyVerbGetVerb####1####2}%
159     \expandafter\@tempa\string#1{%
160        \ifbool{LWR@verbtags}%                    lwarp
161          {%                                      lwarp
162         \LWR@htmltag{span class=\textquotedbl{}fancyvrb\textquotedbl}% lwarp
163            \def\tmpb{##2}%                        lwarp
164            \LWR@HTMLsanitize@tmpb%                lwarp
165            \mbox{\tmpb}%                          lwarp
166            \LWR@htmltag{/span}%                   lwarp
167          }% lwarp
168          {\mbox{##2}}%
169        \endgroup%
170     }%
171     \FancyVerbGetVerb\FV@EOL%
172 }%
173 \endgroup
```

\FV@ReadLine   Modified to sanitize HTML.

```
174 \VerifyCommand[lwarp][fancyvrb]{\FV@ReadLine}{3C3481D735295DAEB5B30DDE9152287D}
175
176 \begingroup
177 \catcode`\^^M=\active
178 \gdef\FV@ReadLine{%
179   \ifeof\FV@InFile\else
180 %    \immediate\read\FV@InFile to\@tempa%
181 %    \expandafter\FV@@ReadLine\@tempa^^M\relax^^M\@nil%
182     \immediate\read\FV@InFile to\tmpb%
183     \LWR@HTMLsanitize@tmpb%
184     \expandafter\FV@@ReadLine\tmpb^^M\relax^^M\@nil%
185   \fi}
186 \endgroup
```

\LWR@FVstyle   Holds the style of the verbatim.

```
187 \newcommand*{\LWR@FVstyle}{}
```

\LWR@new@FVstyle   Starts a new style for the next verbatim.

```
188 \newcommand*{\LWR@new@FVstyle}{\renewcommand*{\LWR@FVstyle}{}}
```

\LWR@addto@FVstyle   {⟨*style text*⟩}   Adds to the style for the next verbatim.

```
189 \newcommand*{\LWR@addto@FVstyle}[1]{%
190     \apptocmd{\LWR@FVstyle}{\protect\LWR@indentHTMLtwo#1}%
191         {}%
192         {%
193             \PackageWarning{lwarp}{%
194                 \string\LWR@addto@FVstyle\space was not
195                 able to patch an HTML style.\MessageBreak
196                 Patching !#1!\MessageBreak%
197                 Currently is !\LWR@FVstyle!\MessageBreak%
198                 HTML styles may be not be complete%
199             }%
200         }%
201 }
```

\LWR@addFVtextcolorstyle   Adds a color style for the current color.

```
202 \newcommand*{\LWR@addFVtextcolorstyle}{%
203     \LWR@findcurrenttextcolor%
204     \ifdefstring{\LWR@tempcolor}{000000}%
205         {}%
206         {\LWR@addto@FVstyle{color: \LWR@origpound\LWR@tempcolor ; }}%
207 }
```

\LWR@addFVbackgroundstyle

```
208 \newcommand*{\LWR@addFVbackgroundstyle}{%
209     \ifdefempty{\LWR@currentFVbackstyle}%
210         {}%
211         {\LWR@addto@FVstyle{\LWR@currentFVbackstyle}}%
212 }
```

\LWR@addFVborderstyle   {⟨*edge*⟩}

```
213 \newcommand*{\LWR@addFVborderstyle}[1]{%
214     \LWR@addto@FVstyle{\LWR@FVborderstyle{#1}}%
215 }
```

\LWR@addFVborderstyles   Redefined on the fly to do \LWR@addFVborderstyle, possibly several times, each
                           with its own edge argument.

```
216 \newcommand*{\LWR@addFVborderstyles}{}
```

                           After the preamble is loaded, after any patches to Verbatim, such as by fvextra:

```
217 \AfterEndPreamble{
218
219 \LWR@traceinfo{Patching fancyvrb.}
```

\VerbatimFootnotes   Patched to use the new version.

```
220 \VerifyCommand[lwarp][fancyvrb]{\VerbatimFootnotes}{931C9BE6284EB9D8B1516D566C997A87}
221
222 \def\VerbatimFootnotes{%
223     \let\@footnotetext\V@footnotetext%
224     \let\footnote\V@footnote%
225     \let\LWR@footnotetext\V@footnotetext% lwarp
226 }
```

\V@@footnotetext   Patches in a subset of lwarp's \LWR@footnotetext to the fancyvrb version of \V@@footnotetext.

```
227 \VerifyCommand[lwarp][fancyvrb]{\V@@footnotetext}{89E1586855997F951F57C3936CBDF87E}
228
229 \def\V@@footnotetext{%
230 \LWR@traceinfo{V@footnotetext}%
```

Place an autopage marker so that back references to citations inside a footnote will link closer to the footnote text, if possible.

```
231     \LWR@newautopagelabel{page}%
```

Take the current footnote box, then append:

```
232     \global\setbox\LWR@footnotebox=\vbox\bgroup%
```

Add to any current footnotes:

```
233     \unvbox\LWR@footnotebox%
```

Remember the footnote number for \ref:

```
234     \protected@edef\@currentlabel{%
235         \csname p@footnote\endcsname\@thefnmark%
236     }% @currentlabel
```

Use HTML superscripts in the footnote even inside a lateximage:

```
237     \renewrobustcmd{\textsuperscript}[1]{\LWR@htmlspan{sup}{##1}}%
```

Verbatim tags and HTML sanitization will have been turned off inside a lateximage, such as in svg math, so turn them on here so they will be active in the HTML footnotes.

```
238     \booltrue{LWR@verbtags}%
239     \booltrue{LWR@HTMLsanitize@tmpb@enable}%
```

Use paragraph tags if in a tabular data cell or a lateximage:

```
240     \LWR@htmltagc{\LWR@tagregularparagraph}\LWR@orignewline%
```

Append the footnote mark to the list:

```
241     \@makefntext{}%
```

The footnote text will follow after \V@@@footnotetext has completed.

```
242     \bgroup%
243     \aftergroup\V@@@footnotetext%
```

Do not generate autopages inside the footnotes, since they are accumulated at the moment before finally being used perhaps on a later page.

```
244     \let\LWR@newautopagelabel\LWR@null@newautopagelabel%
245     \ignorespaces%
246 }%
```

\V@@@footnotetext   Improves <par>.

247 \def\V@@@footnotetext{%
248   \LWR@orignobreakspace\LWR@orignewline%
249   \LWR@htmltagc{/\LWR@tagregularparagraph}\LWR@orignewline%
250   \strut\egroup%
251 }

\FVB@Verbatim

\FVB@LVerbatim   Prevents unexpected page break in the PDF output before HTML conversion.

252 \preto\FVB@Verbatim{\LWR@forcenewpage}
253 \preto\FVB@LVerbatim{\LWR@forcenewpage}
254 % \preto\FVB@BVerbatim{\LWR@forcenewpage}% Fails, so done below.

        Simplified to remove PDF formatting:

255 \def\LWR@HTML@FV@BeginListFrame@Single{%
256   \FV@SingleFrameLine{\z@}%
257 }
258 \LWR@formatted{FV@BeginListFrame@Single}
259
260 \def\LWR@HTML@FV@EndListFrame@Single{%
261   \FV@SingleFrameLine{\@ne}%
262 }
263 \LWR@formatted{FV@EndListFrame@Single}
264
265 \def\LWR@HTML@FV@BeginListFrame@Lines{%
266   \FV@SingleFrameLine{\z@}%
267 }
268 \LWR@formatted{FV@BeginListFrame@Lines}
269
270 \def\LWR@HTML@FV@EndListFrame@Lines{%
271   \FV@SingleFrameLine{\@ne}%
272 }
273 \LWR@formatted{FV@EndListFrame@Lines}
274
275 \newcommand*{\LWR@HTML@FV@SingleFrameSep}{}%
276 \LWR@formatted{FV@SingleFrameSep}

        The following patches to Verbatim are executed at the start and end of the envi-
        ronment, depending on the choice of frame.

\LWR@fvstartnone   Wraps label in a <div> of class fancyvrblabel.

277 \newcommand*{\LWR@fvstartnone}{%
278 \LWR@traceinfo{fvstartnone}%
279 % \hbox to\z@{

        Accumulate the style settings. Text color, background color, and background style
        are detected now. Border styles are selected earlier by the choice of verbatim
        frames.

        If the current text style is empty, do not print an HTML style.

280   \LWR@find@currentFVbackstyle%   lwarp
281   \LWR@new@FVstyle%   lwarp
282   \LWR@addFVtextcolorstyle%   lwarp
283   \LWR@addFVbackgroundstyle%   lwarp

```
284        \LWR@addFVborderstyles%           lwarp
285     \ifdefempty{\LWR@FVstyle}%
286        {\BlockClass{fancyvrb}}%
287        {\BlockClass[\LWR@FVstyle]{fancyvrb}}%

288 \LWR@stoppars
289 \ifx\FV@LabelPositionTopLine\relax\else
290    \ifx\FV@LabelBegin\relax\else
291       \FancyVerbRuleColor{\LWR@FVfindbordercolor}
292       \ifbool{LWR@verbtags}%        lwarp
293          {%                         lwarp
294             \LWR@findcurrenttextcolor%
295             \LWR@htmltagc{%         lwarp
296             div class=\textquotedbl{}fancyvrblabel\textquotedbl\ % space
297                  style=\textquotedbl{}%
298                     color: \LWR@origpound\LWR@tempcolor%
299                  \textquotedbl%
300             }%
301             \LWR@orignewline%    lwarp
302          }%                        lwarp
303          {}%                       lwarp
304       \LWR@print@textrm{\FV@LabelBegin}% \textrm preserves emdash
305       \LWR@orignewline%                 lwarp
306       \ifbool{LWR@verbtags}%            lwarp
307          {%                            lwarp
308             \LWR@htmltagc{/div}%       lwarp
309             \LWR@orignewline%          lwarp
310          }%                           lwarp
311          {}%                          lwarp
312    \fi
313 \fi
314 \LWR@atbeginverbatim{verbatim}%
315 % }% hbox
316 }
```

\LWR@fvendnone   Wraps label in a <div> of class fancyvrblabel.

```
317 \newcommand*{\LWR@fvendnone}{%
318 \LWR@traceinfo{fvendnone}%
319 % \hbox to\z@{
320 \LWR@afterendverbatim%
321 \LWR@stoppars%
322 \ifx\FV@LabelPositionBottomLine\relax\else
323    \ifx\FV@LabelEnd\relax\else
324       \FancyVerbRuleColor{\LWR@FVfindbordercolor}%    lwarp
325       \ifbool{LWR@verbtags}%                          lwarp
326          {%                                          lwarp
327             \LWR@findcurrenttextcolor%
328             \LWR@htmltagc{%                          lwarp
329             div class=\textquotedbl{}fancyvrblabel\textquotedbl\ % space
330                  style=\textquotedbl{}%
331                     color: \LWR@origpound\LWR@tempcolor%
332                  \textquotedbl%
333             }%
334             \LWR@orignewline%                        lwarp
335          }%                                         lwarp
336          {}%                                        lwarp
337       \LWR@print@textrm{\FV@LabelEnd}%              lwarp
338       \LWR@orignewline%                             lwarp
339       \ifbool{LWR@verbtags}%                        lwarp
```

```
340                {%                                    lwarp
341                   \LWR@htmltagc{/div}%               lwarp
342                   \LWR@orignewline%                  lwarp
343                }%                                     lwarp
344                {}%                                    lwarp
345     \fi
346 \fi
347 \endBlockClass%       lwarp
348 % }% hbox
349 }
```

```
350 \newcommand*{\LWR@fvstartsingle}{%
351 \LWR@traceinfo{fvstartsingle}%
352 \LWR@fvstartnone%
353 \FV@BeginListFrame@Single%
354 }
355
356 \newcommand*{\LWR@fvendsingle}{%
357 \LWR@traceinfo{fvendsingle}%
358 \FV@EndListFrame@Single%
359 \LWR@fvendnone%
360 }
361
362 \newcommand*{\LWR@fvstartline}{%
363 \LWR@traceinfo{fvstartline}%
364 \LWR@fvstartnone%
365 % \setlength{\LWR@templengthone}{\baselineskip}%
366 \FV@BeginListFrame@Lines%
367 % \setlength{\baselineskip}{\LWR@templengthone}%
368 % \setlength{\baselineskip}{5pt}%
369 }
370
371 \newcommand*{\LWR@fvendline}{%
372 \LWR@traceinfo{fvendline}%
373 \FV@EndListFrame@Lines%
374 \LWR@fvendnone%
375 }
```

The following patches select the start/left/right/end behaviors depending on frame.

```
376 \newcommand*{\LWR@FVfindbordercolor}{%
377     \FancyVerbRuleColor%
378     \LWR@findcurrenttextcolor%
379     \color{black}%
380 }
```

\LWR@FVborderstyle   {⟨*edge*⟩} Border width of \FV@FrameRule. Edge is empty, -top, etc.

```
381 \newcommand*{\LWR@FVborderstyle}[1]{%
382     padding#1: \strip@pt\dimexpr \FV@FrameSep\relax\relax pt ; % space
383     \LWR@FVfindbordercolor\LWR@indentHTMLtwo%
384     border#1: \strip@pt\dimexpr \FV@FrameRule\relax\relax pt % space
385     solid {\FancyVerbRuleColor{\LWR@origpound\LWR@tempcolor}} ; % space
386 }
```

```
387 \VerifyCommand[lwarp][fancyvrb]{\FV@Frame@none}{C60E1656944AB4C4D2B74410E88FE7C0}
388
389 \def\LWR@HTML@FV@Frame@none{%
```

```
390 \let\FV@BeginListFrame\LWR@fvstartnone%
391 \let\FV@LeftListFrame\relax%
392 \let\FV@RightListFrame\relax%
393 \let\FV@EndListFrame\LWR@fvendnone}
394 \LWR@formatted{FV@Frame@none}
395
396 \FV@Frame@none% default values
```

```
397 \VerifyCommand[lwarp][fancyvrb]{\FV@Frame@single}{CDF78DB9C6408F48D05302D07091C629}
398
399 \def\LWR@HTML@FV@Frame@single{%
400 \renewcommand{\LWR@addFVborderstyles}{\LWR@addFVborderstyle{}}%  lwarp
401 \let\FV@BeginListFrame\LWR@fvstartsingle%
402 \let\FV@LeftListFrame\FV@LeftListFrame@Single%
403 \let\FV@RightListFrame\FV@RightListFrame@Single%
404 \let\FV@EndListFrame\LWR@fvendsingle}
405
406 \LWR@formatted{FV@Frame@single}
```

```
407 \VerifyCommand[lwarp][fancyvrb]{\FV@Frame@lines}{1AADD6691DA93C9A66227F5C5B34EAE4}
408
409 \def\LWR@HTML@FV@Frame@lines{%
410 \renewcommand{\LWR@addFVborderstyles}{%
411     \LWR@addFVborderstyle{-top}%
412     \LWR@addFVborderstyle{-bottom}%
413 }%  lwarp
414 \let\FV@BeginListFrame\LWR@fvstartline%
415 \let\FV@LeftListFrame\relax%
416 \let\FV@RightListFrame\relax%
417 \let\FV@EndListFrame\LWR@fvendline}
418
419 \LWR@formatted{FV@Frame@lines}
```

```
420 \VerifyCommand[lwarp][fancyvrb]{\FV@Frame@topline}{7E102D81F4FD367B398B8E85F48A7754}
421
422 \def\LWR@HTML@FV@Frame@topline{%
423 \renewcommand{\LWR@addFVborderstyles}{\LWR@addFVborderstyle{-top}}%  lwarp
424 \let\FV@BeginListFrame\LWR@fvstartline%
425 \let\FV@LeftListFrame\relax%
426 \let\FV@RightListFrame\relax%
427 \let\FV@EndListFrame\LWR@fvendnone}
428
429 \LWR@formatted{FV@Frame@topline}
```

```
430 \VerifyCommand[lwarp][fancyvrb]{\FV@Frame@bottomline}{A51600F812F57F4211EF9E34F261564A}
431
432 \def\LWR@HTML@FV@Frame@bottomline{%
433 \renewcommand{\LWR@addFVborderstyles}{\LWR@addFVborderstyle{-bottom}}% lwarp
434 \let\FV@BeginListFrame\LWR@fvstartnone%
435 \let\FV@LeftListFrame\relax%
436 \let\FV@RightListFrame\relax%
437 \let\FV@EndListFrame\LWR@fvendline}
438
439 \LWR@formatted{FV@Frame@bottomline}
```

Seems to be required in some situations. Is not \LWR@formatted because it is
defined as needed.

```
440 \def\FV@FrameFillLine{}
```

```
441 \VerifyCommand[lwarp][fancyvrb]{\FV@Frame@leftline}{2A77982C6520FD64F6DBFA1C03B670BA}
442
443 \def\LWR@HTML@FV@Frame@leftline{%
444 \renewcommand{\LWR@addFVborderstyles}{\LWR@addFVborderstyle{-left}}}% lwarp
```

To define the \FV@FrameFillLine macro (from \FV@BeginListFrame)

```
445 \ifx\FancyVerbFillColor\relax%
446 \let\FV@FrameFillLine\relax%
447 \else%
448 \@tempdima\FV@FrameRule\relax%
449 \multiply\@tempdima-\tw@%
450 \edef\FV@FrameFillLine{%
451 {\noexpand\FancyVerbFillColor{\vrule\@width\number\@tempdima sp}%
452 \kern-\number\@tempdima sp}}%
453 \fi%
454 \let\FV@BeginListFrame\LWR@fvstartnone%
455 \let\FV@LeftListFrame\FV@LeftListFrame@Single%
456 \let\FV@RightListFrame\relax%
457 \let\FV@EndListFrame\LWR@fvendnone}
458
459 \LWR@formatted{FV@Frame@leftline}
```

\FV@SingleFrameLine   Adds the optional label to the top and bottom edges.

```
460 \VerifyCommand[lwarp][fancyvrb]{\FV@SingleFrameLine}{2D8B1DAED851500F255E357437FF065C}
461
462 \def\LWR@HTML@FV@SingleFrameLine#1{%
463 %    \hbox to\z@{%
464 %      \kern\leftmargin
465    \ifnum#1=\z@\relax
466      \let\FV@Label\FV@LabelBegin
467    \else
468      \let\FV@Label\FV@LabelEnd
469    \fi
470    \ifx\FV@Label\relax
471 %        \FancyVerbRuleColor{\vrule \@width\linewidth \@height\FV@FrameRule}%
472    \else
473      \ifnum#1=\z@
474 %        \setbox\z@\hbox{\strut\enspace\FV@LabelBegin\enspace\strut}%
475        \ifx\FV@LabelPositionTopLine\relax
476        \else
477        \fi
478      \else
479 %        \setbox\z@\hbox{\strut\enspace\FV@LabelEnd\enspace\strut}%
480        \ifx\FV@LabelPositionBottomLine\relax
481        \else
482        \fi
483      \fi
484    \fi
485 %    \hss
486 %    }
487 }
488 \LWR@formatted{FV@SingleFrameLine}
```

Env BVerbatim   Adds lwarp verbatim patches.

```
489 \xpretocmd{\FV@BeginVBox}
490     {%
491         \LWR@forcenewpage%
492         \LWR@atbeginverbatim{bverbatim}%
493     }
494     {}
495     {\LWR@patcherror{fancyvrb}{FV@BeginVBox}}
496
497 \xapptocmd{\FV@EndVBox}
498     {%
499         \LWR@afterendverbatim%
500     }
501     {}
502     {\LWR@patcherror{fancyvrb}{FV@EndVBox}}
```

End of the modifications to make at the end of the preamble:

```
503 } % \AfterEndPreamble
```

```
504 \VerifyCommand[lwarp][fancyvrb]{\FVB@VerbatimOut}{A0AC591D2DB283DCEBCCC75968FF88CF}
505
506 \def\FVB@VerbatimOut#1{%
507   \@bsphack
508   \begingroup
509     \FV@UseKeyValues
510     \FV@DefineWhiteSpace
511     \def\FV@Space{\space}%
512     \FV@DefineTabOut
513     \def\FV@ProcessLine{\immediate\write\FV@OutFile}%
514     \immediate\openout\FV@OutFile #1\relax
515     \let\FV@FontScanPrep\relax
516 %% DG/SR modification begin - May. 18, 1998 (to avoid problems with ligatures)
517     \let\@noligs\relax
518 %% DG/SR modification end
519   \boolfalse{LWR@HTMLsanitize@tmpb@enable}%        lwarp
520     \FV@Scan}
```

File 158   **lwarp-fbox.sty**

§ 270   Package   # fbox

(*Emulates or patches code by* HERBERT VOSS.)

fbox (*Pkg*)   fbox is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{fbox}[2022/02/20]

This will be \LWR@formatted when \AtBeginDocument:

2 \LetLtxMacro\LWR@HTML@fbox\fbox

Instead of using the original, the new version is used with all borders:

3 \renewcommand*{\orig@fbox}{\FBox@i[tblr]}

\WR@fboxpkg@border   {⟨*1: top/bottom/left/right*⟩} {⟨*2: t/b/l/r*⟩} {⟨*3: padding, or empty*⟩}

Accumulates HTML styles for border, and padding if given:

```
4 \newcommand*{\LWR@fboxpkg@border}[3]{%
5     \colorlet{LWR@border@color}{\csuse{fbox@#2color}}%
6     \protect\convertcolorspec{named}{LWR@border@color}{HTML}\LWR@tempbordercolor\relax%
7     \appto\LWR@tempone{%
8         border-#1: % space
9         \LWR@printlength{\LWR@atleastonept} % space
10         solid \LWR@origpound%
11     }%
12    \expandafter\appto\expandafter\LWR@tempone\expandafter{\LWR@tempbordercolor}%
13    \appto\LWR@tempone{ ;\LWR@indentHTML}%
14    \ifblank{#3}{}{%
15        \appto\LWR@tempone{%
16            padding-#1: \LWR@printlength{#3} ;\LWR@indentHTML
17        }%
18    }%
19 }
```

A hack to reuse the same code for inline and blocks:

```
20 \newbool{LWR@fboxpkg@ispar}
21 \boolfalse{LWR@fboxpkg@ispar}
```

Accumulate HTML styles for left and right padding, depending on \if@fbox@space@left, \if@fbox@space@right:

```
22 \newcommand{\LWR@fboxpkg@lrpadding}[1]{%
23     \csuse{if@fbox@space@#1}%
24         \appto\LWR@tempone{%
25             padding-#1: \LWR@printlength{\fbox@@sep};\LWR@indentHTML
26         }
27     \else%
28         \appto\LWR@tempone{%
29             padding-#1: 0pt;\LWR@indentHTML
30         }
31     \fi%
32 }
```

The HTML version, modified to use HTML styles and either an \InlineClass or BlockClass:

```
33 \newcommand{\LWR@HTML@FBox@iii}[1]{%
```

Find and set the text color, rule width, margin:

```
34     \LWR@forceminwidth{\fbox@@rule}%
35     \LWR@findcurrenttextcolor%
36     \def\LWR@tempone{%
37         color: \LWR@origpound\LWR@tempcolor ; \LWR@indentHTML
38         margin: 1ex ; \LWR@indentHTML
39     }%
```

Add left/right padding:

```
40     \LWR@fboxpkg@lrpadding{left}%
41     \LWR@fboxpkg@lrpadding{right}%
```

Per the original to decode the borders, in a new way:

```
42     \ifnum\the\@tempcntb>8\relax
43         \advance\@tempcntb by -8\relax
44         \LWR@fboxpkg@border{top}{t}{\fbox@@sep}%
45     \fi
46     \ifnum\@tempcntb>3
47         \advance\@tempcntb by -4\relax
48         \LWR@fboxpkg@border{left}{l}{}%
49     \fi
50     \ifnum\@tempcntb>1\relax
51         \LWR@fboxpkg@border{right}{r}{}%
52     \fi
53     \ifodd\@tempcntb
54         \LWR@fboxpkg@border{bottom}{b}{\fbox@@sep}%
55     \fi
```

Generate a BlockClass or \InlineClass with the contents:

```
56     \color@begingroup
57     \ifbool{LWR@fboxpkg@ispar}%
58         {%
59             \begin{BlockClass}[\LWR@tempone]{fboxpkg}%
60                 #1%
61             \end{BlockClass}%
62         }%
63         {%
64             \InlineClass[\LWR@tempone]{fboxpkg}{%
65                 #1%
66             }%
67         }%
68     \color@endgroup
69     \boolfalse{LWR@fboxpkg@ispar}% globally
70 }
71 \LWR@formatted{FBox@iii}
```

For \fparbox, set the use of BlockClass, then reuse the above:

```
72 \long\def\LWR@HTML@FParBox@i[#1]#2{%
73     \booltrue{LWR@fboxpkg@ispar}%
74     \FBox@i[#1]{#2}
75 }
76 \LWR@formatted{FParBox@i}
77
78 \long\def\LWR@HTML@FParBox@ii#1{%
79     \booltrue{LWR@fboxpkg@ispar}%
80     \FBox@i[tblr]{#1}%
81 }
82 \LWR@formatted{FParBox@ii}
```

For MathJax, absorb and ignore star and optional arguments:

```
83 \CustomizeMathJax{\let\LWRorigfbox\fbox}
84 \CustomizeMathJax{\newcommand{\LWRfboxpkgtwo}[2][]{\LWRorigfbox{#2}}}
85 \CustomizeMathJax{\renewcommand{\fbox}{\ifstar\LWRfboxpkgtwo\LWRfboxpkgtwo}}
86 \CustomizeMathJax{\newcommand{\fparbox}{\fbox}}
```

File 159   **lwarp-fewerfloatpages.sty**

§ 271   Package   # fewerfloatpages

fewerfloatpages (*Pkg*)   fewerfloatpages is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{fewerfloatpages}[2020/02/14]

2 \newcommand\floatpagekeepfraction{\textfraction}
3 \newcounter{floatpagedeferlimit}
4 \newcounter{floatpagekeeplimit}

File 160   **lwarp-figcaps.sty**

§ 272   Package   # figcaps

(*Emulates or patches code by* Patrick W. Daly.)

figcaps (*Pkg*)   figcaps is ignored.

for HTML output: Discard all options for lwarp-figcaps:

1 \LWR@ProvidesPackageDrop{figcaps}[1999/02/23]

2 \newcommand*{\figcapson}{}
3 \newcommand*{\figcapsoff}{}
4 \newcommand*{\printfigures}{}
5 \newcommand*{\figmarkon}{}
6 \newcommand*{\figmarkoff}{}
7 \def\figurecapname{Figure Captions}
8 \def\tablepagename{Tables}
9 \def\figurepagename{Figures}

File 161   **lwarp-figsize.sty**

§ 273   Package   # figsize

(*Emulates or patches code by* Anthony A. Tanbakuchi.)

figsize (*Pkg*)   figsize is emulated.

for HTML output: 1 \LWR@ProvidesPackageDrop{figsize}[2002/03/18]

Emulates a virtual 6×9 inch textsize.

2 \newlength{\figwidth}
3 \newlength{\figheight}
4
5 \newcommand{\SetFigLayout}[3][0]{%

```
6 \setlength{\figheight}{8in}%
7 \setlength{\figheight}{\figheight / #2}%
8 %
9 \setlength{\figwidth}{5.5in}%
10 \setlength{\figwidth}{\figwidth / #3}%
11 }
```

File 162 **`lwarp-fitbox.sty`**

§ 274   Package   # fitbox

fitbox (*Pkg*)   fitbox is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{fitbox}[2019/02/20]`

```
2 \NewDocumentCommand{\fitbox}{s o m}{%
3     \begin{BlockClass}{fitbox}
4     #3
5     \end{BlockClass}
6 }
7
8 \newcommand*{\fitboxset}[1]{}
9
10 \newdimen\fitboxnatheight
11 \newdimen\fitboxnatwidth
12
13 \newcommand\SetFitboxLayout[3][]{}
```

File 163   **`lwarp-fix2col.sty`**

§ 275   Package   # fix2col

fix2col (*Pkg*)   fix2col is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{fix2col}[2015/11/13]`

File 164   **`lwarp-fixmath.sty`**

§ 276   Package   # fixmath

(*Emulates or patches code by* WALTER SCHMIDT.)

fixmath (*Pkg*)   fixmath is used as-is for svg math, and emulated for MATHJAX.

⚠ limitations   MATHJAX does not have full font support for bold italic Greek.

for HTML output: `1 \LWR@ProvidesPackagePass{fixmath}[2000/04/11]`

```
2 \LWR@origRequirePackage{lwarp-common-mathjax-letters}
3
4 \begin{warpMathJax}
```

```
5 \LWR@mathjax@addgreek@u@it*{}{}
6 \LWR@mathjax@addletter{\BooleanTrue}{up}{}{delta}{0394}
7 \LWR@mathjax@addletter{\BooleanTrue}{up}{}{omega}{03A9}
8 \CustomizeMathJax{\newcommand{\mathbold}[1]{\boldsymbol{#1}}}
9 \end{warpMathJax}
```

File 165 **`lwarp-fixme.sty`**

§ 277   Package   # fixme

(Emulates or patches code by Didier Verna.*)*

fixme (*Pkg*)   fixme is patched for use by lwarp.

⚠   external layouts   External layouts (\fxloadlayouts) are not supported.

Customized layouts are overwritten by lwarp's versions \AtBeginDocument in order to provide the HTML conversion. If creating a new layout, see lwarp's changes to provide similar for the new layout, inside a warpHTML environment.

User control is provided for setting the HTML styling of the "faces". The defaults are as follows, and may be changed in the preamble after fixme is loaded:

```
\def\FXFaceInlineHTMLStyle{font-weight:bold}
\def\FXFaceEnvHTMLStyle{font-weight:bold}
\def\FXFaceSignatureHTMLStyle{font-style:italic}
\def\FXFaceTargetHTMLStyle{font-style:italic}
```

for HTML output: 1 \LWR@ProvidesPackagePass{fixme}[2019/01/03]

Restore lwarp's version of \@wrindex, ignoring the fixme package's target option:

2 \let\@wrindex\LWR@wrindex

Float-related macros required by lwarp:

```
3 \newcommand{\ext@fixme}{lox}
4
5 \renewcommand{\l@fixme}[2]{%
6     \hypertocfloat{1}{fixme}{lox}%
7       {\LWR@nameref{\BaseJobname-autopage-\arabic{LWR@nextautopage}} --- #1}%
8         {#2}
9 }
```

Other modifications. Done \AtBeginDocument to hopefully work if the user customizes the layouts.

```
10 \AtBeginDocument{
11
12 \def\FXFaceInlineHTMLStyle{font-weight:bold}
13
14 \renewcommand*\FXLayoutInline[3]{ % space
15     \InlineClass[\FXFaceInlineHTMLStyle]{fixmeinline}%
16         {\@fxtextstd{#1}{#2}{#3}}%
17 }
18
```

```
19 \def\FXFaceEnvHTMLStyle{font-weight:bold}
20
21 \renewcommand*\FXEnvLayoutPlainBegin[2]{%
22     \BlockClass[\FXFaceEnvHTMLStyle]{fixmebold}
23     \ignorespaces#2 \fxnotename{#1}: \ignorespaces%
24 }
25
26 \renewcommand*\FXEnvLayoutPlainEnd[2]{\endBlockClass}
27
28 \renewcommand*\FXEnvLayoutSignatureBegin[2]{%
29     \BlockClass[\FXFaceEnvHTMLStyle]{fixmebold}
30     \fxnotename{#1}: \ignorespaces%
31 }
32
33 \renewcommand*\FXEnvLayoutSignatureEnd[2]{\@fxsignature{#2}\endBlockClass}
34
35 \def\FXFaceSignatureHTMLStyle{font-style:italic}
36
37 \DeclareRobustCommand*\@fxsignature[1]{%
38     \ifthenelse{\equal{#1}{}}%
39         {}%
40         { -- {\InlineClass[\FXFaceSignatureHTMLStyle]{fixmesignature}{#1}}}%
41 }
42
43
44 \def\FXFaceTargetHTMLStyle{font-style:italic}
45
46 \renewcommand\FXTargetLayoutPlain[2]{%
47     \InlineClass[\FXFaceTargetHTMLStyle]{fixmetarget}{#2}%
48 }
49
50 }% \AtBeginDocument
```

File 166   **`lwarp-fixmetodonotes.sty`**

§ 278   Package   # fixmetodonotes

(*Emulates or patches code by* GIOELE BARABUCCI.)

fixmetodonotes (*Pkg*)   fixmetodonotes is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{fixmetodonotes}[2013/04/28]

```
2 \VerifyCommand[lwarp][fixmetodonotes]{\NOTES@addtolist}{C8CA636EF295D370F26A278FFAE28B2F}
3
4 \renewcommand{\NOTES@addtolist}[2]{%
5     \refstepcounter{NOTES@note}%
6 %    \phantomsection% REMOVED
7     \addcontentsline{notes}{NOTES@note}{%
8         \protect\numberline{\theNOTES@note}{{#1}: {#2}}%
9     }%
10 }
11
12 \VerifyCommand[lwarp][fixmetodonotes]{\NOTES@marker}{B5B482E83AB149A1B7F0CCFB4099C61E}
13
14 \renewcommand{\NOTES@marker}[2]{\fbox{%
15     \textcolor{#2}{% WAS \color
```

```
16        \textbf{#1}}%
17    }}
18
19 \VerifyCommand[lwarp][fixmetodonotes]{\NOTES@colorline}{816FF1D31286EA48258FE3F2BA58E99C}
20
21 \renewcommand{\NOTES@colorline}[2]{%
22   \bgroup%
23     \ULon{\LWR@backgroundcolor{#1}{#2}}%
24 }
```

File 167 **`lwarp-flafter.sty`**

§ 279　Package **flafter**

flafter (*Pkg*)　flafter is ignored.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{flafter}[2018/01/08]
2 \providecommand\fl@trace[1]{}
```

File 168 **`lwarp-flippdf.sty`**

§ 280　Package **flippdf**

flippdf (*Pkg*)　flippdf is ignored.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{flippdf}[2006/06/30]
```

```
2 \newcommand\FlipPDF{}
3 \newcommand\UnFlipPDF{}
```

File 169 **`lwarp-float.sty`**

§ 281　Package **float**

(*Emulates or patches code by* ANSELM LINGNAU.)

float (*Pkg*)　float is emulated.

Float styles boxed and ruled are emulated by css and a float class according to style.

The HTML <figure> class is set to the float type, so css may also be used to format the float and its caption, according to float type. Furthermore, an additional class is set to the float style: plain, plaintop, boxed, or ruled, so css may be used to format by float style as well. Default formatting by css is provided for ruled and boxed styles.

⚠　not seem to be a floating environment　Always declare a \newfloat before modifying it with \floatname, etc.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{float}[2001/11/08]
```

\LWR@floatstyle   The default float style.

```
2 \newcommand*{\LWR@floatstyle}{plain}
```

\newfloat   {⟨*1: type*⟩} {⟨*2: placement*⟩} {⟨*3: ext*⟩} [⟨*4: within*⟩]

Emulates the \newfloat command from the float package.

"placement" is ignored.

```
3 \NewDocumentCommand{\newfloat}{m m m o}{%
4     \IfValueTF{#4}%
5         {\DeclareFloatingEnvironment[fileext=#3,within=#4]{#1}}%
6         {\DeclareFloatingEnvironment[fileext=#3]{#1}}%
```

Remember the float style:

```
7     \csedef{LWR@floatstyle@#1}{\LWR@floatstyle}%
```

newfloat package automatically creates the \listof command for new floats, but float does not, so remove \listof here in case it is manually created later.

```
8     \cslet{listof#1s}\relax%
9     \cslet{listof#1es}\relax%
```

Likesize, newfloat also creates \l@<type>, but float does not, so remove it here:

```
10     \cslet{l@#1}\relax%
11 }
```

\floatname   {⟨*type*⟩} {⟨*name*⟩}

Sets the text name of the float, such as "Figure". Avoids trying to set a recursive name, from trivfloat.

```
12 \NewDocumentCommand{\floatname}{m +m}{%
13     \def\LWR@tempone{#2}%
14     \def\LWR@temptwo{\@nameuse{#1name}}%
15     \ifdefequal{\LWR@tempone}{\LWR@temptwo}{}{%
16         \SetupFloatingEnvironment{#1}{name=#2}%
17     }%
18 }
```

\floatplacement   {⟨*type*⟩} {⟨*placement*⟩}

Float placement is ignored.

```
19 \newcommand*{\floatplacement}[2]{%
20     \SetupFloatingEnvironment{#1}{placement=#2}%
21 }
```

\floatstyle   {⟨*style*⟩}

Remember the style for future floats:

```
22 \newcommand{\floatstyle}[1]{%
23     \def\LWR@floatstyle{#1}%
24 }%
```

\restylefloat   * {⟨*type*⟩}

Remember the style for this float:

```
25 \NewDocumentCommand{\restylefloat}{s m}{%
26     \csedef{LWR@floatstyle@#2}{\LWR@floatstyle}%
27 }
```

\listof   See section 80.2 for the \LWR@listof command in the lwarp core.

```
28 \newcommand{\listof}{\LWR@listof}
```

File 170   **lwarp-floatflt.sty**

§ 282   Package   **floatflt**

(*Emulates or patches code by* MATS DAHLGREN.)

floatflt (*Pkg*)   floatflt is emulated.

for HTML output: Discard all options for lwarp-floatflt:

```
1 \LWR@ProvidesPackageDrop{floatflt}[1997/07/16]
```

Env   [⟨⟩]   offset {⟨*type*⟩} {⟨*width*⟩} Borrowed from the lwarp version of keyfloat:

```
2 \NewDocumentEnvironment{KFLTfloatflt@marginfloat}{O{-1.2ex} m m}
3 {%
4     \begin{LWR@setvirtualpage}*%
5     \ifblank{#3}{%
6         \LWR@BlockClassWP{%
7             float:right; %
8             width: 1.5in; % reasonable dummy width for word processor
9             margin:10pt%
10        }{}%
11        (note)%
12        {marginblock}%
13    }{%
14        \setlength{\LWR@templengthone}{#3}%
15        \LWR@BlockClassWP{%
16            float:right; %
17            width:\LWR@printlength{\LWR@templengthone};  % extra space
18            margin:10pt%
19        }{%
20            width:\LWR@printlength{\LWR@templengthone}%
21        }%
22        (note)%
23        {marginblock}%
24    }%
25    \renewcommand*{\@captype}{#2}%
26 }
27 {%
28    \endLWR@BlockClassWP%
29    \end{LWR@setvirtualpage}%
30 }
```

Env   floatingfigure   [⟨*placement*⟩] {⟨*width*⟩}

```
31 \DeclareDocumentEnvironment{floatingfigure}{o m}
32   {\begin{KFLTfloatflt@marginfloat}{figure}{#2}}
33   {\end{KFLTfloatflt@marginfloat}}
```

Env   floatingtable   [⟨*placement*⟩]

```
34 \DeclareDocumentEnvironment{floatingtable}{o}
35   {\begin{KFLTfloatflt@marginfloat}{table}{}}
36   {\end{KFLTfloatflt@marginfloat}}
```

File 171   **`lwarp-floatpag.sty`**

§ 283   Package   # floatpag

(*Emulates or patches code by* VYTAS STATULEVIČIUS AND SIGITAS TOLUŠIS.)

floatpag (*Pkg*)   floatpag is ignored.

for HTML output: Discard all options for lwarp-floatpag:

```
1 \LWR@ProvidesPackageDrop{floatpag}[2012/05/29]
```

```
2 \newcommand*{\floatpagestyle}[1]{}
3 \newcommand*{\rotfloatpagestyle}[1]{}
4 \newcommand*{\thisfloatpagestyle}[1]{}
```

File 172   **`lwarp-floatrow.sty`**

§ 284   Package   # floatrow

(*Emulates or patches code by* OLGA LAPKO.)

floatrow (*Pkg*)   floatrow is emulated.

for HTML output: `1 \LWR@ProvidesPackageDrop{floatrow}[2008/08/02]`

⚠ Misplaced alignment   Use \StartDefiningTabulars and \StopDefiningTabulars before and after defin-
tab character &   ing macros using \ttabbox with a tabular inside. See section 8.10.1.

⚠   subfig package   When combined with the subfig package, while inside a subfloatrow \ffigbox
and \ttabbox must have the caption in the first of the two of the mandatory
arguments.

The emulation of **floatrow** does not support \FBwidth or \FBheight. These values
⚠   \FBwidth, \FBheight   are pre-set to .3\linewidth and 2in. Possible solutions include:

- Use fixed lengths. lwarp will scale the HTML lengths appropriately.

- Use warpprint and warpHTML environments to select appropriate values for
  each case.

- Inside a warpHTML environment, manually change \FBwidth or \FBheight
  before the \ffigbox or \ttabbox. Use \FBwidth or \FBheight normally after-
  wards; it will be used as expected in print output, and will use your custom-
  selected value in HTML output. This custom value will be used repeatedly,
  until it is manually changed to a new value.

After everything has loaded, remember whether subcaption was loaded. If not, it is assumed that subfig is used instead:

```
2 \newbool{LWR@subcaptionloaded}
3
4 \AtBeginDocument{
5 \IfPackageLoadedTF{subcaption}
6     {\booltrue{LWR@subcaptionloaded}}
7     {\boolfalse{LWR@subcaptionloaded}}
8 }
```

\floatbox
    [⟨*1 preamble*⟩] {⟨*2 captype*⟩} [⟨*3 width*⟩] [⟨*4 height*⟩] [⟨*5 vert pos*⟩] {⟨*6 caption*⟩} {⟨*7 object*⟩}

Only parameters for captype, width, caption, and object are used.

LWR@insubfloatrow is true if inside a subfloatrow environment.

There are two actions, depending on the use of subcaption or subfig.

```
9 \NewDocumentCommand{\floatbox}{o m o o o +m +m}{%
10 \ifbool{LWR@subcaptionloaded}%
11 {% subcaption
```

For subcaption:

```
12     \ifbool{LWR@insubfloatrow}%
13     {% subcaption in a subfloatrow
```

subfigure and subtable environments take width as an argument.

```
14         \IfValueTF{#3}%
15         {\@nameuse{sub#2}{#3}}%
16         {\@nameuse{sub#2}{\linewidth}}%
17     }% subcaption in a subfloatrow
18     {% subcaption not in subfloatrow
```

figure and table environments do not take a width argument.

```
19         \@nameuse{#2}%
20     }% subcaption not in subfloatrow
21     #6
22
23     #7
```

End the environments:

```
24     \ifbool{LWR@insubfloatrow}%
25     {\@nameuse{endsub#2}}%
26     {\@nameuse{end#2}}%
27 }% subcaption
28 {% assume subfig
```

For subfig:

```
29 \ifbool{LWR@insubfloatrow}%
30 {% subfig in a subfloatrow
```

\subfloat is a macro, not an environment.

Package subfig's \subfloat command takes an optional argument which is the caption, but \floatbox argument #6 contains commands to create the caption and label, not the caption itself. Thus, \caption is temporarily disabled to return its own argument without braces.

```
31     \begingroup
```

```
32      \let\caption\@firstofone
33      \subfloat[#6]{#7}
34      \endgroup
35 }% subfig in a subfloatrow
36 {% subfig package, but not a subfig
```

figure and table are environments:

```
37 \@nameuse{#2}
38 #6
39
40 #7
41 \@nameuse{end#2}
42 }% subfig package, but not a subfig
43 }% assume subfig
44 }
```

Not used:

```
45 \newcommand*{\nocapbeside}{}
46 \newcommand*{\capbeside}{}
47 \newcommand*{\captop}{}
48 \newlength{\FBwidth}
49 \setlength{\FBwidth}{.3\linewidth}
50 \newlength{\FBheight}
51 \setlength{\FBheight}{2in}
52 \newcommand*{\useFCwidth}{}
53 \newcommand{\floatsetup}[2][]{}
54 \newcommand{\thisfloatsetup}[1]{}
55 \newcommand{\clearfloatsetup}[1]{}
56 \newcommand*{\killfloatstyle}{}
```

\newfloatcommand   {⟨*1 command*⟩} {⟨*2 captype*⟩} [⟨*3 preamble*⟩] [⟨*4 default width*⟩]

Preamble and default width are ignored.

```
57 \NewDocumentCommand{\newfloatcommand}{m m o o}{%
58 \@namedef{#1}{
59 \floatbox{#2}
60 }
61 }
```

\renewfloatcommand   {⟨*1 command*⟩} {⟨*2 captype*⟩} [⟨*3 preamble*⟩] [⟨*4 default width*⟩]

Preamble and default width are ignored.

```
62 \NewDocumentCommand{\renewfloatcommand}{m m o o}{%
63 \@namedef{#1}{%
64 \floatbox{#2}
65 }
66 }
```

\ffigbox   [⟨*width*⟩] [⟨*height*⟩] [⟨*vposn*⟩] {⟨*caption commands*⟩} {⟨*contents*⟩}

```
67 \newfloatcommand{ffigbox}{figure}[\nocapbeside][]
```

\ttabbox   [⟨*width*⟩] [⟨*height*⟩] [⟨*vposn*⟩] {⟨*caption commands*⟩} {⟨*contents*⟩}

```
68 \newfloatcommand{ttabbox}{table}[\captop][\FBwidth]
```

\fcapside   [⟨*width*⟩] [⟨*height*⟩] [⟨*vposn*⟩] {⟨*caption commands*⟩} {⟨*contents*⟩}

```
69 \newfloatcommand{fcapside}{figure}[\capbeside][]
```

Env floatrow   [⟨*numfloats*⟩]

The row of floats is placed into a `<div>` of class floatrow.

```
70 \newenvironment*{floatrow}[1][2]
71 {%
72     \begin{LWR@setvirtualpage}*%
73     \BlockClass{floatrow}%
74 }
75 {
76     \endBlockClass%
77     \end{LWR@setvirtualpage}%
78 }
```

Keys for \DeclareNewFloatType:

```
79 \newcommand*{\LWR@frowkeyplacement}{}
80 \newcommand*{\LWR@frowkeyname}{}
81 \newcommand*{\LWR@frowkeyfileext}{}
82 \newcommand*{\LWR@frowkeywithin}{}
83 \newcommand*{\LWR@frowkeycapstyle}{}
84
85 \define@key{frowkeys}{placement}{}%
86 \define@key{frowkeys}{name}{\renewcommand{\LWR@frowkeyname}{#1}}%
87 \define@key{frowkeys}{fileext}{\renewcommand{\LWR@frowkeyfileext}{#1}}%
88 \define@key{frowkeys}{within}{\renewcommand{\LWR@frowkeywithin}{#1}}%
89 \define@key{frowkeys}{relatedcapstyle}{}%
```

\DeclareNewFloatType   {⟨*type*⟩} {⟨*options*⟩}

Use \listof{type}{Title} to print a list of the floats.

```
90 \newcommand*{\DeclareNewFloatType}[2]{%
```

Reset key values:

```
91 \renewcommand*{\LWR@frowkeyplacement}{}%
92 \renewcommand*{\LWR@frowkeyname}{}%
93 \renewcommand*{\LWR@frowkeyfileext}{}%
94 \renewcommand*{\LWR@frowkeywithin}{}%
95 \renewcommand*{\LWR@frowkeycapstyle}{}%
```

Read new key values:

```
96 \LWR@traceinfo{about to setkeys frowkeys}%
97 \setkeys{frowkeys}{#2}%
98 \LWR@traceinfo{finished setkeys frowkeys}%
```

Create a new float with optional [within]:

```
99 \ifthenelse{\equal{\LWR@frowkeywithin}{}}%
100 {%
101     \DeclareFloatingEnvironment[
102         placement=\LWR@frowkeyplacement,
103         fileext=\LWR@frowkeyfileext
104     ]{#1}%
105 }%
106 {%
107     \DeclareFloatingEnvironment[
```

```
108         placement=\LWR@frowkeyplacement,
109         fileext=\LWR@frowkeyfileext,
110         within=\LWR@frowkeywithin
111     ]{#1}%
112 %       \LWR@traceinfo{finished newfloat #1}%
113 }%
```

Rename the float if a name was given:

```
114 \ifthenelse{\equal{\LWR@frowkeyname}{}}%
115     {}%
116     {%
117         \SetupFloatingEnvironment{#1}{name={\LWR@frowkeyname}}%
118     }%
119 }
```

Not used:

```
120 \newcommand{\buildFBBOX}[2]{}
121 \newcommand*{\CenterFloatBoxes}{}
122 \newcommand*{\TopFloatBoxes}{}
123 \newcommand*{\BottomFloatBoxes}{}
124 \newcommand*{\PlainFloatBoxes}{}
125
126 \newcommand{\capsubrowsettings}{}
127
128 \NewDocumentCommand{\RawFloats}{o o}{}
```

\RawCaption   {⟨*text*⟩}

To be used inside a minipage or parbox.

```
129 \newcommand{\RawCaption}[1]{#1}
```

\floatfoot   {⟨*text*⟩}

Places additional text inside a float, inside a css <div> of class floatfoot.

```
130 \NewDocumentCommand{\floatfoot}{s +m}{%
131     \begin{BlockClass}{floatfoot}
132     #2
133     \end{BlockClass}
134 }
```

Used to compute \linewidth.

```
135 \newbool{LWR@insubfloatrow}
136 \boolfalse{LWR@insubfloatrow}
```

Env   subfloatrow   [⟨*num_floats*⟩]

```
137 \newenvironment*{subfloatrow}[1][2]
138 {
```

The row of floats is placed into a <div> of class floatrow:

```
139     \LWR@forcenewpage
140     \BlockClass{floatrow}
```

While inside the floatrow, LWR@insubfloatrow is set true, which tells \floatbox to use \subfigure or \subtable.

```
141     \begingroup%
142     \booltrue{LWR@insubfloatrow}%
143 }
144 {%
145     \endgroup%
146     \endBlockClass%
147     \boolfalse{LWR@insubfloatrow}%
148 }
```

File 173 **lwarp-fltrace.sty**

§ 285   Package   **fltrace**

fltrace (*Pkg*)   fltrace is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{fltrace}[2018/01/08]

2 \def\tracefloats{}
3 \def\tracefloatsoff{}
4 \def\tracefloatvals{}

File 174 **lwarp-flushend.sty**

§ 286   Package   **flushend**

(*Emulates or patches code by* Sigitas Tolušis.)

flushend (*Pkg*)   flushend is ignored.

for HTML output: Discard all options for lwarp-flushend:

1 \LWR@ProvidesPackageDrop{flushend}[2021/10/04]

2 \newcommand*{\flushend}{}
3 \newcommand*{\raggedend}{}
4 \newcommand*{\flushcolsend}{}
5 \newcommand*{\raggedcolsend}{}
6 \newtoks\atColsBreak \atColsBreak={}
7 \newtoks\atColsEnd   \atColsEnd={}
8 \newcommand*{\showcolsendrule}{}

File 175 **lwarp-fnbreak.sty**

§ 287   Package   **fnbreak**

fnbreak (*Pkg*)   fnbreak is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{fnbreak}[2012/01/01]

2 \newcommand*{\fnbreakverbose}{}
3 \newcommand*{\fnbreaknonverbose}{}
4 \newcommand*{\fnbreaklabel}{}
5 \newcommand*{\fnbreaknolabel}{}

File 176   `lwarp-fncychap.sty`

§ 288   Package   # fncychap

(*Emulates or patches code by* ULF A. LINDGREN.)

fncychap (*Pkg*)   fncychap is ignored.

for HTML output: Discard all options for lwarp-fncychap:

```
1 \LWR@ProvidesPackageDrop{fncychap}[2007/07/30]
```

```
2 \def\mghrulefill#1{}
3 \def\ChNameLowerCase{}
4 \def\ChNameUpperCase{}
5 \def\ChNameAsIs{}
6 \def\ChTitleLowerCase{}
7 \def\ChTitleUpperCase{}
8 \def\ChTitleAsIs{}
9 \newcommand{\ChRuleWidth}[1]{}
10 \newcommand{\ChNameVar}[1]{}
11 \newcommand{\ChNumVar}[1]{}
12 \newcommand{\ChTitleVar}[1]{}
13 \newcommand{\TheAlphaChapter}{}
14 \newcommand{\DOCH}{}
15 \newcommand{\DOTI}[1]{}
16 \newcommand{\DOTIS}[1]{}
17 \newlength{\mylen}
18 \newlength{\myhi}
19 \newlength{\px}
20 \newlength{\py}
21 \newlength{\pyy}
22 \newlength{\pxx}
23 \newlength{\RW}
24 \newcommand{\FmN}[1]{#1}
25 \newcommand{\FmTi}[1]{#1}
```

File 177   `lwarp-fnlineno.sty`

§ 289   Package   # fnlineno

fnlineno (*Pkg*)   fnlineno is ignored.

for HTML output: ```
1 \LWR@ProvidesPackageDrop{fnlineno}[2011/01/07]
```

File 178   `lwarp-fnpara.sty`

§ 290   Package   # fnpara

fnpara (*Pkg*)   fnpara is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{fnpara}

File 179   **`lwarp-fnpos.sty`**

§ 291   Package   # fnpos

(*Emulates or patches code by* HIROSHI NAKASHIMA.)

fnpos (*Pkg*)   fnpos is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{fnpos}[1999/07/14]

2 \newcommand*{\makeFNbottom}{}
3 \newcommand*{\makeFNmid}{}
4 \newcommand*{\makeFNbelow}{}
5 \newcommand*{\makeFNabove}{}

File 180   **`lwarp-fontawesome.sty`**

§ 292   Package   # fontawesome

(*Emulates or patches code by* XAVIER DANAUX.)

fontawesome (*Pkg*)   fontawesome is patched for use by lwarp.

Hashed inline images are used, as there may not be Unicode support for all icons.

⚠ *poppler* syntax warning   If using PDF LATEX, *poppler* may issue a syntax warning regarding parsing a ligature component. XƎLATEX or LuaLATEX may be used to avoid this warning.

In the following, the general strategy is to intercept \symbol and embed it inside a lateximage. These changes are done inside a local group.

For PDF LATEX, the alt tag includes the icon (symbol) number. For XƎLATEX and LuaLATEX, the alt tag is generic.

for HTML output: 1 \LWR@ProvidesPackagePass{fontawesome}[2016/05/15]

2 \LetLtxMacro\LWR@orig@symbol\symbol
3
4 \ifxetexorluatex
5
6 \newfontfamily{\LWR@orig@FA}{FontAwesome}
7
8 \newcommand*{\LWR@fontawesome@xelatex@symbol}[1]{%
9   \LWR@findcurrenttextcolor%
10   \begin{lateximage}*[icon]%
11   [fontawesomexetex#1SZ\LWR@font@size{}CL\LWR@tempcolor]%
12   \csuse{\LWR@font@size}%
13   \LWR@orig@FA%
14   \LWR@orig@symbol{#1}%
15   \end{lateximage}%
16 }
17

```
18 \RenewDocumentCommand{\FA}{}{%
19     \LetLtxMacro\symbol\LWR@fontawesome@xelatex@symbol%
20 }
21
22 \else
23
24 \newcommand*{\LWR@fontawesome@symbolX}[2]{%
25     \LWR@findcurrenttextcolor%
26     \begin{lateximage}*[icon #1]?%
27         [fontawesome#2#1SZ\LWR@font@size{}CL\LWR@tempcolor]%
28     \csuse{\LWR@font@size}%
29     \fontencoding{U}\fontfamily{fontawesome#2}\selectfont%
30     \LWR@orig@symbol{#1}%
31     \end{lateximage}%
32 }
33
34 \newcommand*{\LWR@fontawesome@symbolone}[1]{%
35     \LWR@fontawesome@symbolX{#1}{one}%
36 }
37
38 \newcommand*{\LWR@fontawesome@symboltwo}[1]{%
39     \LWR@fontawesome@symbolX{#1}{two}%
40 }
41
42 \newcommand*{\LWR@fontawesome@symbolthree}[1]{%
43     \LWR@fontawesome@symbolX{#1}{three}%
44 }
45
46 \renewrobustcmd\FAone{%
47     \LetLtxMacro\symbol\LWR@fontawesome@symbolone%
48 }
49
50 \renewrobustcmd\FAtwo{%
51     \LetLtxMacro\symbol\LWR@fontawesome@symboltwo%
52 }
53
54 \renewrobustcmd\FAthree{%
55     \LetLtxMacro\symbol\LWR@fontawesome@symbolthree%
56 }
57 \fi
```

File 181   `lwarp-fontawesome5.sty`

§ 293   Package   # fontawesome5

(*Emulates or patches code by* MARCEL KRÜGER.)

fontawesome5 (*Pkg*)   **fontawesome5** is patched for use by lwarp.

Hashed inline images are used, as there may not be Unicode support for all icons.

The alt tag has the name of the icon.

for HTML output: 1 `\LWR@ProvidesPackagePass{fontawesome5}[2022/05/02]`

This used to contain code, but now it is split into the related two packages.

File 182   `lwarp-fontawesome5-generic-helper.sty`

§ 294   Package   # fontawesome5-generic-helper

(*Emulates or patches code by* MARCEL KRÜGER.)

fontawesome5-generic-helper   fontawesome5-generic-helper is patched for use by lwarp.
(*Pkg*)

Hashed inline images are used, as there may not be Unicode support for all icons.

The alt tag has the name of the icon.

for HTML output: 1 `\LWR@ProvidesPackagePass{fontawesome5-generic-helper}[2022/05/02]`

```
2 \ExplSyntaxOn
3
4 \VerifyCommand[lwarp][fontawesome5-generic-helper]{\fontawesome_use_icon:nn}
5     {0260A9C94303C43957AAEBEA2B4D3DB1}
6
7 \cs_set:Nn\fontawesome_use_icon:nn{
8     \LWR@findcurrenttextcolor
9   \cs_if_exist:cTF{c__fontawesome_slot_#2_tl}{
10     \begin{lateximage}*[#2]?%   lwarp
11        [fontawesome5#1SZ\LWR@font@size{}CL\LWR@tempcolor]%     lwarp
12     \csuse{\LWR@font@size}%       lwarp
13     \bool_if:NTF\c__fontawesome_fixed_bool{
14       \makebox[1.5em][c]
15     }{
16       \use:n
17     }
18     {
19       \exp_last_unbraced:Nv
20         \__fontawesome_icon_at:nnnn
21         {c__fontawesome_slot_#2_tl}
22           {#1}{#2}
23     }
24     \end{lateximage}%   lwarp
25   }{
26     \msg_error:nnxx{fontawesome5}{icon-not-found}{#2}{#1}
27   }
28 }
29 \ExplSyntaxOff
```

File 183   `lwarp-fontawesome5-utex-helper.sty`

§ 295   Package   # fontawesome5-utex-helper

(*Emulates or patches code by* MARCEL KRÜGER.)

fontawesome5-utex-helper   fontawesome5-utex-helper is patched for use by lwarp.
(*Pkg*)

Hashed inline images are used, as there may not be Unicode support for all icons.

The alt tag has the name of the icon.

```
1 \LWR@ProvidesPackagePass{fontawesome5-utex-helper}[2022/05/02]

2 \ExplSyntaxOn
3
4 \VerifyCommand[lwarp][fontawesome5-utex-helper]{\fontawesome_use_icon:nn}
5   {8452FF2BF0A317552B0920628ADD8C18}
6
7 \cs_set:Nn\fontawesome_use_icon:nn{
8   \group_begin:
9     \LWR@findcurrenttextcolor
10    \usefont
11      {TU}
12      {fontawesome\c__fontawesome_kind_tl}
13      {#1}
14      {n}
15    \bool_set:Nn \l__fontawesome_duotone_bool {
16      \str_if_eq_p:ee {#1} {duotone}
17    }
18    \int_set:Nn\l_tmpa_int{
19      \__fontawesome_glyphindex:n{
20        #2
21        \bool_if:NT \l__fontawesome_duotone_bool { -primary }
22      }
23    }
24    \int_compare:nNnT{\l_tmpa_int}={0}{
25      \fontseries{solid}
26      \selectfont
27      \bool_set_false:N \l__fontawesome_duotone_bool
28      \int_set:Nn\l_tmpa_int{\__fontawesome_glyphindex:n{#2}}
29      \int_compare:nNnTF{\l_tmpa_int}={0}{
30        \fontfamily{fontawesomebrands}
31        \fontseries{regular}
32        \selectfont
33        \int_set:Nn\l_tmpa_int{\__fontawesome_glyphindex:n{#2}}
34        \int_compare:nNnT{\l_tmpa_int}={0}{
35          \msg_error:nnxx{fontawesome5}{icon-not-found}{#2}{#1}
36        }
37      }{
38        \msg_warning:nnxx{fontawesome5}{style-substitution}{#2}{#1}
39      }
40    }
41    \begin{lateximage}*[#2]?%    lwarp
42      [fontawesome5#1SZ\LWR@font@size{}CL\LWR@tempcolor]%        lwarp
43    \csuse{\LWR@font@size}%        lwarp
44    \bool_if:NTF\c__fontawesome_fixed_bool{
45      \makebox[1.5em][c]
46    }{
47      \use:n
48    }
49    {
50      \bool_if:NTF \l__fontawesome_duotone_bool {
51        \__fontawesome_glyph:w \l_tmpa_int
52       \int_set:Nn\l_tmpa_int{ \__fontawesome_glyphindex:n{ #2-secondary } }
53        \int_compare:nNnF { \l_tmpa_int } = { 0 } {
54          \llap {
55            \l_fontawesome_duotone_secondary_style_tl
56            { \__fontawesome_glyph:w \l_tmpa_int }
57          }
```

```
58        }
59        % \ooalign {
60        %   \hss \__fontawesome_glyph:w \l_tmpa_int \hss \cr
61        %   \hss
62   %   \int_set:Nn\l_tmpa_int{ \__fontawesome_glyphindex:n{ #2-secondary } } }
63        %   \int_compare:nNnF { \l_tmpa_int } = { 0 } {
64        %     \color{gray}\__fontawesome_glyph:w \l_tmpa_int
65        %   }
66        %   \hss \cr
67        % }
68      } {
69        \__fontawesome_glyph:w \l_tmpa_int
70      }
71    }
72    \end{lateximage}% lwarp
73  \group_end:
74 }
75 \ExplSyntaxOff
```

File 184 **`lwarp-fontaxes.sty`**

§ 296 Package # fontaxes

(*Emulates or patches code by* ANDREAS BÜHMANN, MICHAEL UMMELS.)

fontaxes (*Pkg*) fontaxes is emulated for HTML, and used as-is for print output.

Functionality for small caps is in the lwarp core. Swashes and figure styles are ignored for HTML.

for HTML output: 1 `\LWR@ProvidesPackagePass{fontaxes}[2014/03/23]`

```
2 \ifdef{\LWR@HTML@swshape}{}{% duplicated by nfssext-cfr
3    \newcommand{\LWR@HTML@swshape}{}
4    \LWR@formatted{swshape}
5
6    \newrobustcmd{\LWR@HTML@textsw}[1]{#1}
7    \LWR@formatted{textsw}
8
9    \FilenameNullify{%
10       \LetLtxMacro\swshape\@empty%
11       \LetLtxMacro\textsw\firstofone%
12   }
13 }
```

File 185 **`lwarp-fontenc.sty`**

§ 297 Package # fontenc

fontenc (*Pkg*) If using PDF LATEX, lwarp used to require fontenc be loaded before lwarp, but now lwarp itself loads \fontenc with T1 encoding, which lwarp requires. fontenc is now allowed to be loaded with another encoding after lwarp.

lwarp-fontenc is no longer necessary, but is still provided to overwrite older versions.

1 \LWR@ProvidesPackagePass{fontenc}[2017/04/05]

File 186   **`lwarp-footmisc.sty`**

§ 298   Package   **footmisc**

(*Emulates or patches code by* ROBIN FAIRBAIRNS.)

footmisc (*Pkg*)   **footmisc** is emulated.

lwarp incidentally happens to emulate the stable option.

1 \LWR@ProvidesPackageDrop{footmisc}[2011/06/06]

Some nullified commands:

```
2 \newcommand{\footnotelayout}{}
3 \newcommand{\setfnsymbol}[1]{}
4 \NewDocumentCommand{\DefineFNsymbols}{s m o m}{}
5
6 \newdimen\footnotemargin
7 \footnotemargin1.8em\relax
8
9 \newcommand*\hangfootparskip{0.5\baselineskip}
10 \newcommand*\hangfootparindent{0em}%
11
12 \let\pagefootnoterule\footnoterule
13 \let\mpfootnoterule\footnoterule
14 \def\splitfootnoterule{\kern-3\p@ \hrule \kern2.6\p@}
15
16 \providecommand*{\multiplefootnotemarker}{3sp}
17 \providecommand*{\multfootsep}{,}
```

Using **cleveref**. \labelcref only prints the number of the object, not its type.

18 \providecommand*{\footref}[1]{\labelcref{#1}}

The following work as-is:

```
19 \newcommand\mpfootnotemark{%
20   \@ifnextchar[%
21     \@xmpfootnotemark%
22     {%
23       \stepcounter\@mpfn%
24       \protected@xdef\@thefnmark{\thempfn}%
25       \@footnotemark%
26     }%
27 }
28 \def\@xmpfootnotemark[#1]{%
29   \begingroup%
30     \csname c@\@mpfn\endcsname #1\relax%
31     \unrestored@protected@xdef\@thefnmark{\thempfn}%
32   \endgroup%
33   \@footnotemark%
34 }
```

File 187 **`lwarp-footnote.sty`**

§ 299   Package **footnote**

(*Emulates or patches code by* Mark Wooding.)

footnote (*Pkg*)   **footnote** is used with minor patches.

for HTML output: **footnote** patches \@makefntext in a strange way. It must be restored to the expected defintion before loading **footnote**, then replaced again after.

```
1 \long\def\@makefntext#1{\textsuperscript{\@thefnmark}~#1}
2
3 \LWR@ProvidesPackagePass{footnote}[1997/01/28]
4
5 \long\def\@makefntext#1{\textsuperscript{\@thefnmark}~{#1}}
```

```
6 \VerifyCommand[lwarp][footnote]{\spewnotes}{BCC4919F5404BADA8F1CF486E5709072}
7
8 \def\spewnotes{%
9    \endgroup%
10   \if@savingnotes\else\ifvoid\fn@notes\else\begingroup%
11     \let\@makefntext\@empty%
12     \let\@finalstrut\@gobble%
13     \let\rule\@gobbletwo%
14     \booltrue{LWR@spewingnotes}%          lwarp
15     \@footnotetext{\unvbox\fn@notes}%
16   \endgroup\fi\fi%
17 }
18 \let\endsavenotes\spewnotes
19
20 \VerifyCommand[lwarp][footnote]{\fn@fntext}{4C750987515F28FE665A08AB710193BA}
21
22 \def\fn@fntext#1{%
23   \ifx\ifmeasuring@\@@undefined%
24     \expandafter\@secondoftwo\else\expandafter\@iden%
25   \fi%
26   {\ifmeasuring@\expandafter\@gobble\else\expandafter\@iden\fi}%
27   {%
28     \global\setbox\fn@notes\vbox{%
29       \unvbox\fn@notes%
30       \LWR@htmltagc{\LWR@tagregularparagraph}%     lwarp
31       \LWR@orignewline%                            lwarp
32       \fn@startnote%
33       \@makefntext{%
34         \rule\z@\footnotesep%
35         \ignorespaces%
36         #1%
37         \@finalstrut\strutbox%
38       }%
39       \fn@endnote%
40     }%
41   }%
42 }
```

Removed print-version formatting:

```
43 \VerifyCommand[lwarp][footnote]{\fn@startnote}{D101A3D1B9653A6FDD7E9CF37BD5A4DD}
44
45 \def\fn@startnote{%
46 %    \@parboxrestore%
47   \protected@edef\@currentlabel{\csname p@\@mpfn\endcsname\@thefnmark}%
48 %    \color@begingroup% *** conflicts with lwarp
49 }
50
51 % \let\fn@endnote\color@endgroup% *** conflicts with lwarp
52 \def\fn@endnote{%
53     \LWR@orignobreakspace\LWR@orignewline%
54     \LWR@htmltagc{/\LWR@tagregularparagraph}\LWR@orignewline%
55     \LWR@orignobreakspace\LWR@orignewline%
56 }
```

Removed print-version formatting:

```
57 \VerifyCommand[lwarp][footnote]{\fn@startfntext}{7270AD27C28391C41DA1FE47C49B5E7A}
58
59 \def\fn@startfntext{%
60   \setbox\z@\vbox\bgroup%
61     \LWR@htmltagc{\LWR@tagregularparagraph}%     lwarp
62     \LWR@orignewline%                            lwarp
63     \fn@startnote%
64     \fn@prefntext% Req'd for numbering.
65 %      \rule\z@\footnotesep%
66     \ignorespaces%
67 }
68
```

Removed print-version formatting, added closing paragraph tag:

```
69 \VerifyCommand[lwarp][footnote]{\fn@endfntext}{17BC1D2CD9A84BAFFBE765CC1618C36D}
70
71 \def\fn@endfntext{%
72     \fn@postfntext%
73     \LWR@orignobreakspace\LWR@orignewline%
74     \LWR@htmltagc{/\LWR@tagregularparagraph}%
75     \LWR@orignewline%
76   \egroup%
77   \begingroup%
78     \let\@makefntext\@empty%
79     \let\@finalstrut\@gobble%

80     \LetLtxMacro\rule\@gobbletwo%
81     \booltrue{LWR@spewingnotes}%    lwarp
82     \@footnotetext{\unvbox\z@}%
83   \endgroup%
84 }
```

These have been redefined, so re-\let them again:

```
85 \let\endfootnote\fn@endfntext
86 \let\endfootnotetext\endfootnote
```

File 188 `lwarp-footnotebackref.sty`

§ 300 Package **footnotebackref**

footnotebackref (*Pkg*) footnotebackref is ignored.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{footnotebackref}[2012/07/01]
```

File 189 `lwarp-footnotehyper.sty`

§ 301 Package **footnotehyper**

footnotehyper (*Pkg*) footnotehyper is a hyperref-safe version of footnote. For lwarp, footnotehyper is emulated.

for HTML output: Discard all options for lwarp-footnotehyper:

```
1 \RequirePackage{footnote}
2
3 \LWR@ProvidesPackageDrop{footnotehyper}[2018/01/23]
```

File 190 `lwarp-footnoterange.sty`

§ 302 Package **footnoterange**

(*Emulates or patches code by* H.-Martin Münch.)

footnoterange (*Pkg*) footnoterange is patched for use by lwarp.

for HTML output:
```
1 \LWR@ProvidesPackagePass{footnoterange}[2012/02/17]
```

```
2 \csletcs{footnoterange}{footnoterange*}
3 \csletcs{endfootnoterange}{endfootnoterange*}
```

File 191 `lwarp-footnpag.sty`

§ 303 Package **footnpag**

footnpag (*Pkg*) footnpag is ignored.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{footnpag}
```

File 192 `lwarp-foreign.sty`

§ 304 Package **foreign**

(*Emulates or patches code by* Philip G. Ratcliffe.)

foreign (*Pkg*)   foreign is patched for use by lwarp.

for HTML output: 1 `\LWR@ProvidesPackagePass{foreign}[2012/09/25]`

2 `\renewcommand\foreignabbrfont{\emph}`

File 193   **`lwarp-forest.sty`**

§ 305   Package   # forest

(*Emulates or patches code by* SAŠO ŽIVANOVIĆ.)

forest (*Pkg*)   forest is patched for use by lwarp.

⚠   \Forest*   The starred version of the macro \Forest* is not supported. lwarp encases each lateximage in an environment, so the global results of the starred \Forest* are lost.

for HTML output: 1 `\LWR@ProvidesPackagePass{forest}[2017/07/14]`

```
2 \BeforeBeginEnvironment{forest}{%
3     \begin{lateximage}[-forest-~\PackageDiagramAltText]?%
4 }
5
6 \AfterEndEnvironment{forest}{\end{lateximage}}
7
8 \VerifyCommand[lwarp][forest]{\Forest}{D44A6D1EAFFC86653905CC666F563E6D}
9
10 \RenewDocumentCommand{\Forest}{s D(){} m}{%
11  \forest@config{#2}%
12  \IfBooleanTF{#1}{%
13      \PackageError{lwarp-forest}%
14      {\protect\Forest* is not supported}%
15      {Lwarp uses an environment for images,\MessageBreak
16          but \protect\Forest* cannot work in an environment.}%
17      \let\forest@next\forest@env%
18  }{\let\forest@next\forest@group@env}%
19      \begin{lateximage}[-forest-~\PackageDiagramAltText]?%     lwarp
20  \forest@next{#3}%
21      \end{lateximage}%                 lwarp
22 }
```

File 194   **`lwarp-fouridx.sty`**

§ 306   Package   # fouridx

(*Emulates or patches code by* STEFAN KARRMANN.)

fouridx (*Pkg*)   fouridx works as-is with SVG math, and is emulated for MATHJAX.

for HTML output: 1 `\LWR@ProvidesPackagePass{fouridx}[2013/11/21]`

```
2 \begin{warpMathJax}
3 \CustomizeMathJax{%
```

```
4    \newcommand{\fourIdx}[5]{%
5        \vphantom{#5}^{\hphantom{#2}#1}_{\hphantom{#1}#2}{#5}^{#3}_{#4}%
6    }%
7 }
8 \end{warpMathJax}
```

File 195 **`lwarp-fourier.sty`**

§ 307   Package   **fourier**

(*Emulates or patches code by* MICHEL BOVANI.)

fourier (*Pkg*)   fourier is used as-is for SVG math, and is emulated for MATHJAX.

⚠ limitations   The MATHJAX emulation ignores all package options, except sloped and upright are honored for Greek characters, but MATHJAX cannot yet honor these for Latin characters.

The dedicated macros for upright and italic Greek letters do work correctly.

SVG math should appear the same as the printed output.

for HTML output:
```
1 \LWR@ProvidesPackagePass{fourier}[2020/03/03]
2
3 \LWR@infoprocessingmathjax{fourier}

4 \LWR@origRequirePackage{lwarp-common-mathjax-letters}
5
6 \LWR@origRequirePackage{lwarp-common-mathjax-overlaysymbols}
7
8 \begin{warpMathJax}
9
10 \IfPackageLoadedWithOptionsTF{fourier}{sloped}
11    {
12        \LWR@mathjax@addgreek@l@up{other}{}
13        \LWR@mathjax@addgreek@u@it*{other}{}
14    }% sloped
15    {% not sloped
16        \IfPackageLoadedWithOptionsTF{fourier}{upright}
17            {% upright option
18                \LWR@mathjax@addgreek@l@up{}{}
19                \LWR@mathjax@addgreek@u@up*{}{}
20                \LWR@mathjax@addgreek@l@it{other}{}
21                \LWR@mathjax@addgreek@u@it*{other}{}
22            }
23            {% neither sloped nor upright
24                \LWR@mathjax@addgreek@l@up{other}{}
25                \LWR@mathjax@addgreek@u@it*{other}{}
26            }
27    }
28
29 \CustomizeMathJax{\newcommand{\othergreek}[1]{#1}}
30 \CustomizeMathJax{\let\varvarrho\varrho}
31 \CustomizeMathJax{\let\varvarpi\varpi}
32 \CustomizeMathJax{\let\othervarvarpi\othervarpi}
33 \CustomizeMathJax{\let\othervarvarrho\othervarrho}
34 \CustomizeMathJax{\let\varpartialdiff\partial}
```

lwarp_mathjax.txt adds \left/\right support for delimiters.

```
35 \CustomizeMathJax{\let\llbracket\lBrack}
36 \CustomizeMathJax{\let\rrbracket\rBrack}
37 \CustomizeMathJax{\let\dblbrackleft\lBrack}
38 \CustomizeMathJax{\let\dblbrackright\rBrack}
39
40 \CustomizeMathJax{\let\VERT|}
41
42 \CustomizeMathJax{\newcommand{\parallelslant}{\mathrel{\unicode{x02AFD}}}}
43 \CustomizeMathJax{\newcommand{\thething}{\mathord{\unicode{x1F60E}}}}
44 \CustomizeMathJax{\newcommand{\nparallelslant}{%
45     \mathrel{\LWRoverlaysymbols{-}{\unicode{x02AFD}}}%
46 }}
47 \CustomizeMathJax{\newcommand{\xswordsup}{\mathord{\unicode{x2694}}}}
48 \CustomizeMathJax{\newcommand{\xswordsdown}{\mathord{\unicode{x2694}}}}% up
49 \CustomizeMathJax{\newcommand{\notowns}{\mathrel{\unicode{x220C}}}}
50
51 \CustomizeMathJax{\newcommand{\iintop}{\mathop{\unicode{x222C}}\limits}}
52 \CustomizeMathJax{\newcommand{\iiintop}{\mathop{\unicode{x222D}}\limits}}
53 \CustomizeMathJax{\newcommand{\oiint}{\mathop{\unicode{x222F}}\limits}}
54 \CustomizeMathJax{\let\oiintop\oiint}
55 \CustomizeMathJax{\newcommand{\oiiint}{\mathop{\unicode{x2230}}\limits}}
56 \CustomizeMathJax{\let\oiiintop\oiiint}
57 \CustomizeMathJax{\newcommand{\slashint}{\mathop{\unicode{x2A0D}}\limits}}
58 \CustomizeMathJax{\let\slashintop\slashint}
59
60 \CustomizeMathJax{\let\overgroup\overparen}
61 \CustomizeMathJax{\let\wideparen\overparen}
62 \CustomizeMathJax{\let\widearc\overparen}
63 \CustomizeMathJax{\let\wideOarc\overrightarrow}
64 \CustomizeMathJax{\newcommand{\widering}[1]{\stackrel{\unicode{x2218}}{\overgroup{#1}}}}
65
66 \end{warpMathJax}
```

File 196   **`lwarp-framed.sty`**

§ 308   Package   # framed

(*Emulates or patches code by* DONALD ARSENEAU.)

framed (*Pkg*)   framed is supported and patched by lwarp.

for HTML output: Accept all options for lwarp-framed:

```
1 \LWR@ProvidesPackagePass{framed}[2011/10/22]
2
3 \AtBeginDocument{\RequirePackage{xcolor}}% for \convertcolorspec

4 \renewenvironment{framed}
5 {%
6     \LWR@forcenewpage
7     \BlockClass{framed}%
8 }
9 {\endBlockClass}
10
11 \renewenvironment{oframed}
12 {%
```

```
13      \LWR@forcenewpage
14      \BlockClass{framed}%
15 }
16 {\endBlockClass}
17
18
19 \renewenvironment{shaded}
20 {%
21      \convertcolorspec{named}{shadecolor}{HTML}\LWR@tempcolor%
22      \LWR@forcenewpage
23      \BlockClass[background: \LWR@origpound\LWR@tempcolor]{shaded}%
24 }
25 {\endBlockClass}
26
27 \renewenvironment{shaded*}
28 {%
29      \convertcolorspec{named}{shadecolor}{HTML}\LWR@tempcolor%
30      \LWR@forcenewpage
31      \BlockClass[background: \LWR@origpound\LWR@tempcolor]{shaded}%
32 }
33 {\endBlockClass}
34
35
36 \renewenvironment{leftbar}{%
37      \LWR@forcenewpage
38      \BlockClass{framedleftbar}
39      \def\FrameCommand{}%
40      \MakeFramed {}
41 }%
42 {\endMakeFramed\endBlockClass}
43
44
45 \renewenvironment{snugshade}
46 {%
47      \convertcolorspec{named}{shadecolor}{HTML}\LWR@tempcolor%
48      \LWR@forcenewpage
49      \BlockClass[background: \LWR@origpound\LWR@tempcolor]{snugframed}%
50 }
51 {\endBlockClass}
52
53 \renewenvironment{snugshade*}
54 {%
55      \convertcolorspec{named}{shadecolor}{HTML}\LWR@tempcolor%
56      \LWR@forcenewpage
57      \BlockClass[background: \LWR@origpound\LWR@tempcolor]{snugframed}%
58 }
59 {\endBlockClass}
60
61 \let\oframed\framed
62 \let\endoframed\endframed
63
64
65 \RenewEnviron{titled-frame}[1]{%
66      \CustomFBox{#1}{}{0pt}{0pt}{0pt}{0pt}{\BODY}
67 }
```

\CustomFBox   {⟨*toptitle*⟩} {⟨*bottitle*⟩} {⟨*thicknesstop*⟩} {⟨*bottom*⟩} {⟨*left*⟩} {⟨*right*⟩} {⟨*text contents*⟩}

```
68 \renewcommand{\CustomFBox}[7]{%
69     \convertcolorspec{named}{TFFrameColor}{HTML}\LWR@tempcolor%
70     \LWR@forcenewpage
71   \begin{BlockClass}[border: 3px solid \LWR@origpound\LWR@tempcolor]{framed}%
72     \ifthenelse{\isempty{#1}}{}{% not empty
73       \begin{BlockClass}[background: \LWR@origpound\LWR@tempcolor]{framedtitle}%
74         \textcolor{TFTitleColor}{\textbf{#1}}%
75         \end{BlockClass}%
76     }% not empty
77
78     #7
79
80     \ifthenelse{\isempty{#2}}{}{% not empty
81         \convertcolorspec{named}{TFFrameColor}{HTML}\LWR@tempcolor%
82      \begin{BlockClass}[background: \LWR@origpound\LWR@tempcolor]{framedtitle}%
83         \textcolor{TFTitleColor}{\textbf{#2}}%
84         \end{BlockClass}%
85     }% not empty
86     \end{BlockClass}%
87 }
```

\TitleBarFrame   [⟨*marker*⟩] {⟨*title*⟩} {⟨*contents*⟩}

```
88 \renewcommand\TitleBarFrame[3][]{%
89     \CustomFBox%
90         {#2}{}%
91         \fboxrule\fboxrule\fboxrule\fboxrule%
92         {#3}%
93 }
```

```
94 \renewcommand{\TF@Title}[1]{#1}
```

Env   MakeFramed   {⟨*settings*⟩}

```
95 \let\MakeFramed\relax
96 \let\endMakeFramed\relax
97
98 \NewEnviron{MakeFramed}[1]{%
99     \FrameCommand{\begin{minipage}{\linewidth}\BODY\end{minipage}}%
100 }
```

\fb@put@frame   {⟨*frame cmd no split*⟩} {⟨*frame cmd split*⟩}

```
101 \renewcommand*{\fb@put@frame}[2]{%
102     \relax%
103     \@tempboxa%
104 }
```

File 197   **lwarp-froufrou.sty**

§ 309   Package   **froufrou**

(*Emulates or patches code by* NELSON LAGO.)

froufrou (*Pkg*)   froufrou is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{froufrou}[2020/12/22]

```
2 \ExplSyntaxOn
3 \xpretocmd{\setfroufrou}
4     {\edef\LWR@latestfroufrou{\detokenize{#1}}}
5     {}
6     {\LWR@patcherror{froufrou}{setfroufrou}}
7 \ExplSyntaxOff
8
9 \VerifyCommand[lwarp][froufrou]{\froufrou}{E60D7F93008BB892149BBBC2E09983D6}
10
11 \RenewDocumentCommand{\froufrou}{s O{}}{%
12   \nopagebreak[4]\par
13
14   \IfBooleanTF{#1}{\@afterindenttrue}{\@afterindentfalse}
15
16   \nopagebreak[4]\@froufrouspacebefore\nopagebreak[4]
17
18   \bgroup
19     \setfroufrou{#2}%
20     \normalsize
21     \ifdefvoid{\setstretch}{}{\setstretch{\setspace@singlespace}}% normally 1
22     \setlength{\parskip}{0pt}
23     \noindent\centering\bgroup%
24         \begin{center}%                                         lwarp
25         \begin{lateximage}*[froufrou][\LWR@latestfroufrou]%     lwarp
26         \@froufrouOrnament%
27         \end{lateximage}%                                       lwarp
28         \end{center}%                                           lwarp
29     \egroup\par
30   \egroup
31
32   \nopagebreak[4]\@froufrouspaceafter\nopagebreak[4]
33
34   \@froufrouFixSpacingAfter
35
36   \nopagebreak[3]
37
38   \@afterheading
39 }
```

File 198   `lwarp-ftcap.sty`

§ 310   Package   # ftcap

ftcap (*Pkg*)   ftcap is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{ftcap}

File 199   `lwarp-ftnright.sty`

§ 311   Package   # ftnright

ftnright (*Pkg*)   ftnright is ignored.

for HTML output:

Discard all options for lwarp-ftnright:

```
1 \LWR@ProvidesPackageDrop{ftnright}[2014/10/28]
```

File 200   `lwarp-fullminipage.sty`

§ 312   Package   # fullminipage

fullminipage (*Pkg*)   fullminipage is ignored.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{fullminipage}[2014/07/06]
```

```
2 \newenvironment{fullminipage}[1][]{}{}
```

File 201   `lwarp-fullpage.sty`

§ 313   Package   # fullpage

fullpage (*Pkg*)   fullpage is ignored.

for HTML output: Discard all options for lwarp-fullpage:

```
1 \LWR@ProvidesPackageDrop{fullpage}[1994/06/01]
```

File 202   `lwarp-fullwidth.sty`

§ 314   Package   # fullwidth

(*Emulates or patches code by* Marco Daniel.)

fullwidth (*Pkg*)   fullwidth is emulated.

A minipage is used, of no HTML width.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{fullwidth}[2011/11/18]
```

```
2 \newenvironment*{fullwidth}[1][]{%
3 \minipagefullwidth%
4 \minipage{\linewidth}%
5 }
6 {%
7 \endminipage%
8 }
```

File 203   `lwarp-fvextra.sty`

§ 315   Package   # fvextra

(*Emulates or patches code by* Geoffrey M. Poore.)

fvextra (*Pkg*) fvextra is patched for use by lwarp.

for HTML output: 1 `\LWR@ProvidesPackagePass{fvextra}[2024/11/17]`

If line numbers on the right side are used along with `breaklines`, the line numbers will not be aligned.

```
 2 \define@booleankey{FV}{obeytabs}%
 3 %   {\let\FV@ObeyTabsInit\FV@@ObeyTabsInit}%
 4   {\let\FV@ObeyTabsInit\relax}%      lwarp
 5   {\let\FV@ObeyTabsInit\relax}
```

`tabcolor` causes extra HTML tags, destroying the verbatim text alignment, so `tabcolor` is ignored.

```
 6 \define@key{FV}{tabcolor}{}%
```

```
 7 \define@booleankey{FV}{showtabs}%
 8 {\def\FV@TabChar{\FV@TabColor{\FancyVerbTab}}}%
 9 {\let\FV@TabChar\relax}
10
11 \newbool{LWR@FV@breaklines}
12
13 \define@booleankey{FV}{breaklines}%
14   {\boolfalse{FV@breaklines}%
15     \booltrue{LWR@FV@breaklines}%                            lwarp
16 %  \let\FV@ListProcessLine\FV@ListProcessLine@Break}%
17     \let\FV@ListProcessLine\FV@ListProcessLine@NoBreak}%   lwarp
18   {\boolfalse{FV@breaklines}%
19     \boolfalse{LWR@FV@breaklines}%                          lwarp
20     \let\FV@ListProcessLine\FV@ListProcessLine@NoBreak}
21 % \fvset{breaklines}
22
23 \define@key{FV}{breakanywheresymbolpre}{\def\FancyVerbBreakAnywhereSymbolPre{}}
24 \fvset{breakanywheresymbolpre={}}
25
26 \define@key{FV}{breakanywheresymbolpost}{\def\FancyVerbBreakAnywhereSymbolPost{}}
27 \fvset{breakanywheresymbolpost={}}
28
29 \define@key{FV}{breakbeforesymbolpre}{\def\FancyVerbBreakBeforeSymbolPre{}}
30 \fvset{breakbeforesymbolpre={}}
31
32 \define@key{FV}{breakbeforesymbolpost}{\def\FancyVerbBreakBeforeSymbolPost{}}
33 \fvset{breakbeforesymbolpost={}}
34
35 \define@key{FV}{breakaftersymbolpre}{\def\FancyVerbBreakAfterSymbolPre{}}
36 \fvset{breakaftersymbolpre={}}
37
38 \define@key{FV}{breakaftersymbolpost}{\def\FancyVerbBreakAfterSymbolPost{}}
39 \fvset{breakaftersymbolpost={}}
40
41 \define@key{FV}{breaksymbolleft}{\def\FancyVerbBreakSymbolLeft{}}
42
43 \define@key{FV}{breaksymbol}{\fvset{breaksymbolleft={}}}
44
45 \fvset{breaksymbolleft={}}
46
47 \define@key{FV}{breaksymbolright}{\def\FancyVerbBreakSymbolRight{}}
48 \fvset{breaksymbolright={}}
```

Modified to insert a fixed-width space (\nobreakspace) to indent the left margin on indented code, but also allow a line break if needed (\allowbreaK), to allow for break lines.

```
49 \def\FV@DefFVSpace{%
50   \ifbool{FV@showspaces}%
51     {%
52       \def\FV@Space{%
53         \FV@SpaceColor{\FancyVerbSpace}%
54         \allowbreak%
55       }%
56     }%
57     {\def\FV@Space{\nobreakspace\allowbreak}}%
58 }
```

\FancyVerbSpace   Force the use of a vibible space instead of an empty box. From fancyvrb.

```
59 \ifxetexorluatex
60 \def\FancyVerbSpace{\textvisiblespace}
61 \else
62 \@ifundefined{verbvisiblespace}%
63   {%
64     \@ifundefined{textvisiblespace}%
65     {\begingroup\catcode`\ =12 \gdef\FancyVerbSpace{\tt }\endgroup}
66     {\def\FancyVerbSpace{\textvisiblespace}}
67   }%
68   {\def\FancyVerbSpace{\verbvisiblespace}}
69 \fi
```

\LWR@currentFVbackstyle   Contains the style text for the background of the current environment. This is defined in lwarp-fancyvrb.

\LWR@find@currentFVbackstyle   Figures out the style text for the background of the current environment. This is initially defined in lwarp-fancyvrb, and enhanced here.

```
70 \renewcommand*{\LWR@find@currentFVbackstyle}{%
71   \ifundef{\FancyVerbBackgroundColor}%
72   {%
73     \renewcommand*{\LWR@currentFVbackstyle}{}%
74   }%
75   {%
76     \protect\colorlet{LWR@current@color}{\FancyVerbBackgroundColor}%
77     \protect\convertcolorspec%
78       {named}{LWR@current@color}{HTML}%
79       \LWR@currentFVbackcolor\relax%
80     \renewcommand*{\LWR@currentFVbackstyle}{%
81       background:\LWR@origpound\LWR@currentFVbackcolor ; %
82     }%
83   }%
84 }%
```

\FV@BGColor@List

```
85 \def\FV@BGColor@List#1{%
86   \ifx\FancyVerbBackgroundColor\relax
87     \expandafter\@firstoftwo
88   \else
89     \expandafter\@secondoftwo
```

```
90   \fi
91   {#1}%
92   {%
93 %   \setlength{\FV@TmpLength}{\fboxsep}%
94 %    \setlength{\fboxsep}{0pt}%
95 %    \colorbox{\FancyVerbBackgroundColor}{%
96 %      \setlength{\fboxsep}{\FV@TmpLength}%
97         #1%         lwarp
98 %    \rlap{\FancyVerbBackgroundColorVPhantom\strut#1}%
99 %    \hspace{\linewidth}%
100 %    \ifx\FV@RightListFrame\relax\else
101 %      \hspace{-\FV@FrameSep}%
102 %      \hspace{-\FV@FrameRule}%
103 %    \fi
104 %    \ifx\FV@LeftListFrame\relax\else
105 %      \hspace{-\FV@FrameSep}%
106 %      \hspace{-\FV@FrameRule}%
107 %    \fi%
108 %    }%
109 %    \hss%
110 }}
```

{⟨*text*⟩}

\FV@ListProcessLine@NoBreak   Modified to always allow line wrapping because added HTML tags may make run
off the end of the line in the PDF output file before conversion to HTML.

```
111 \VerifyCommand[lwarp][fvextra]{\FV@ListProcessLine@NoBreak}
112    {CF94595D40774FB18D2002C3C9276956}
113
114 \def\FV@ListProcessLine@NoBreak#1{%
115 %   \hbox to \hsize{%
116 %    \kern\leftmargin
117 %    \hbox to \linewidth{%
118     \FV@LeftListNumber%
119     \FV@LeftListFrame%
120     \FV@BGColor@List{%
121     \FancyVerbFormatLine{%
122       \FancyVerbHighlightLine{%
123         \FV@ObeyTabs{\FancyVerbFormatText{#1}}}}}%\hss
124     \FV@RightListFrame%
125     \FV@RightListNumber%
126 %    }%
127 %    \hss}%
128 \null\par%                    lwarp
129 }
130 \newcommand*{\LWR@FV@linethensep}{%
131    \ifbool{LWR@FV@breaklines}%
132      {\theFancyVerbLine\kern\FV@NumberSep}%
133      {\hbox to\z@{\hss\theFancyVerbLine\kern\FV@NumberSep}}%
134 }
135
136 \newcommand*{\LWR@FV@septhenline}{%
137    \ifbool{LWR@FV@breaklines}%
138      {\kern\FV@NumberSep\theFancyVerbLine}%
139      {\hbox to\z@{\kern\FV@NumberSep\theFancyVerbLine\hss}}%
140 }
141
142 \VerifyCommand[lwarp][fvextra]{\FV@Numbers@left}
```

```
143      {57A16473A8AA4214529F6BABEC435311}
144
145  \xpatchcmd{\FV@Numbers@left}%
146      {\hbox to\z@{\hss\theFancyVerbLine\kern\FV@NumberSep}}
147      {\LWR@FV@linethensep}
148      {}
149      {\LWR@patcherror{fvextra}{FV@Numbers@left A}}
150
151  \xpatchcmd{\FV@Numbers@left}%
152      {\hbox to\z@{\hss\theFancyVerbLine\kern\FV@NumberSep}}
153      {\LWR@FV@linethensep}
154      {}
155      {\LWR@patcherror{fvextra}{FV@Numbers@left B}}
156
157  \xpatchcmd{\FV@Numbers@left}%
158      {\hbox to\z@{\hss\theFancyVerbLine\kern\FV@NumberSep}}
159      {\LWR@FV@linethensep}
160      {}
161      {\LWR@patcherror{fvextra}{FV@Numbers@left C}}
162
163  \VerifyCommand[lwarp][fvextra]{\FV@Numbers@right}
164      {6D0F98326BCB22695874D94BEC12E32F}
165
166  \xpatchcmd{\FV@Numbers@right}%
167      {\hbox to\z@{\kern\FV@NumberSep\theFancyVerbLine\hss}}
168      {\LWR@FV@septhenline}
169      {}
170      {\LWR@patcherror{fvextra}{FV@Numbers@right A}}
171
172  \xpatchcmd{\FV@Numbers@right}%
173      {\hbox to\z@{\kern\FV@NumberSep\theFancyVerbLine\hss}}
174      {\LWR@FV@septhenline}
175      {}
176      {\LWR@patcherror{fvextra}{FV@Numbers@right B}}
177
178  \xpatchcmd{\FV@Numbers@right}%
179      {\hbox to\z@{\hss\theFancyVerbLine\kern\FV@NumberSep}}
180      {\LWR@FV@linethensep}
181      {}
182      {\LWR@patcherror{fvextra}{FV@Numbers@right C}}
183
184  \VerifyCommand[lwarp][fvextra]{\FV@Numbers@both}
185      {C349DC2B800D5DD085FFB7620A6289EA}
186
187  \xpatchcmd{\FV@Numbers@both}%
188      {\hbox to\z@{\hss\theFancyVerbLine\kern\FV@NumberSep}}
189      {\LWR@FV@linethensep}
190      {}
191      {\LWR@patcherror{fvextra}{FV@Numbers@both A}}
192
193  \xpatchcmd{\FV@Numbers@both}%
194      {\hbox to\z@{\hss\theFancyVerbLine\kern\FV@NumberSep}}
195      {\LWR@FV@linethensep}
196      {}
197      {\LWR@patcherror{fvextra}{FV@Numbers@both B}}
198
199  \xpatchcmd{\FV@Numbers@both}%
200      {\hbox to\z@{\hss\theFancyVerbLine\kern\FV@NumberSep}}
201      {\LWR@FV@linethensep}
202      {}
```

```
203        {\LWR@patcherror{fvextra}{FV@Numbers@both C}}
204
205 \xpatchcmd{\FV@Numbers@both}%
206        {\hbox to\z@{\kern\FV@NumberSep\theFancyVerbLine\hss}}
207        {\LWR@FV@septhenline}
208        {}
209        {\LWR@patcherror{fvextra}{FV@Numbers@both D}}
210
211 \xpatchcmd{\FV@Numbers@both}%
212        {\hbox to\z@{\kern\FV@NumberSep\theFancyVerbLine\hss}}
213        {\LWR@FV@septhenline}
214        {}
215        {\LWR@patcherror{fvextra}{FV@Numbers@both E}}
216
217 \xpatchcmd{\FV@Numbers@both}%
218        {\hbox to\z@{\hss\theFancyVerbLine\kern\FV@NumberSep}}
219        {\LWR@FV@linethensep}
220        {}
221        {\LWR@patcherror{fvextra}{FV@Numbers@both F}}
```

\FVC@SaveVerb@Extra@ii   {⟨*name*⟩} {⟨*raw text*⟩} {⟨*text*⟩}

Modified to add \LWR@HTMLsanitize@use@tmb to the stored macro. This is used
on recall to sanitize for HTML unless in a lateximage.

```
222 \VerifyCommand[lwarp][fvextra]{\FVC@SaveVerb@Extra@ii}
223        {BCE88217BA577F70BAC8158E110E404C}
224
225 \def\FVC@SaveVerb@Extra@ii#1#2#3{%
226    \global\let\FV@AfterSave\FancyVerbAfterSave
227    \endgroup
228 %  \@namedef{FV@SV@#1}{#3}%
229    \@namedef{FV@SV@#1}{\LWR@HTMLsanitize@use@tmpb{#3}}%       lwarp
230 %  \@namedef{FV@SVRaw@#1}{#2}%
231    \@namedef{FV@SVRaw@#1}{\LWR@HTMLsanitize@use@tmpb{#2}}%   lwarp
232    \FV@AfterSave}%
```

\FVC@Verb@Extra@ii   {⟨*text*⟩}

Adds the opening and closing tags.

```
233 \VerifyCommand[lwarp][fvextra]{\FVC@Verb@Extra@ii}
234        {C81AC0F7DFE7FBB55CF8B5B7F24FA56A}
235
236 \def\FVC@Verb@Extra@ii#1{%
237    \def\tmpb{#1}%                                           lwarp
238    \ifbool{LWR@verbtags}%                    lwarp
239     {\LWR@htmltag{span class=\textquotedbl{}fancyvrb\textquotedbl}}% lwarp
240        {}%                                    lwarp
241    \LWR@HTMLsanitize@tmpb%                              lwarp
242  \ifx\FancyVerbBackgroundColor\relax
243    \expandafter\@firstoftwo
244  \else
245    \expandafter\@secondoftwo
246  \fi
247    {\ifbool{FV@breaklines}%
248 %    {\FV@InsertBreaks{\FancyVerbFormatInline}{#1}}%
249 %    {\mbox{\FancyVerbFormatInline{#1}}}}%
250    {\FV@InsertBreaks{\FancyVerbFormatInline}{\tmpb}}%  lwarp
251    {\mbox{\FancyVerbFormatInline{\tmpb}}}}}%     lwarp
```

```
252   {\setlength{\FV@TmpLength}{\fboxsep}%
253   \ifx\FancyVerbBackgroundColorPadding\relax
254     \setlength{\fboxsep}{0pt}%
255   \else
256     \setlength{\fboxsep}{\FancyVerbBackgroundColorPadding}%
257   \fi
258   \colorbox{\FancyVerbBackgroundColor}{%
259     \setlength{\fboxsep}{\FV@TmpLength}%
260 %       \FancyVerbBackgroundColorVPhantom\FancyVerbFormatInline{#1}}}%
261     \FancyVerbBackgroundColorVPhantom\FancyVerbFormatInline{\tmpb}}}% lwarp
262   \ifbool{LWR@verbtags}%                    lwarp
263     {\LWR@htmltag{/span}}%                  lwarp
264     {}%                                     lwarp
265   \endgroup}
```

\FV@UseVerb@Extra   {⟨*text*⟩}

Adds the opening and closing tags.

```
266 \VerifyCommand[lwarp][fvextra]{\FV@UseVerb@Extra}
267     {C81AC0F7DFE7FBB55CF8B5B7F24FA56A}
268
269 \let\FV@UseVerb@Extra\FVC@Verb@Extra@ii%
```

\FVC@EscVerb@ii   {⟨*text*⟩}

Adds the opening and closing tags.

```
270 \VerifyCommand[lwarp][fvextra]{\FVC@EscVerb@ii}
271     {C81AC0F7DFE7FBB55CF8B5B7F24FA56A}
272
273 \let\FVC@EscVerb@ii\FVC@Verb@Extra@ii%
```

\FVB@VerbatimWrite   Disable santizing HTML while writing the file. HTML will be sanitized on \VerbatimInput.

```
274 \VerifyCommand[lwarp][fvextra]{\FVB@VerbatimWrite}
275     {B092E8AB57DB2ABBA815BC39DB5256DC}
276
277 \xpatchcmd{\FVB@VerbatimWrite}
278     {\FV@Scan}
279     {\boolfalse{LWR@HTMLsanitize@tmpb@enable}\FV@Scan}
280     {}
281     {\LWR@patcherror{fvextra}{FVB@VerbatimWrite}}
```

\FVB@VerbatimBuffer   Disable santizing HTML while writing the buffer. HTML will be sanitized on \VerbatimInsertBuffer.

```
282 \VerifyCommand[lwarp][fvextra]{\FVB@VerbatimBuffer}
283     {151F97F8D2944BDA11300CBF70FB40C9}
284
285 \xpatchcmd{\FVB@VerbatimBuffer}
286     {\FV@Scan}
287     {\boolfalse{LWR@HTMLsanitize@tmpb@enable}\FV@Scan}
288     {}
289     {\LWR@patcherror{fvextra}{FVB@VerbatimBuffer}}
```

\VerbatimInsertBuffer@def@FV@Line
```
290 \VerifyCommand[lwarp][fvextra]{\VerbatimInsertBuffer@def@FV@Line}
```

```
291    {5C6EACB0FB4432BEA081C784456845CD}
292
293 \def\VerbatimInsertBuffer@def@FV@Line#1{%
294    \FVExtraRetokenizeVArg{\def\FV@Line}{}{#1}%
295    \LetLtxMacro\tmpb\FV@Line%        lwarp
296    \LWR@HTMLsanitize@tmpb%           lwarp
297    \LetLtxMacro\FV@Line\tmpb%        lwarp
298 }%
```

File 204   **lwarp-fwlw.sty**

§ 316   Package   # fwlw

fwlw (*Pkg*)   fwlw is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{fwlw}

2 \newbox\FirstWordBox   \global\setbox\FirstWordBox\hbox{}
3 \newbox\NextWordBox   \global\setbox\NextWordBox\hbox{}
4 \newbox\LastWordBox   \global\setbox\LastWordBox\hbox{}
5 \def\ps@fwlwhead{}
6 \def\ps@NextWordFoot{}

File 205   **lwarp-gensymb.sty**

§ 317   Package   # gensymb

(*Emulates or patches code by* WALTER SCHMIDT.)

gensymb (*Pkg*)   gensymb works as-is for SVG math, and uses the MATHJAX package.

for HTML output: 1 \LWR@ProvidesPackagePass{gensymb}[2003/07/02]

2 \begin{warpMathJax}
3 \CustomizeMathJax{\require{gensymb}}
4 \end{warpMathJax}

File 206   **lwarp-gentombow.sty**

§ 318   Package   # gentombow

gentombow (*Pkg*)   gentombow is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{gentombow}[2018/05/17]

2 \newcommand{\settombowbanner}[1]{}
3 \newcommand{\settombowbannerfont}[1]{}
4 \newcommand{\settombowwidth}[1]{}
5 \newcommand{\settombowbleed}[1]{}
6 \newcommand{\settombowcolor}[1]{}

File 207   `lwarp-geometry.sty`

§ 319   Package   # geometry

(*Emulates or patches code by* HIDEO UMEKI.)

geometry (*Pkg*)   geometry is preloaded by lwarp, but must be nullified as seen by the user's source code.

for HTML output: Discard all options for lwarp-geometry:

1 \LWR@ProvidesPackageDropA{geometry}{2018/04/16}

If geometry is never loaded by the user, it will be loaded by lwarp \AtBeginDocument. If this is the case, the page layout should not be changed but the user macros should still be nullified.

2 \ifbool{LWR@allowanothergeometry}{%

Assign and set the selected geometry with reset prepended. \AtEndPreamble lwarp will save this, then set its own geometry.

3   \edef\LWR@tempone{reset,\@ptionlist{\@currname.\@currext}}%
4   \expandafter\LWR@origgeometry\expandafter{\LWR@tempone}%
5 }{}% LWR@allowanothergeometry

The user-level commands are nullified:

6 \renewcommand*{\geometry}[1]{}
7 \renewcommand*{\newgeometry}[1]{}
8 \renewcommand*{\restoregeometry}{}
9 \renewcommand*{\savegeometry}[1]{}
10 \renewcommand*{\loadgeometry}[1]{}

File 208   `lwarp-ghsystem.sty`

§ 320   Package   # ghsystem

(*Emulates or patches code by* CLEMENS NIEDERBERGER.)

ghsystem (*Pkg*)   ghsystem is patched for use by lwarp.

⚠ \ghspic images   Images must be provided in SVG format, unless JPG is specified. It is recommended to create a local images directory, copy into it the relevent PDF ghsystem images, and then convert them with

Enter ⇒   **lwarpmk pdftosvg images/*.pdf**

for HTML output: 1 \LWR@ProvidesPackagePass{ghsystem}[2020/02/17]

2 \ExplSyntaxOn
3

```
4 \VerifyCommand[lwarp][ghsystem]{\ghsystem_filler:n}{2B8CCE2EC0EC4AB8FA4C4E4A68FFCE70}
5
6 \cs_set_protected:Npn \ghsystem_filler:n #1
7   { \emph { \textless #1 \textgreater } }
8
9 \VerifyCommand[lwarp][ghsystem]{\ghsystem_pic:n}{950F001D9FCDAFF7A9154739DC8025BB}
10
11 \cs_set_protected:Npn \ghsystem_pic:n #1
12   {
13     \__ghsystem_includegraphics:xn
14       {
15 %        scale = \fp_to_tl:N \l__ghsystem_picture_scale_fp
16        width = 1.25cm
17        \exp_not:V \l__ghsystem_picture_includegraphics_tl
18      }
19     { ghsystem_ #1 . \l__ghsystem_picture_type_tl }
20   }
21
22 \ExplSyntaxOff
```

File 209   **`lwarp-gindex.sty`**

§ 321   Package   # gindex

(*Emulates or patches code by* JAVIER BEZOS.)

gindex (*Pkg*)   **gindex** is patched for use by lwarp.

See section 8.6.16.

for HTML output: 1 \LWR@ProvidesPackagePass{gindex}[2019/10/07]

Set the index page and range separators. These are set \AtBeginDocument to allow the user to change them. They are then protected so that the lwarp core looks for the tokens instead of their expanded contents, since the *.ind files will contain \indexpagessep and \indexrangesep instead of their literal contents. Finally, lwarp is told of the gindex macros.

```
2 \AtBeginDocument{
3     \robustify{\indexpagessep}
4     \robustify{\indexrangesep}
5     \renewcommand*{\IndexPageSeparator}{\indexpagessep}
6     \renewcommand*{\IndexRangeSeparator}{\indexrangesep}
7 }
```

\hyperindexref is added:

```
8 \def\addindexitem#1#2{%
9   \indexflushitem
10 \gix@getspecial#1\indexspecial\indexspecial\@@\indexitem{\hyperindexref{#2}}}
11
12 \def\addindexsubitem#1#2{%
13   \stepcounter{indexsubitems}%
14 \gix@getspecial#1\indexspecial\indexspecial\@@\indexsubitem{\hyperindexref{#2}}}
15
16 \def\addindexsubsubitem#1#2{%
17 \gix@getspecial#1\indexspecial\indexspecial\@@\indexsubsubitem{\hyperindexref{#2}}}
```

Uses a <div> of class indexheading:

```
18 \renewcommand\indexheading[1]{%
19     \begin{BlockClass}{indexheading}
20     \MakeUppercase{#1}%
21     \end{BlockClass}
22 }
```

File 210   **`lwarp-gloss.sty`**

§ 322   Package   # gloss

(*Emulates or patches code by* JOSE LUIS DÍIAZ, JAVIER BEZOS.)

gloss (*Pkg*)   gloss is patched for use by lwarp.

To process the HTML glossary:

> bibtex <projectname>_html.gls

for HTML output: 1 \LWR@ProvidesPackagePass{gloss}[2002/07/26]

\BaseJobname is added to the label in case xr or xr-hyper are used.

```
2 \VerifyCommand[lwarp][gloss]{\gls@gloss@iii}{96590CC8FAE12295596B9F664BE4AF8C}
3
4 \xpatchcmd{\gls@gloss@iii}
5     {\thepage}
6     {\theLWR@previousautopagelabel}
7     {}
8     {\LWR@patcherror{gloss}{gls@gloss@iii}}
9
10 \VerifyCommand[lwarp][gloss]{\gls@page@i}{C05FCEACF0A1F96FC09A218684543574}
11
12 \def\gls@page@i#1#2{%
13   \endgroup%
14   \global\@namedef{glsp@#1}{\nameref{\BaseJobname-autopage-#2}}}%
```

File 211   **`lwarp-glossaries.sty`**

§ 323   Package   # glossaries

(*Emulates or patches code by* NICOLA L.C. TALBOT.)

glossaries (*Pkg*)   *lwarpmk* has the commands **lwarpmk printglossary** and **lwarpmk htmlglossary**,
processing glossaries   which process the glossaries created by the glossaries package using that package's
GlossaryCmd (*Opt*)   *makeglossaries* program.

Default: makeglossaries

printglossary (*Opt*) [lwarpmk]   The shell command to execute is set by the lwarp option GlossaryCmd, which
htmlglossary (*Opt*) [lwarpmk]   defaults to **makeglossaries**. The print or HTML glossary filename is appended to
this command.

⚠ *makeglossaries* not   In some situations it may be required to modify the default command, such as to
found   add the **perl** command in front:

```
\usepackage[
    GlossaryCmd={perl makeglossaries},
] {lwarp}
```

xindy language To set the language to use for processing glossaries with *xindy*:

```
\usepackage[
    GlossaryCmd={makeglossaries -L english},
] {lwarp}
```

Other options for *makeglossaries* may be set as well.

placement and TOC options The glossaries may be placed in a numbered or unnumbered section, given a TOC entry, and placed inline or on their own HTML page:

Numbered section, on its own HTML page:

```
\usepackage[xindy,toc,numberedsection=nolabel]{glossaries}
...
\printglossaries
```

Unnumbered section, inline with the current HTML page:

```
\usepackage[xindy,toc]{glossaries}
...
\printglossaries
```

Unnumbered section, on its own HTML page:

```
\usepackage[xindy,toc]{glossaries}
...
\ForceHTMLPage
\printglossaries
```

⚠ glossary style The default style=item option for glossaries conflicts with lwarp, so the style is forced to index instead.

⚠ number list The page number list in the printed form would become \namerefs in HTML, which could become a very long string if many items are referenced. For now, the number list is simply turned off.

print/HTML versions The print and HTML versions of the glossary differ in their internal page numbers. Separate commands for generating print and HTML glossaries are used, even though the page number is currently ignored.

for HTML output:
```
1 \PassOptionsToPackage{xindy}{glossaries}
2
3 \LWR@ProvidesPackagePass{glossaries}[2018/07/23]
4
5 \setupglossaries{nonumberlist}
6 \setglossarystyle{index}
```

Patched to fix TOC pointing to the previous page:

```
7 \VerifyCommand[lwarp][glossaries]{\@p@glossarysection}{129DC9CFB9484FC34C7B81E32BBB0452}
8
9 \renewcommand*{\@p@glossarysection}[2]{%
10   \glsclearpage
11   \LWR@phantomsection
12   \ifdefempty\@@glossarysecstar
13   {%
```

```
14        \csname\@@glossarysec\endcsname{#2}%
15    }%
16    {%
```

In the original, the TOC entry was made before the section, thus linking to the phantomsection in the printed version, but for HTML, this caused the link to point to the page before the glossaries, which could be a different HTML file. Here, the TOC entry is made after the section is created:

```
17        \csname\@@glossarysec\endcsname*{#2}%
18    \@gls@toc{#1}{\@@glossarysec}% Moved after the previous line.
19    }%
20    \@@glossaryseclabel
21 }
```

lwarp's sectioning commands cannot handle robust macros when splitting HTML into named filenames. glossaries uses \translate in sectioning names, and \translate is robust and cannot be expanded. The following pre-expands the translations at this moment, making use of \translatelet.

```
22 \newcommand*{\LWR@comp@glossaryname}{\translate{Glossary}}
23
24 \ifdefstrequal{\glossaryname}{\LWR@comp@glossaryname}{
25     \translatelet\LWR@translatetemp{Glossary}
26     \edef\glossaryname{\LWR@translatetemp}
27 }{}
28
29 \newcommand*{\LWR@comp@acronymname}{\translate{Acronym}}
30
31 \ifdefstrequal{\acronymname}{\LWR@comp@acronymname}{
32     \translatelet\LWR@translatetemp{Acronym}
33     \edef\acronymname{\LWR@translatetemp}
34 }{}
35
36 \newcommand*{\LWR@comp@glssymbolsgroupname}{\translate{Symbols (glossaries)}}
37
38 \ifdefstrequal{\glssymbolsgroupname}{\LWR@comp@glssymbolsgroupname}{
39     \translatelet\LWR@translatetemp{Symbols (glossaries)}
40     \edef\glssymbolsgroupname{\LWR@translatetemp}
41 }{}
42
43 \newcommand*{\LWR@comp@glsnumbersgroupname}{\translate{Numbers (glossaries)}}
44
45 \ifdefstrequal{\glsnumbersgroupname}{\LWR@comp@glsnumbersgroupname}{
46     \translatelet\LWR@translatetemp{Numbers (glossaries)}
47     \edef\glsnumbersgroupname{\LWR@translatetemp}
48 }{}
```

File 212   **`lwarp-gmeometric.sty`**

§ 324   Package   **gmeometric**

gmeometric (*Pkg*)   gmeometric is ignored.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{gmeometric}[2008/11/22]
2 \RequirePackageWithOptions{geometry}
```

File 213 **`lwarp-graphics.sty`**

§ 325 Package **graphics**

(*Emulates or patches code by* D. P. CARLISLE.)

graphics (*Pkg*) **graphics** is emulated.

for HTML output: 1 \LWR@ProvidesPackagePass{graphics}[2020/08/30]

§ 325.1 Graphics extensions

\DeclareGraphicsExtensions {⟨*list*⟩}

\AtBeginDocument allow SVG files instead of PDF:

```
2 \AtBeginDocument{
3 \DeclareGraphicsExtensions{.svg,.SVG,.gif,.GIF,.png,.PNG,.jpg,.JPG,.jpeg,.JPEG}
4 \DeclareGraphicsRule{.svg}{svg}{.svg}{}
5 \DeclareGraphicsRule{.SVG}{svg}{.SVG}{}
6 }
```

Inside a lateximage, allow PDF instead of SVG:

```
7 \ifpdf
8 \appto\LWR@restoreorigformatting{%
9 \DeclareGraphicsExtensions{.pdf,.PDF,.gif,.GIF,.png,.PNG,.jpg,.JPG,.jpeg,.JPEG}%
10 }
11 \else% \ifpdf
12      \ifXeTeX
13 \appto\LWR@restoreorigformatting{%
14 \DeclareGraphicsExtensions{.pdf,.PDF,.gif,.GIF,.png,.PNG,.jpg,.JPG,.jpeg,.JPEG}%
15 }
16      \else
17 \appto\LWR@restoreorigformatting{%
18 \DeclareGraphicsExtensions{.eps,.EPS,.gif,.GIF,.png,.PNG,.jpg,.JPG,.jpeg,.JPEG}%
19 }
20      \fi
21 \fi
```

§ 325.2 Length conversions and graphics options

⚠ whitespace A scaled image in LaTeX by default takes only as much space on the page as it requires, but HTML browsers use as much space as the original unscaled image would have taken, with the scaled image over- or under-flowing the area.

Used to store the user's selected dimensions and HTML class.

The class defaults to "inlineimage" unless changed by a class=xyx option.

```
22 \newlength{\LWR@igwidth}
23 \newlength{\LWR@igheight}
24 \newcommand*{\LWR@igwidthstyle}{}
25 \newcommand*{\LWR@igheightstyle}{}
26 \newcommand*{\LWR@igorigin}{}
```

```
27 \newcommand*{\LWR@igangle}{}
28 \newcommand*{\LWR@igxscale}{1}
29 \newcommand*{\LWR@igyscale}{1}
30
31 \newbool{LWR@igkeepaspectratio}
32 \boolfalse{LWR@igkeepaspectratio}
33
34 \newcommand*{\LWR@igclass}{inlineimage}


35 \newcommand*{\LWR@igalt}{\ImageAltText}
```

Set the actions of each of the key/value combinations for `\includegraphics`. Many are ignored.

If an optional width was given, set an HTML style:

```
36 \define@key{igraph}{width}{%
37 \setlength{\LWR@igwidth}{#1}%
38 \ifthenelse{\lengthtest{\LWR@igwidth > 0pt}}%
39 {%
```

Default to use the converted fixed length given:

```
40    \renewcommand*{\LWR@igwidthstyle}{width:\LWR@printlength{\LWR@igwidth}}%
```

If ex or em dimensions were given, use those instead:

```
41    \IfEndWith{#1}{ex}%
42    {\renewcommand*{\LWR@igwidthstyle}{width:#1}}% yes ex
43    {}% not ex
44    \IfEndWith{#1}{em}%
45    {\renewcommand*{\LWR@igwidthstyle}{width:#1}}% yes em
46    {}% not em
47    \IfEndWith{#1}{\%}%
48    {\renewcommand*{\LWR@igwidthstyle}{width:#1}}% yes percent
49    {}% not percent
50    \IfEndWith{#1}{px}%
51    {\renewcommand*{\LWR@igwidthstyle}{width:#1}}% yes px
52    {}% not px
53 }{}% end of length > 0pt
54 }
```

If an optional height was given, set an HTML style:

```
55 \define@key{igraph}{height}{%
56 \setlength{\LWR@igheight}{#1}%
57 \ifthenelse{\lengthtest{\LWR@igheight > 0pt}}%
58 {%
```

Default to use the converted fixed length given:

```
59    \renewcommand*{\LWR@igheightstyle}{%
60    height:\LWR@printlength{\LWR@igheight} % extra space
61    }%
```

If ex or em dimensions were given, use those instead:

```
62    \IfEndWith{#1}{ex}%
```

```
63    {\renewcommand*{\LWR@igheightstyle}{height:#1}}% yes ex
64    {}% not ex
65    \IfEndWith{#1}{em}%
66    {\renewcommand*{\LWR@igheightstyle}{height:#1}}% yes em
67    {}% not em
68    \IfEndWith{#1}{\%}%
69    {\renewcommand*{\LWR@igheightstyle}{height:#1}}% yes percent
70    {}% not percent
71    \IfEndWith{#1}{px}%
72    {\renewcommand*{\LWR@igheightstyle}{height:#1}}% yes px
73    {}% not px
74 }{}% end of length > 0pt
75 }
```

Handle keepaspectratio key:

```
76 \define@key{igraph}{keepaspectratio}[false]{%
77    \booltrue{LWR@igkeepaspectratio}%
78 }
```

Handle origin key:

```
79 \define@key{igraph}{origin}[c]{%
80    \renewcommand*{\LWR@igorigin}{#1}%
81 }
```

Handle angle key:

```
82 \define@key{igraph}{angle}{\renewcommand*{\LWR@igangle}{#1}}
```

Handle class key:

```
83 \define@key{igraph}{class}{\renewcommand*{\LWR@igclass}{#1}}
```

Handle alt key:

```
84 \define@key{igraph}{alt}{\renewcommand*{\LWR@igalt}{#1}}
```

It appears that **graphicx** does not have separate keys for xscale and yscale. scale adjusts both at the same time.

```
85 \define@key{igraph}{scale}{%
86    \ifthenelse{\equal{#1}{1}}{}{%  must expand #1
87        \PackageNote{lwarp}{%
88        It is recommended to use ``[width=xx\protect\linewidth]''\MessageBreak
89            instead of ``[scale=yy]'',%
90        }%
91    }%
92    \renewcommand*{\LWR@igxscale}{#1}%
93    \renewcommand*{\LWR@igyscale}{#1}%
94 }
```

Numerous ignored keys:

```
95 \define@key{igraph}{bb}{}
96 \define@key{igraph}{bbllx}{}
97 \define@key{igraph}{bblly}{}
98 \define@key{igraph}{bburx}{}
```

```
 99 \define@key{igraph}{bbury}{}
100 \define@key{igraph}{natwidth}{}
101 \define@key{igraph}{natheight}{}
102 \define@key{igraph}{hiresbb}[true]{}
103 \define@key{igraph}{viewport}{}
104 \define@key{igraph}{trim}{}
105 \define@key{igraph}{totalheight}{}
106 \define@key{igraph}{clip}[true]{}
107 \define@key{igraph}{draft}[true]{}
108 \define@key{igraph}{type}{}
109 \define@key{igraph}{ext}{}
110 \define@key{igraph}{read}{}
111 \define@key{igraph}{command}{}
```

New in v1.1a:

```
112 \define@key{igraph}{quite}{}
113 \define@key{igraph}{page}{}
114 \define@key{igraph}{pagebox}{}
115 \define@key{igraph}{interpolate}[true]{}
```

New in v1.1b:

```
116 \define@key{igraph}{decodearray}{}
```

§ 325.3 Printing HTML styles

\LWR@rotstyle   {⟨*prefix*⟩} {⟨*degrees*⟩}

Prints the rotate style with the given prefix.

prefix is -ms- or -webkit- or nothing, and is used to generate three versions of the transform:rotate style.

```
117 \newcommand*{\LWR@rotstyle}[2]{%
118     \edef\LWR@tempone{#2}%
119     \setcounter{LWR@tempcountone}{-1*\real{\LWR@tempone}} % space
120     #1transform:rotate(\arabic{LWR@tempcountone}deg); % space
121 }
```

\LWR@scalestyle   {⟨*prefix*⟩} {⟨*xscale*⟩} {⟨*yscale*⟩}

Prints the scale style with the given prefix.

prefix is -ms- or -webkit- or nothing, and is used to generate three versions of the transform:scale style.

```
122 \newcommand*{\LWR@scalestyle}[3]{%
123     #1transform:scale(#2,#3);
124 }
```

§ 325.4 **\includegraphics**

\LWR@opacity   For HTML, used only for \includegraphics.

\LWR@opacity may be set by the transparent package.

```
125 \def\LWR@opacity{1}
```

\LWR@imagesizebox   Used to determine the actual image size if needed.

```
126 \newsavebox{\LWR@imagesizebox}
```

\LWR@HTML@Gin@setfile   {⟨*w*⟩} {⟨*h*⟩} {⟨*filename*⟩} Sets the parsed filename for HTML output.

```
127 \newcommand*{\LWR@HTML@Gin@setfile}[3]{%
128     \xdef\LWR@parsedfilename{#3}%
129 }
```

class (*Key*) [Gin]   CSS class for the image.

Define the new class key for the print-mode version of \includegraphics, which is enabled inside a lateximage.

```
130 \AtBeginDocument{
131 \define@key{Gin}{class}{}
132 }
```

\LWR@replaceEPSSVG

Usually, references to EPS files become SVG files, but if the epstopdf package is being used, it automatically converts EPS to PDF, and the following must NOT be done.

```
133 \AtBeginDocument{
134 \IfPackageLoadedTF{epstopdf}
135 {
136     \newcommand*{\LWR@replaceEPSSVG}{}
137 }{%
138     \newcommand*{\LWR@replaceEPSSVG}{%
139         \StrSubstitute{\LWR@tempone}{.eps}{.svg}[\LWR@tempone]%
140         \StrSubstitute{\LWR@tempone}{.EPS}{.SVG}[\LWR@tempone]%
141     }
142 }%
143 }
```

         * [⟨*2: options*⟩] [⟨*3: options*⟩] {⟨*4: filename*⟩}

\LWR@ig@useactualimagesize   If formatting for a word processor, find and set the actual image size, without rotation, using PDF instead of SVG to find the original bounding box:

```
144 \newcommand*{\LWR@ig@useactualimagesize}[4]{%
145     \begingroup%
146     \LWR@restoreorigformatting%
147     \ifpdf%
148     \appto\LWR@restoreorigformatting{%
149         \DeclareGraphicsExtensions{%
150             .pdf,.PDF,.gif,.GIF,.png,.PNG,.jpg,.JPG,.jpeg,.JPEG%
151         }%
152     }%
153     \else% \ifpdf
154             \ifXeTeX%
```

```
155    \appto\LWR@restoreorigformatting{%
156        \DeclareGraphicsExtensions{%
157            .pdf,.PDF,.gif,.GIF,.png,.PNG,.jpg,.JPG,.jpeg,.JPEG%
158        }%
159    }%
160        \else%
161    \appto\LWR@restoreorigformatting{%
162        \DeclareGraphicsExtensions{%
163            .eps,.EPS,.gif,.GIF,.png,.PNG,.jpg,.JPG,.jpeg,.JPEG%
164        }%
165    }%
166        \fi%
167    \fi% \ifpdf
```

For a word processor, do not use rotation:

```
168    \ifbool{FormatWP}{\define@key{Gin}{angle}{}}{}%

169    \IfBooleanTF{#1}%
170    {% starred
171        \IfValueTF{#3}%
172        {%
173            \global\sbox{\LWR@imagesizebox}{%
174                \LWR@origincludegraphics*[#2][#3]{#4}%
175            }%
176        }%
177        {%
178            \IfValueTF{#2}%
179            {%
180                \global\sbox{\LWR@imagesizebox}{%
181                    \LWR@origincludegraphics*[#2]{#4}%
182                }%
183            }{%
184                \global\sbox{\LWR@imagesizebox}{%
185                    \LWR@origincludegraphics*{#4}%
186                }%
187            }%
188        }%
189    }% starred
190    {% not starred
191        \IfValueTF{#3}%
192        {%
193            \global\sbox{\LWR@imagesizebox}{%
194                \LWR@origincludegraphics[#2][#3]{#4}%
195            }%
196        }%
197        {%
198            \IfValueTF{#2}%
199            {%
200                \global\sbox{\LWR@imagesizebox}{%
201                    \LWR@origincludegraphics[#2]{#4}%
202                }%
203            }{%
204                \global\sbox{\LWR@imagesizebox}{%
205                    \LWR@origincludegraphics{#4}%
206                }%
207            }%
208        }%
209    }% not starred
210    \endgroup%
211    \settowidth{\LWR@igwidth}{\usebox{\LWR@imagesizebox}}%
```

```
212    \global\renewcommand*{\LWR@igwidthstyle}{%
213        width:\LWR@printlength{\LWR@igwidth}}%
214    }%
215    \settoheight{\LWR@igheight}{\usebox{\LWR@imagesizebox}}%
216    \global\renewcommand*{\LWR@igheightstyle}{%
217        height:\LWR@printlength{\LWR@igheight}}%
218    }%
219 }
```

\LWR@ig@htmltag   For the HTML reference, add the graphicspath, filename, extension, alt tag, style,
                       and class.

```
220 \newcommand*{\LWR@ig@htmltag}{%
221    img\LWR@indentHTML%
222    src=\textquotedbl%

223    \detokenize\expandafter{\LWR@parsedfilename}%

224    \textquotedbl\LWR@indentHTML%
```

Only include a style tag if a width, height, angle, or scale was given:

```
225    \ifthenelse{
226        \NOT\equal{\LWR@igwidthstyle}{} \OR
227        \NOT\equal{\LWR@igheightstyle}{} \OR
228        \NOT\equal{\LWR@igorigin}{} \OR
229        \NOT\equal{\LWR@igangle}{} \OR
230        \NOT\equal{\LWR@igxscale}{1} \OR
231        \NOT\equal{\LWR@igyscale}{1}
232    }%
233    {%
234        style=\textquotedbl\LWR@indentHTML
235        \ifthenelse{\NOT\equal{\LWR@igwidthstyle}{}}%
236            {\LWR@igwidthstyle;\LWR@indentHTML}{}%
237        \ifthenelse{\NOT\equal{\LWR@igheightstyle}{}}%
238            {\LWR@igheightstyle;\LWR@indentHTML}{}%
239        \ifthenelse{\NOT\equal{\LWR@igorigin}{}}%
240            {%
241                transform-origin: \LWR@originnames{\LWR@igorigin};%
242                \LWR@indentHTML%
243            }{}%
244        \ifthenelse{\NOT\equal{\LWR@igangle}{}}%
245        {%
246            \LWR@rotstyle{-ms-}{\LWR@igangle}\LWR@indentHTML
247            \LWR@rotstyle{-webkit-}{\LWR@igangle}\LWR@indentHTML
248            \LWR@rotstyle{}{\LWR@igangle }\LWR@indentHTML
249        }{}%
250        \ifthenelse{%
251            \NOT\equal{\LWR@igxscale}{1}\OR%
252            \NOT\equal{\LWR@igyscale}{1}%
253        }%
254        {%
255            \LWR@scalestyle{-ms-}{\LWR@igxscale}{\LWR@igyscale}%
256            \LWR@indentHTML
257            \LWR@scalestyle{-webkit-}{\LWR@igxscale}{\LWR@igyscale}%
258            \LWR@indentHTML
259            \LWR@scalestyle{}{\LWR@igxscale}{\LWR@igyscale}%
260            \LWR@indentHTML
261        }{}%
262        %
263        \ifthenelse{\NOT\equal{\LWR@opacity}{1}}%
```

```
264            {opacity:\LWR@opacity;\LWR@indentHTML}{}%
265        %
266            \textquotedbl\LWR@indentHTML%
267    }{}%
```

Set the class and alt tag:

```
268    class=\textquotedbl\LWR@igclass\textquotedbl\LWR@indentHTML%
269    alt=\textquotedbl\AltTextOpen\LWR@igalt\AltTextClose\textquotedbl\ \LWR@orignewline%
270 }% end of image tags
```

\LWR@includegraphicsb   * [⟨*2: options*⟩] [⟨*3: options*⟩] {⟨*4: filename*⟩}

graphics syntax is \includegraphics * [⟨*llx,lly*⟩] [⟨*urx,ury*⟩] {⟨*filename*⟩}

graphicx syntax is \includegraphics [⟨*key values*⟩] {⟨*filename*⟩}

If #3 is empty, only one optional argument was given, thus graphicx syntax.

If using \epsfig or \psfig from the epsfig package, #4 will be \LWR@epsfig@filename, which will have been set by the file or figure keys. Therefore, #4 must not be used until after the keys have been processed.

```
271 \NewDocumentCommand{\LWR@includegraphicsb}{s o o m}
272 {%
```

Start the image tag on a new line, allow PDF output word wrap:

```
273    \LWR@orignobreakspace \LWR@orignewline%
```

Temporarily compute \linewidth, \textwidth, \textheight arguments with a 6x9 inch size until the next \endgroup.

```
274    \begin{LWR@setvirtualpage}%
```

For correct em sizing during the width and height conversions:

```
275    \large%
```

Temporarily prevent underfull \hbox warnings.

```
276    \hbadness=10000\relax%
```

Reset some defaults, possibly will be changed below if options were given:

```
277    \setlength{\LWR@igwidth}{0pt}%
278    \setlength{\LWR@igheight}{0pt}%
279    \renewcommand*{\LWR@igwidthstyle}{}%
280    \renewcommand*{\LWR@igheightstyle}{}%
281    \renewcommand*{\LWR@igorigin}{}%
282    \renewcommand*{\LWR@igangle}{}%
283    \renewcommand*{\LWR@igxscale}{1}%
284    \renewcommand*{\LWR@igyscale}{1}%
285    \renewcommand*{\LWR@igclass}{inlineimage}%
286    \boolfalse{LWR@igkeepaspectratio}%

287    \ifdefvoid{\LWR@ThisAltText}{%
288        \edef\LWR@igalt{\ImageAltText}%
289    }{%
290        \edef\LWR@igalt{\LWR@ThisAltText}%
291    }%
```

If #3 is empty, only one optional argument was given, thus graphicx syntax:

```
292    \IfValueF{#3}{%
293        \IfValueTF{#2}%
294            {\setkeys{igraph}{#2}}%
```

```
295            {\setkeys{igraph}{}}%
296        }%
```

Fully expand and detokenize the filename, changing the file extension to .svg if necessary.

Note that uppercase file extensions are detected and reported as lowercase, so lwarp can only report to the browser lowercase extensions, so all images must have lowercase file extensions.

```
297        \begingroup%
298        \LetLtxMacro\Gin@setfile\LWR@HTML@Gin@setfile%
299        \edef\LWR@tempone{#4}%
```

PDF extensions are removed to allow a search for another graphics format such as SVG or PNG.

```
300        \StrSubstitute{\LWR@tempone}{.pdf}{}[\LWR@tempone]%
301        \StrSubstitute{\LWR@tempone}{.PDF}{}[\LWR@tempone]%

302        \LWR@replaceEPSSVG%
303        \xdef\LWR@parsedfilename{\LWR@tempone}%
304        \Ginclude@graphics{\detokenize\expandafter{\LWR@parsedfilename}}%
305        \endgroup%
306        \filename@parse{\LWR@parsedfilename}%
```

Remove doubled // in the directory path, from the 2020/10/01 LATEX kernel change.

```
307        \StrSubstitute{\LWR@parsedfilename}{//}{/}[\LWR@parsedfilename]%

308        \LWR@traceinfo{LWR@parsedfilename is \LWR@parsedfilename}%
```

If formatting for a word processor, or if using keepaspectratio, find and set the actual image size, without rotation, using PDF instead of SVG to find the original bounding box:

```
309        \ifboolexpr{
310            bool {FormatWP} or
311            bool {LWR@igkeepaspectratio}
312        }{\LWR@ig@useactualimagesize{#1}{#2}{#3}{#4}}{}%
```

Create the HTML reference with the graphicspath, filename, extension, alt tag, style, and class:

```
313        \LWR@traceinfo{LWR@includegraphicsb: about to create href}%
314        \boolfalse{LWR@HTMLsanitize@tmpb@removebackslash}%
315        \LWR@href@partsanitized{\LWR@parsedfilename}%
316        {% start of href
317            \LWR@traceinfo{LWR@includegraphicsb: about to LWR@htmltag}%
318            \LWR@htmltag{\LWR@ig@htmltag}%
319        }% end of href
```

Return to original page size and font size:

```
320        \end{LWR@setvirtualpage}%
```

Clear the single-use alt text:

```
321        \gdef\LWR@ThisAltText{}%
322        \LWR@traceinfo{LWR@includegraphicsb done}%
323 }
```

\includegraphics [⟨*key=val*⟩] {⟨*filename*⟩}

Handles width and height, converted to fixed width and heights.

The user should always use no file suffix in the document source.

```
324 \AtBeginDocument{
325
326 \LWR@traceinfo{Patching includegraphics.}
327
328 \LetLtxMacro\LWR@origincludegraphics\includegraphics
```

```
329 \renewrobustcmd*{\includegraphics}
330 {%
```

This graphic should trigger an HTML paragraph even if alone, so ensure that are doing paragraph handling:

```
331 \LWR@traceinfo{includegraphics}%
332 \LWR@ensuredoingapar%
333 \LWR@includegraphicsb%
334 }% includegraphics
335 }% AtBeginDocument
```

§ 325.5 **Boxes**

\LWR@rotboxorigin   Holds the origin key letters.

```
336 \newcommand*{\LWR@rotboxorigin}{}
```

\LWR@originname   {⟨*letter*⟩}

Given one LATEX origin key value, translate into an HTML origin word:

```
337 \newcommand*{\LWR@originname}[1]{%
338     \ifthenelse{\equal{#1}{t}}{top}{}%
339     \ifthenelse{\equal{#1}{b}}{bottom}{}%
340     \ifthenelse{\equal{#1}{c}}{center}{}%
341     \ifthenelse{\equal{#1}{l}}{left}{}%
342     \ifthenelse{\equal{#1}{r}}{right}{}%
343 }
```

\LWR@originnames   {⟨*letters*⟩}

Given one- or two-letter LATEX origin key values, translate into HTML origin words:

```
344 \newcommand*{\LWR@originnames}[1]{%
345 \StrChar{#1}{1}[\LWR@strresult]%
346 \LWR@originname{\LWR@strresult}
347 \StrChar{#1}{2}[\LWR@strresult]%
348 \LWR@originname{\LWR@strresult}
349 }
```

Handle the origin key for \rotatebox:

```
350 \define@key{krotbox}{origin}{%
351 \renewcommand*{\LWR@rotboxorigin}{#1}%
352 }
```

These keys are ignored:

```
353 \define@key{krotbox}{x}{}
354 \define@key{krotbox}{y}{}
355 \define@key{krotbox}{units}{}
```

\rotatebox [⟨*keyval list*⟩] {⟨*angle*⟩} {⟨*text*⟩}

356 \AtBeginDocument{

The HTML version:

357 \NewDocumentCommand{\LWR@HTML@rotatebox}{O{} m +m}{%

Reset the origin to "none-given":

358 \renewcommand*{\LWR@rotboxorigin}{}

Process the optional keys, which may set \LWR@rotateboxorigin:

359 \setkeys{krotbox}{#1}%

Select `inline-block` so that HTML will transform this span:

```
360 \LWR@htmltagc{%
361     span\LWR@indentHTML
362     style=\textquotedbl\LWR@indentHTML
363     display: inline-block;\LWR@indentHTML
```

If an origin was given, translate and print the origin information:

```
364     \ifthenelse{\NOT\equal{\LWR@rotboxorigin}{}}%
365       {transform-origin: \LWR@originnames{\LWR@rotboxorigin};\LWR@indentHTML}%
366         {}%
```

Print the rotation information:

```
367     \LWR@rotstyle{-ms-}{#2}\LWR@indentHTML
368     \LWR@rotstyle{-webkit-}{#2}\LWR@indentHTML
369     \LWR@rotstyle{}{#2}\textquotedbl\LWR@orignewline%
370 }\LWR@orignewline%
```

Print the text to be rotated:

```
371 \begin{LWR@nestspan}%
372 #3%
```

Close the span:

```
373 \LWR@htmltagc{/span}%
374 \end{LWR@nestspan}%
375 }
```

The high-level interface:

```
376 \LWR@formatted{rotatebox}
377
378 }% AtBeginDocument
```

\scalebox {⟨*h-scale*⟩} [⟨*v-scale*⟩] {⟨*text*⟩}

379 \AtBeginDocument{

The HTML version:

```
380 \NewDocumentCommand{\LWR@HTML@scalebox}{m o m}{%
```

Select `inline-block` so that HTML will transform this span:

```
381 \LWR@htmltagc{%
382     span\LWR@indentHTML
383     style=\textquotedbl\LWR@indentHTML
384     display: inline-block;\LWR@indentHTML
```

Print the scaling information:

```
385     \LWR@scalestyle{-ms-}{#1}{\IfNoValueTF{#2}{#1}{#2}}\LWR@indentHTML
386     \LWR@scalestyle{-webkit-}{#1}{\IfNoValueTF{#2}{#1}{#2}}\LWR@indentHTML
387     \LWR@scalestyle{}{#1}{\IfNoValueTF{#2}{#1}{#2}}
388     \textquotedbl\LWR@orignewline
389 }\LWR@orignewline%
```

Print the text to be scaled:

```
390 \begin{LWR@nestspan}%
391 #3%
```

Close the span:

```
392 \LWR@htmltagc{/span}%
393 \end{LWR@nestspan}%
394 }
```

The high-level interface:

```
395 \LWR@formatted{scalebox}
396
397 }% AtBeginDocument
```

$\reflectbox\ \{\langle text\rangle\}$

```
398 \AtBeginDocument{
399
400 \newcommand{\LWR@HTML@reflectbox}[1]{%
401     \scalebox{-1}[1]{#1}%
402 }% \reflectbox
403
404 \LWR@formatted{reflectbox}
405
406 }% AtBeginDocument
```

$\resizebox\ \{\langle h\text{-}length\rangle\}\ \{\langle v\text{-}length\rangle\}\ \{\langle text\rangle\}$

Simply prints its text argument.

```
407 \AtBeginDocument{
408
409 \NewDocumentCommand{\LWR@HTML@resizebox}{s m m m}{%
410     #4%
411 }
```

```
412
413 \LWR@formatted{resizebox}
414
415 }% AtBeginDocument
```

File 214   **lwarp-graphicx.sty**

§ 326   Package   # graphicx

graphicx (*Pkg*)   graphicx is emulated.

graphicx loads graphics, which also loads lwarp-graphics, which remembers the original graphics definitions for use inside a lateximage, and then patches them \AtBeginDocument for HTML output.

lwarp-graphics handles the syntax of either graphics or graphicx.

for HTML output: `1 \LWR@ProvidesPackagePass{graphicx}[2020/09/09]`

File 215   **lwarp-grffile.sty**

§ 327   Package   # grffile

grffile (*Pkg*)   grffile is supported as-is. File types known to the browser are displayed, and unknown file types are given a link. Each PDF image for print mode should be

⚠ matching PDF and SVG   accompanied by an SVG, PNG, or JPG version for HTML.

lwarp-grffile now exists as a placeholder since grffile used to be emulated by lwarp, and thus older versions of lwarp-grffile may exist and should be overwritten by this newer version.

for HTML output: `1 \LWR@ProvidesPackagePass{grffile}[2017/06/30]`

File 216   **lwarp-grid.sty**

§ 328   Package   # grid

grid (*Pkg*)   grid is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{grid}[2009/06/16]`

`2 \newenvironment*{gridenv}{}{}`

File 217   **lwarp-grid-system.sty**

§ 329   Package   # grid-system

(*Emulates or patches code by* MARCUS BITZL.)

grid-system (*Pkg*) grid-system is patched for use by lwarp.

for HTML output: 1 `\LWR@ProvidesPackagePass{grid-system}[2014/02/16]`

(`\ifdef` is in case the older syntax is removed.)

```
2 \AtBeginEnvironment{Row}{\setlength{\linewidth}{6in}}
3
4 \ifdef{\endrow}{
5     \AtBeginEnvironment{row}{\setlength{\linewidth}{6in}}
6 }{}
7
8 \renewcommand{\gridsystem@finishcell}{\hspace{\gridsystem@cellsep}}
```

File 218 **`lwarp-gridset.sty`**

§ 330   Package   # gridset

gridset (*Pkg*) gridset is ignored.

for HTML output: 1 `\LWR@ProvidesPackageDrop{gridset}[2020-02-12]`

```
2 \newcommand*{\gridbase}{}
3 \newcommand*{\gridinterval}{}
4 \newcommand*{\SavePos}[1]{}
5 \ifLuaTeX
6 \else
7 \let\savepos\SavePos
8 \fi
9 \newcommand*{\vskipnextgrid}{}
10 \newcommand*{\thegridinfo}[1]{(thegridinfo)}
11 \newcommand*{\theposinfo}[1]{(theposinfo)}
12 \newcommand*{\theypos}[1]{(theypos)}
```

File 219 **`lwarp-hang.sty`**

§ 331   Package   # hang

(*Emulates or patches code by* ANDREAS NOLDA.)

hang (*Pkg*) hang is emulated.

for HTML output: 1 `\LWR@ProvidesPackageDrop{hang}[2017/02/18]`

```
2 \newlength{\hangingindent}
3 \setlength{\hangingindent}{1em}
4 \newlength{\hangingleftmargin}
5 \setlength{\hangingleftmargin}{0em}
6
7 \newcommand*{\LWR@findhangingleftmargin}{%
8 \setlength{\LWR@templengthone}{\hangingleftmargin}%
9 \addtolength{\LWR@templengthone}{\hangingindent}%
10 }
```

```
11
12 \newenvironment{hangingpar}
13 {
14     \LWR@findhangingleftmargin%
15     \BlockClass[%
16       \LWR@print@mbox{margin-left:\LWR@printlength{\LWR@templengthone}} ; %
17       \LWR@print@mbox{text-indent:-\LWR@printlength{\hangingindent}}%
18     ]%
19     {hangingpar}%
20 }
21 {\endBlockClass}
22


23 \newenvironment{hanginglist}
24 {%
25     \renewcommand*{\LWR@printcloselist}{\LWR@printcloseitemize}%
26     \renewcommand*{\LWR@printopenlist}{%
27       \LWR@findhangingleftmargin%
28       ul % space
29       class=\textquotedbl{}hanging\textquotedbl{} % space
30       style=\textquotedbl%
31         \LWR@print@mbox{list-style-type:none;} % extra space
32         \LWR@print@mbox{%
33           margin-left:\LWR@printlength{\LWR@templengthone}%
34         } ; % extra space
35         \LWR@print@mbox{%
36           text-indent:-\LWR@printlength{\hangingindent}%
37         }%
38       \textquotedbl%
39     }%
40     \LetLtxMacro\item\LWR@itemizeitem%
41     \list{}{}%
42 }
43 {\endlist}
44
45 \newenvironment{compacthang}
46 {\hanginglist}
47 {\endhanginglist}
48
49 \newlength{\labeledleftmargin}
50 \setlength{\labeledleftmargin}{0em}
51
52 \newenvironment{labeledpar}[2]
53 {%
54     \BlockClass[%
55       \LWR@findhangingleftmargin%
56       \LWR@print@mbox{margin-left:\LWR@printlength{\LWR@templengthone}} ; %
57       \LWR@print@mbox{text-indent:-\LWR@printlength{\hangingindent}}%
58     ]{labeledpar}%
59     \InlineClass{labeledparlabel}{#2}%
60 }
61 {\endBlockClass}
62
63 \newenvironment{labeledlist}[1]
64 {\hanginglist}
65 {\endhanginglist}
66
67 \newenvironment{compactlabel}[1]
68 {\hanginglist}
69 {\endhanginglist}
```

File 220 **`lwarp-hanging.sty`**

§ 332   Package   # hanging

hanging (*Pkg*)   hanging is emulated.

for HTML output: 1 `\LWR@ProvidesPackageDrop{hanging}[2009/09/02]`

2 `\IfClassLoadedTF{memoir}{`
3 `\let\hangpara\relax`
4 `\let\hangparas\relax`
5 `\let\endhangparas\relax`
6 `\let\hangpunct\relax`
7 `\let\endhangpunct\relax`
8 `}{}`

\hangpara   {⟨*indent*⟩} {⟨*afternum*⟩}

Use hangparas instead.

9 `\newcommand*{\hangpara}[2]{}`

Env   hangparas   {⟨*indent*⟩} {⟨*afternum*⟩}

10 `\newenvironment*{hangparas}[2]`
11 `{%`
12 `    \BlockClass[%`
13 `        \LWR@print@mbox{margin-left:\LWR@printlength{#1}} ; %`
14 `        \LWR@print@mbox{text-indent:-\LWR@printlength{#1}}%`
15 `    ]%`
16 `    {hangingpar}%`
17 `}`
18 `{\endBlockClass}`

Env   hangpunct

19 `\newenvironment*{hangpunct}`
20 `{\BlockClass{hangpunct}}`
21 `{\endBlockClass}`

22 `\newcommand{\nhpt}{.}`
23 `\newcommand{\nhlq}{'}`
24 `\newcommand{\nhrq}{'}`

File 221   **`lwarp-hepunits.sty`**

§ 333   Package   # hepunits

(*Emulates or patches code by* ANDY BUCKLEY.)

hepunits (*Pkg*)   hepunits is used as-is, and emulated for MATHJAX.

for HTML output: 1 `\LWR@ProvidesPackagePass{hepunits}[2020/04/10]`

```
2 \begin{warpMathJax}
3 \LWR@infoprocessingmathjax{hepunits}
4
5 \ifx\@HEPopt@sicmds\@yes
6 \CustomizeMathJax{\newcommand{\micron}{\micro\metre}}
7 \CustomizeMathJax{\newcommand{\mrad}{\milli\radian}}
8 \fi
9
10 \CustomizeMathJax{\newcommand{\gauss}{\mathrm{G}}}
11
12 \CustomizeMathJax{\newcommand{\invcmsq}{\centi\metre\tothe{-2}}}
13 \CustomizeMathJax{\newcommand{\invcmsqpersecond}{\invcmsq\second\tothe{-1}}}
14 \CustomizeMathJax{\newcommand{\invcmsqpersec}{\invcmsqpersecond}}
15
16 %% (Inverse) cross-sections
17 \CustomizeMathJax{\newcommand{\invbarn}{\barn\tothe{-1}}}
18
19 \ifx\@HEPopt@noprefixcmds\@empty
20 \CustomizeMathJax{\newcommand{\millibarn}{\milli\barn}}
21 \CustomizeMathJax{\newcommand{\microbarn}{\micro\barn}}
22 \CustomizeMathJax{\newcommand{\nanobarn}{\nano\barn}}
23 \CustomizeMathJax{\newcommand{\picobarn}{\pico\barn}}
24 \CustomizeMathJax{\newcommand{\femtobarn}{\femto\barn}}
25 \CustomizeMathJax{\newcommand{\attobarn}{\atto\barn}}
26 \CustomizeMathJax{\newcommand{\zeptobarn}{\zepto\barn}}
27 \CustomizeMathJax{\newcommand{\yoctobarn}{\yocto\barn}}
28 \CustomizeMathJax{\newcommand{\invnanobarn}{\nano\invbarn}}
29 \CustomizeMathJax{\newcommand{\invpicobarn}{\pico\invbarn}}
30 \CustomizeMathJax{\newcommand{\invfemtobarn}{\femto\invbarn}}
31 \CustomizeMathJax{\newcommand{\invattobarn}{\atto\invbarn}}
32 \CustomizeMathJax{\newcommand{\invzeptobarn}{\zepto\invbarn}}
33 \CustomizeMathJax{\newcommand{\invyoctobarn}{\yocto\invbarn}}
34 \CustomizeMathJax{\newcommand{\invnb}{\invnanobarn}}
35 \CustomizeMathJax{\newcommand{\invpb}{\invpicobarn}}
36 \CustomizeMathJax{\newcommand{\invfb}{\invfemtobarn}}
37 \CustomizeMathJax{\newcommand{\invab}{\invattobarn}}
38 \CustomizeMathJax{\newcommand{\invzb}{\invzeptobarn}}
39 \CustomizeMathJax{\newcommand{\invyb}{\invyoctobarn}}
40 \fi
41
42 \CustomizeMathJax{\newcommand{\electronvoltc}{\electronvolt\per\mathit{c}}}
43 \CustomizeMathJax{\newcommand{\electronvoltcsq}{\electronvolt\per\mathit{c}\squared}}
44 \CustomizeMathJax{\let\eVc\electronvoltc}
45 \CustomizeMathJax{\let\eVcsq\electronvoltcsq}
46
47 \ifx\@HEPopt@noprefixcmds\@empty
48 \CustomizeMathJax{\newcommand{\meV}{\milli\eV}}
49 \CustomizeMathJax{\newcommand{\keV}{\kilo\eV}}
50 \CustomizeMathJax{\newcommand{\MeV}{\mega\eV}}
51 \CustomizeMathJax{\newcommand{\GeV}{\giga\eV}}
52 \CustomizeMathJax{\newcommand{\TeV}{\tera\eV}}
53 \CustomizeMathJax{\newcommand{\meVc}{\milli\eVc}}
54 \CustomizeMathJax{\newcommand{\keVc}{\kilo\eVc}}
55 \CustomizeMathJax{\newcommand{\MeVc}{\mega\eVc}}
56 \CustomizeMathJax{\newcommand{\GeVc}{\giga\eVc}}
57 \CustomizeMathJax{\newcommand{\TeVc}{\tera\eVc}}
58 \CustomizeMathJax{\newcommand{\meVcsq}{\milli\eVcsq}}
59 \CustomizeMathJax{\newcommand{\keVcsq}{\kilo\eVcsq}}
60 \CustomizeMathJax{\newcommand{\MeVcsq}{\mega\eVcsq}}
61 \CustomizeMathJax{\newcommand{\GeVcsq}{\giga\eVcsq}}
```

```
62 \CustomizeMathJax{\newcommand{\TeVcsq}{\tera\eVcsq}}
63 \fi
64 \end{warpMathJax}
```

File 222   `lwarp-hhline.sty`

§ 334   Package   # hhline

(*Emulates or patches code by* DAVID CARLISLE.)

hhline (*Pkg*)   hhline is patched for use by lwarp.

Only a rudimentary emulation is provided so far. If the argument contains any = characters, the result is a double \hline. If none, the result is a single \hline.

for HTML output: 1 \LWR@ProvidesPackagePass{hhline}[2014/10/28]

```
2 \newrobustcmd*{\LWR@HTML@hhline}[1]{%
3     \edef\LWR@tempone{\detokenize\expandafter{#1}}%
4     \IfSubStr[1]{\LWR@tempone}{=}{\hline\hline}{\hline}%
5 }
6 % ^^A or:
7 % ^^A \newrobustcmd*{\LWR@HTML@hhline}[1]{\LWR@getmynexttoken}
8
9 \AtBeginDocument{\LWR@expandableformatted{hhline}}
```

For MATHJAX. A simple \hline is used.

```
10 \begin{warpMathJax}
11 \CustomizeMathJax{\newcommand{\hhline}[1]{\hline}}
12 \end{warpMathJax}
```

File 223   `lwarp-hhtensor.sty`

§ 335   Package   # hhtensor

(*Emulates or patches code by* HARALD HARDERS.)

hhtensor (*Pkg*)   hhtensor is used as-is, and emulated for MATHJAX.

for HTML output: 1 \LWR@ProvidesPackagePass{hhtensor}[2011/12/29]

```
2 \begin{warpMathJax}
3 \iftensor@bold
4   \CustomizeMathJax{\newcommand{\vec}[1]{\boldsymbol{#1}}}
5   \CustomizeMathJax{\newcommand{\matr}[1]{\boldsymbol{#1}}}
6   \CustomizeMathJax{\newcommand{\tens}[2]{\boldsymbol{#1}}}
7 \else
8   \iftensor@uline
9     \CustomizeMathJax{\newcommand{\vec}[1]{\ushort{#1}}}
10    \CustomizeMathJax{\newcommand{\matr}[1]{\ushortd{#1}}}
11    \CustomizeMathJax{\newcommand{\tens}[2]{
12       \underset{
13          \raise{.5ex}{\underset{#2}{\sim}}}
```

```
14         }{#1}
15      }}
16    \else
17      \CustomizeMathJax{\newcommand{\matr}[1]{\vec{\vec{#1}}}}
18      \CustomizeMathJax{\newcommand{\tens}[2]{
19          \underset{
20              \raise{.5ex}{\underset{#2}{\sim}}
21          }{#1}
22      }}
23    \fi
24 \fi
25 \CustomizeMathJax{\newcommand{\dcdot}{\mathrel{\cdot\mkern 0.0mu \cdot}}}
26 \CustomizeMathJax{\newcommand{\trans}{{}^{\mathrm{T}}}}
27 \end{warpMathJax}
```

File 224   **`lwarp-hypbmsec.sty`**

§ 336   Package   # hypbmsec

hypbmsec (*Pkg*)   hypbmsec is emulated by the lwarp core.

for HTML output: 1 \LWR@ProvidesPackageDrop{hypbmsec}[2016/05/16]

File 225   **`lwarp-hypcap.sty`**

§ 337   Package   # hypcap

hypcap (*Pkg*)   hypcap is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{hypcap}[2016/05/16]

```
2 \newcommand*{\capstart}{}
3 \newcommand*{\hypcapspace}{}
4 \newcommand*{\hypcapredef}[1]{}
5 \newcommand*{\capstartfalse}{}
6 \newcommand*{\capstarttrue}{}
```

File 226   **`lwarp-hypdestopt.sty`**

§ 338   Package   # hypdestopt

hypdestopt (*Pkg*)   hypdestopt is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{hypdestopt}[2016/05/21]

File 227   **`lwarp-hypernat.sty`**

§ 339   Package   # hypernat

hypernat (*Pkg*)   hypernat is ignored.

1 \LWR@ProvidesPackageDrop{hypernat}[2001/07/09]

File 228   **`lwarp-hyperref.sty`**

§ 340   Package   # hyperref

(*Emulates or patches code by* Sebastian Rahtz, Heiko Oberdiek, The LaTeX3 Project.)

hyperref (*Pkg*)   hyperref is emulated.

```
 1 % \LWR@ProvidesPackageDrop{hyperref}% not allowed
 2 % \ProvidesPackage{lwarp-#1-#2}% not allowed
 3 \PackageInfo{lwarp}{%
 4 Using the lwarp HTML version of package `hyperref',\MessageBreak
 5 and discarding options except backref, pagebackref.\MessageBreak
 6 (Not using \protect\ProvidesPackage, so that other packages\MessageBreak
 7 do not attempt to patch lwarp's version of `hyperref'.)\MessageBreak}
 8 \SetupKeyvalOptions{family=LWR@hyperref,prefix=LWR@hyperref@}
 9
10 \newcommand{\hypersetup}[1]{\setkeys{LWR@hyperref}{#1}}
11
12 \define@key{LWR@hyperref}{a4paper}[]{}
13 \define@key{LWR@hyperref}{a5paper}[]{}
14 \define@key{LWR@hyperref}{b5paper}[]{}
15 \define@key{LWR@hyperref}{letterpaper}[]{}
16 \define@key{LWR@hyperref}{legalpaper}[]{}
17 \define@key{LWR@hyperref}{executivepaper}[]{}
18 \define@key{LWR@hyperref}{implicit}[]{}
19 \define@key{LWR@hyperref}{draft}[]{}
20 \define@key{LWR@hyperref}{final}[]{}
21 \define@key{LWR@hyperref}{setpagesize}[]{}
22 \define@key{LWR@hyperref}{debug}[]{}
23 \define@key{LWR@hyperref}{linktocpage}[]{}
24 \define@key{LWR@hyperref}{linktoc}[]{}
25 \define@key{LWR@hyperref}{extension}[]{}
26 \define@key{LWR@hyperref}{verbose}[]{}
27 \define@key{LWR@hyperref}{typexml}[]{}
28 \define@key{LWR@hyperref}{raiselinks}[]{}
29 \define@key{LWR@hyperref}{breaklinks}[]{}
30 \define@key{LWR@hyperref}{localanchorname}[]{}
31 \define@key{LWR@hyperref}{pageanchor}[]{}
32 \define@key{LWR@hyperref}{plainpages}[]{}
33 \define@key{LWR@hyperref}{naturalnames}[]{}
34 \define@key{LWR@hyperref}{hypertexnames}[]{}
35 \define@key{LWR@hyperref}{nesting}[]{}
36 \define@key{LWR@hyperref}{destlabel}[]{}
37 \define@key{LWR@hyperref}{unicode}[]{}
38 \define@key{LWR@hyperref}{pdfencoding}[]{}
39 \define@key{LWR@hyperref}{psdextra}[]{}
40 \define@key{LWR@hyperref}{pdfversion}[]{}
41 \define@key{LWR@hyperref}{dvipdfmx-outline-open}[]{}
42 \define@key{LWR@hyperref}{driverfallback}[]{}
43 \define@key{LWR@hyperref}{customdriver}[]{}
44 \define@key{LWR@hyperref}{hyperfigures}[]{}
45 \define@key{LWR@hyperref}{hyperfootnotes}[]{}
46 \define@key{LWR@hyperref}{hyperindex}[]{}
```

```
47 \define@key{LWR@hyperref}{encap}[]{}
48 \define@key{LWR@hyperref}{colorlinks}[]{}
49 \define@key{LWR@hyperref}{ocgcolorlinks}[]{}
50 \define@key{LWR@hyperref}{frenchlinks}[]{}
51 \define@key{LWR@hyperref}{bookmarks}[]{}
52 \define@key{LWR@hyperref}{bookmarksopen}[]{}
53 \define@key{LWR@hyperref}{bookmarksdepth}[]{}
54 \define@key{LWR@hyperref}{bookmarksopenlevel}[]{}
55 \define@key{LWR@hyperref}{bookmarkstype}[]{}
56 \define@key{LWR@hyperref}{bookmarksnumbered}[]{}
57 \define@key{LWR@hyperref}{CJKbookmarks}[]{}
58 \define@key{LWR@hyperref}{link}[]{}
59 \define@key{LWR@hyperref}{anchor}[]{}
60 \define@key{LWR@hyperref}{cite}[]{}
61 \define@key{LWR@hyperref}{file}[]{}
62 \define@key{LWR@hyperref}{url}[]{}
63 \define@key{LWR@hyperref}{menu}[]{}
64 \define@key{LWR@hyperref}{run}[]{}
65 \define@key{LWR@hyperref}{linkbordercolor}[]{}
66 \define@key{LWR@hyperref}{anchorbordercolor}[]{}
67 \define@key{LWR@hyperref}{citebordercolor}[]{}
68 \define@key{LWR@hyperref}{filebordercolor}[]{}
69 \define@key{LWR@hyperref}{urlbordercolor}[]{}
70 \define@key{LWR@hyperref}{menubordercolor}[]{}
71 \define@key{LWR@hyperref}{runbordercolor}[]{}
72 \define@key{LWR@hyperref}{pagecolor}[]{}
73 \define@key{LWR@hyperref}{baseurl}[]{}
74 \define@key{LWR@hyperref}{linkfileprefix}[]{}
75 \define@key{LWR@hyperref}{pdfpagetransition}[]{}
76 \define@key{LWR@hyperref}{pdfpageduration}[]{}
77 \define@key{LWR@hyperref}{pdfpagehidden}[]{}
78 \define@key{LWR@hyperref}{pagebordercolor}[]{}
79 \define@key{LWR@hyperref}{allbordercolors}[]{}
80 \define@key{LWR@hyperref}{pdfhighlight}[]{}
81 \define@key{LWR@hyperref}{pdfborder}[]{}
82 \define@key{LWR@hyperref}{pdfborderstyle}[]{}
83 \define@key{LWR@hyperref}{pdfprintpagerange}[]{}
84 \define@key{LWR@hyperref}{pdfusetitle}[]{}
85 \define@key{LWR@hyperref}{pdftitle}[]{}
86 \define@key{LWR@hyperref}{pdfauthor}[]{}
87 \define@key{LWR@hyperref}{pdfproducer}[]{}
88 \define@key{LWR@hyperref}{pdfcreator}[]{}
89 \define@key{LWR@hyperref}{addtopdfcreator}[]{}
90 \define@key{LWR@hyperref}{pdfcreationdate}[]{}
91 \define@key{LWR@hyperref}{pdfmoddate}[]{}
92 \define@key{LWR@hyperref}{pdfsubject}[]{}
93 \define@key{LWR@hyperref}{pdfkeywords}[]{}
94 \define@key{LWR@hyperref}{pdftrapped}[]{}
95 \define@key{LWR@hyperref}{pdfinfo}[]{}
96 \define@key{LWR@hyperref}{pdfview}[]{}
97 \define@key{LWR@hyperref}{pdflinkmargin}[]{}
98 \define@key{LWR@hyperref}{pdfstartpage}[]{}
99 \define@key{LWR@hyperref}{pdfstartview}[]{}
100 \define@key{LWR@hyperref}{pdfremotestartview}[]{}
101 \define@key{LWR@hyperref}{pdfpagescrop}[]{}
102 \define@key{LWR@hyperref}{pdftoolbar}[]{}
103 \define@key{LWR@hyperref}{pdfmenubar}[]{}
104 \define@key{LWR@hyperref}{pdfwindowui}[]{}
105 \define@key{LWR@hyperref}{pdffitwindow}[]{}
106 \define@key{LWR@hyperref}{pdfcenterwindow}[]{}
```

```
107 \define@key{LWR@hyperref}{pdfdisplaydoctitle}[]{}
108 \define@key{LWR@hyperref}{pdfa}[]{}
109 \define@key{LWR@hyperref}{pdfnewwindow}[]{}
110 \define@key{LWR@hyperref}{pdflang}[]{}
111 \define@key{LWR@hyperref}{pdfpagelabels}[]{}
112 \define@key{LWR@hyperref}{pdfescapeform}[]{}
113 \define@key{LWR@hyperref}{english}[]{}
114 \define@key{LWR@hyperref}{UKenglish}[]{}
115 \define@key{LWR@hyperref}{british}[]{}
116 \define@key{LWR@hyperref}{USenglish}[]{}
117 \define@key{LWR@hyperref}{american}[]{}
118 \define@key{LWR@hyperref}{german}[]{}
119 \define@key{LWR@hyperref}{austrian}[]{}
120 \define@key{LWR@hyperref}{ngerman}[]{}
121 \define@key{LWR@hyperref}{naustrian}[]{}
122 \define@key{LWR@hyperref}{russian}[]{}
123 \define@key{LWR@hyperref}{brazil}[]{}
124 \define@key{LWR@hyperref}{brazilian}[]{}
125 \define@key{LWR@hyperref}{portuguese}[]{}
126 \define@key{LWR@hyperref}{spanish}[]{}
127 \define@key{LWR@hyperref}{catalan}[]{}
128 \define@key{LWR@hyperref}{afrikaans}[]{}
129 \define@key{LWR@hyperref}{french}[]{}
130 \define@key{LWR@hyperref}{frenchb}[]{}
131 \define@key{LWR@hyperref}{francais}[]{}
132 \define@key{LWR@hyperref}{acadian}[]{}
133 \define@key{LWR@hyperref}{canadien}[]{}
134 \define@key{LWR@hyperref}{italian}[]{}
135 \define@key{LWR@hyperref}{magyar}[]{}
136 \define@key{LWR@hyperref}{hungarian}[]{}
137 \define@key{LWR@hyperref}{greek}[]{}
138 \define@key{LWR@hyperref}{dutch}[]{}
139 \define@key{LWR@hyperref}{tex4ht}[]{}
140 \define@key{LWR@hyperref}{pdftex}[]{}
141 \define@key{LWR@hyperref}{luatex}[]{}
142 \define@key{LWR@hyperref}{nativepdf}[]{}
143 \define@key{LWR@hyperref}{dvipdfm}[]{}
144 \define@key{LWR@hyperref}{dvipdfmx}[]{}
145 \define@key{LWR@hyperref}{xetex}[]{}
146 \define@key{LWR@hyperref}{pdfmark}[]{}
147 \define@key{LWR@hyperref}{dvips}[]{}
148 \define@key{LWR@hyperref}{hypertex}[]{}
149 \define@key{LWR@hyperref}{vtex}[]{}
150 \define@key{LWR@hyperref}{vtexpdfmark}[]{}
151 \define@key{LWR@hyperref}{dviwindo}[]{}
152 \define@key{LWR@hyperref}{dvipsone}[]{}
153 \define@key{LWR@hyperref}{textures}[]{}
154 \define@key{LWR@hyperref}{latex2html}[]{}
155 \define@key{LWR@hyperref}{ps2pdf}[]{}
156 \define@key{LWR@hyperref}{vietnamese}[]{}
157 \define@key{LWR@hyperref}{vietnam}[]{}
158 \define@key{LWR@hyperref}{arabic}[]{}
159 \define@key{LWR@hyperref}{hidelinks}[]{}
160 \define@key{LWR@hyperref}{draft}[]{}
161 \define@key{LWR@hyperref}{nolinks}[]{}
162 \define@key{LWR@hyperref}{final}[]{}
163 \define@key{LWR@hyperref}{pdfa}[]{}
164 \define@key{LWR@hyperref}{pdfversion}[]{}
165 \define@key{LWR@hyperref}{typexml}[]{}
166 \define@key{LWR@hyperref}{tex4ht}[]{}
```

```
167 \define@key{LWR@hyperref}{pdftex}[]{}
168 \define@key{LWR@hyperref}{nativepdf}[]{}
169 \define@key{LWR@hyperref}{dvipdfm}[]{}
170 \define@key{LWR@hyperref}{dvipdfmx}[]{}
171 \define@key{LWR@hyperref}{dvipdfmx-outline-open}[]{}
172 \define@key{LWR@hyperref}{pdfmark}[]{}
173 \define@key{LWR@hyperref}{dvips}[]{}
174 \define@key{LWR@hyperref}{hypertex}[]{}
175 \define@key{LWR@hyperref}{vtex}[]{}
176 \define@key{LWR@hyperref}{vtexpdfmark}[]{}
177 \define@key{LWR@hyperref}{dviwindo}[]{}
178 \define@key{LWR@hyperref}{dvipsone}[]{}
179 \define@key{LWR@hyperref}{textures}[]{}
180 \define@key{LWR@hyperref}{latex2html}[]{}
181 \define@key{LWR@hyperref}{ps2pdf}[]{}
182 \define@key{LWR@hyperref}{xetex}[]{}
183 \define@key{LWR@hyperref}{driverfallback}[]{}
184 \define@key{LWR@hyperref}{customdriver}[]{}
185 \define@key{LWR@hyperref}{pdfversion}[]{}
186 \define@key{LWR@hyperref}{bookmarks}[]{}
187 \define@key{LWR@hyperref}{ocgcolorlinks}[]{}
188 \define@key{LWR@hyperref}{colorlinks}[]{}
189 \define@key{LWR@hyperref}{frenchlinks}[]{}
190 \define@key{LWR@hyperref}{backref}[]{}
191 \define@key{LWR@hyperref}{pagebackref}[]{}
192 \define@key{LWR@hyperref}{destlabel}[]{}
193 \define@key{LWR@hyperref}{pdfpagescrop}[]{}
194 \define@key{LWR@hyperref}{pdfpagemode}[]{}
195 \define@key{LWR@hyperref}{pdfnonfullscreenpagemode}[]{}
196 \define@key{LWR@hyperref}{pdfdirection}[]{}
197 \define@key{LWR@hyperref}{pdfviewarea}[]{}
198 \define@key{LWR@hyperref}{pdfviewclip}[]{}
199 \define@key{LWR@hyperref}{pdfprintarea}[]{}
200 \define@key{LWR@hyperref}{pdfprintclip}[]{}
201 \define@key{LWR@hyperref}{pdfprintscaling}[]{}
202 \define@key{LWR@hyperref}{pdfduplex}[]{}
203 \define@key{LWR@hyperref}{pdfpicktraybypdfsize}[]{}
204 \define@key{LWR@hyperref}{pdfprintpagerange}[]{}
205 \define@key{LWR@hyperref}{pdfnumcopies}[]{}
206 \define@key{LWR@hyperref}{pdfstartview}[]{}
207 \define@key{LWR@hyperref}{pdfstartpage}[]{}
208 \define@key{LWR@hyperref}{pdftoolbar}[]{}
209 \define@key{LWR@hyperref}{pdfmenubar}[]{}
210 \define@key{LWR@hyperref}{pdfwindowui}[]{}
211 \define@key{LWR@hyperref}{pdffitwindow}[]{}
212 \define@key{LWR@hyperref}{pdfcenterwindow}[]{}
213 \define@key{LWR@hyperref}{pdfdisplaydoctitle}[]{}
214 \define@key{LWR@hyperref}{pdfpagelayout}[]{}
215 \define@key{LWR@hyperref}{pdflang}[]{}
216 \define@key{LWR@hyperref}{baseurl}[]{}
217 \define@key{LWR@hyperref}{pdfusetitle}[]{}
218 \define@key{LWR@hyperref}{pdfpagelabels}[]{}
219 \define@key{LWR@hyperref}{hyperfootnotes}[]{}
220 \define@key{LWR@hyperref}{hyperfigures}[]{}
221 \define@key{LWR@hyperref}{hyperindex}[]{}
222 \define@key{LWR@hyperref}{encap}[]{}
223 \define@key{LWR@hyperref}{linkcolor}[]{}
224 \define@key{LWR@hyperref}{anchorcolor}[]{}
225 \define@key{LWR@hyperref}{citecolor}[]{}
226 \define@key{LWR@hyperref}{filecolor}[]{}
```

```
227 \define@key{LWR@hyperref}{urlcolor}[]{}
228 \define@key{LWR@hyperref}{menucolor}[]{}
229 \define@key{LWR@hyperref}{runcolor}[]{}
230 \define@key{LWR@hyperref}{allcolors}[]{}
231
232 \DeclareStringOption[false]{backref}[section]
233
234 \DeclareBoolOption{pagebackref}
235
236 \DeclareDefaultOption{}
237
238 \ProcessKeyvalOptions*\relax
```

Maybe load backref:

```
239 \ifdefstring{\LWR@hyperref@backref}{section}
240     {\RequirePackage{backref}}
241     {}
242
243 \ifdefstring{\LWR@hyperref@backref}{slide}
244     {\RequirePackage{backref}}
245     {}
246
247 \ifdefstring{\LWR@hyperref@backref}{page}
248     {\RequirePackage{backref}}
249     {}
250
251 \ifLWR@hyperref@pagebackref
252     \RequirePackage{backref}
253 \fi
```

Emualted:

```
254 \newdimen\XeTeXLinkMargin
255 \newcommand*{\XeTeXLinkBox}[1]{#1}


256 \LetLtxMacro\href\LWR@href
257 \LetLtxMacro\nolinkurl\LWR@nolinkurl
258 \LetLtxMacro\url\LWR@url
259 \LetLtxMacro\phantomsection\LWR@phantomsection


260 \newcommand*{\hyperbaseurl}[1]{}
```

No application for lwarp:

```
261 \newcommand*{\HyperDestNameFilter}[1]{#1}
262 \newcommand*{\HyperDestLabelReplace}[1]{#1}
263 \newcommand*{\HyperDestRename}[2]{}
```

No application for lwarp:

```
264 \newcommand*{\hyperget}[2]{}
```

\hyperimage   {⟨*URL*⟩} {⟨*alt text*⟩}

Insert an image with ALT text. The given ALT text need not be sanitized because
\&, \%, \textless, \textgreater are already sanitized.

```
265 \NewDocumentCommand{\LWR@hyperimageb}{m +m}{%
266     \LWR@ensuredoingapar%
267     \edef\tmpb{#1}%
268     \LWR@HTMLsanitize@tmpb%
269     \LWR@htmltag{%
270         img src=\textquotedbl\tmpb\textquotedbl\ %
271         alt=\textquotedbl#2\textquotedbl\ %
272         class=\textquotedbl{}hyperimage\textquotedbl%
273     }%
274     \LWR@ensuredoingapar%
275     \endgroup%
276 }
277
278 \newrobustcmd*{\hyperimage}{%
279     \begingroup%
280     \LWR@linkcatcodes%
281     \LWR@hyperimageb%
282 }
283
```

\hyperdef   {⟨1: category⟩} {⟨2: name⟩} {⟨3: text⟩}

Creates an HTML anchor to category.name with the given text.

```
284 \NewDocumentCommand{\LWR@hyperdefb}{m m +m}{%
285     \LWR@ensuredoingapar%
286     \LWR@label@createtag{#1.#2}%
287     #3%
288     \endgroup%
289 }
290
291 \newcommand*{\hyperdef}{%
292     \begingroup%
293     \LWR@linkcatcodes%
294     \LWR@hyperdefb%
295 }
296
```

\LWR@hyperrefb   {⟨1: URL⟩} {⟨2: category⟩} {⟨3: name⟩} {⟨4: text⟩}

Creates an HTML link to URL#category.name with the given text.

```
297 \newcommand{\LWR@hyperreffinish}[1]{%
298     \begingroup%
299     \RenewDocumentCommand{\ref}{s m}{\LWR@print@ref{##2}}%
300     #1%
301     \endgroup%
302     \LWR@htmltag{/a}%
303 }
304
305 \newcommand*{\LWR@hyperrefbb}[3]{%
306     \LWR@htmltag{%
307         a href=\textquotedbl%
308             \detokenize\expandafter{#1}\LWR@hashmark%
309             \detokenize\expandafter{#2}.\detokenize\expandafter{#3}%
310         \textquotedbl%
311         \LWR@addlinktitle%
312     }%
313     \endgroup%
314     \LWR@hyperreffinish%
315 }
```

```
316
317 \newrobustcmd*{\LWR@hyperrefb}{%
318     \begingroup%
319     \LWR@linkcatcodes%
320     \LWR@hyperrefbb%
321 }
```

\LWR@hyperrefc   [⟨*label*⟩] {⟨*text*⟩}

Creates text as an HTML link to the LATEX label.

```
322 \NewDocumentCommand{\LWR@hyperrefcb}{O{label}}{%
323     \LWR@startref{#1}%
324     \endgroup%
325     \LWR@hyperreffinish%
326 }
327
328 \newcommand*{\LWR@hyperrefc}{%
329     \begingroup%
330     \LWR@linkcatcodes%
331     \LWR@hyperrefcb%
332 }
```

\hyperref   {⟨*1: URL*⟩} {⟨*2: category*⟩} {⟨*3: name*⟩} {⟨*4: text*⟩}   — or —
[⟨*1: label*⟩] {⟨*2: text*⟩}

```
333 \DeclareRobustCommand*{\hyperref}{%
334     \LWR@ensuredoingapar%
335     \@ifnextchar[\LWR@hyperrefc\LWR@hyperrefb%
336 }
```

\hypertarget   {⟨*name*⟩} {⟨*text*⟩}

Creates an anchor to name with the given text.

```
337 \NewDocumentCommand{\LWR@hypertargetb}{m +m}{%
338     \label{LWR-ht-#1}%
339     #2%
340     \endgroup%
341 }
342
343 \newcommand*{\hypertarget}{%
344     \LWR@ensuredoingapar%
345     \begingroup%
346     \LWR@linkcatcodes%
347     \LWR@hypertargetb%
348 }
```

\hyperlink   {⟨*name*⟩} {⟨*text*⟩}

Creates a link to the anchor created by hypertarget, with the given link text.

Declared because also defined by memoir.

```
349 \DeclareDocumentCommand{\LWR@hyperlinkb}{m}{%
350     \ifbool{LWR@insidemathcomment}%
351         {\endgroup}%
352         {\LWR@hyperrefcb[LWR-ht-#1]}%
353 }
354
355 \DeclareDocumentCommand{\hyperlink}{}{%
```

```
356        \LWR@ensuredoingapar%
357        \begingroup%
358        \LWR@linkcatcodes%
359        \LWR@hyperlinkb%
360 }
```

\LWR@nullify@hyperref   {⟨*1: URL*⟩} {⟨*2: category*⟩} {⟨*3: name*⟩} {⟨*4: text*⟩}   — or —
[⟨*1: label*⟩] {⟨*2: text*⟩}

```
361 \newcommand{\LWR@nullify@hyperrefb}[2][]{}
362
363 \newcommand*{\LWR@nullify@hyperref}{%
364        \@ifnextchar[\LWR@nullify@hyperrefb\@fourthoffour%
365 }
```

To nullify in a lateximage or svg math. \hypertarget must be left active for
references to work, and does not harm.

```
366 \appto\LWR@restoreorigformatting{%
367        \LetLtxMacro\hyperdef\@thirdofthree
368        \LetLtxMacro\hyperlink\@secondoftwo%
369        \LetLtxMacro\hyperref\LWR@nullify@hyperref%
370 }
```

\autoref   * {⟨*label*⟩}

For HTML, \cleveref is used instead.

```
371 \NewDocumentCommand{\autoref}{s m}{%
372        \IfBooleanTF{#1}{\ref{#2}}{\cref{#2}}%
373 }
```

\autopageref   {⟨*label*⟩}

For HTML, \cleveref is used instead.

```
374 \NewDocumentCommand{\autopageref}{s m}{%
375        \IfBooleanTF{#1}{\cpageref{#2}}{\cref{#2}}%
376 }
```

Default names:

```
377 \def\equationautorefname{Equation}%
378 \def\footnoteautorefname{footnote}%
379 \def\itemautorefname{item}%
380 \def\figureautorefname{Figure}%
381 \def\tableautorefname{Table}%
382 \def\partautorefname{Part}%
383 \def\appendixautorefname{Appendix}%
384 \def\chapterautorefname{chapter}%
385 \def\sectionautorefname{section}%
386 \def\subsectionautorefname{subsection}%
387 \def\subsubsectionautorefname{subsubsection}%
388 \def\paragraphautorefname{paragraph}%
389 \def\subparagraphautorefname{subparagraph}%
390 \def\FancyVerbLineautorefname{line}%
391 \def\theoremautorefname{Theorem}%
392 \def\pageautorefname{page}%
```

\pdfstringdef {⟨*macroname*⟩} {⟨*TEXstring*⟩}

393 \newcommand{\pdfstringdef}[2]{}

\pdfbookmark [⟨*level*⟩] {⟨*text*⟩} {⟨*name*⟩}

394 \newcommand{\pdfbookmark}[3][]{}

\currentpdfbookmark {⟨*text*⟩} {⟨*name*⟩}

395 \newcommand{\currentpdfbookmark}[2]{}

\subpdfbookmark {⟨*text*⟩} {⟨*name*⟩}

396 \newcommand{\subpdfbookmark}[2]{}

\belowpdfbookmark {⟨*text*⟩} {⟨*name*⟩}

397 \newcommand{\belowpdfbookmark}[2]{}

\texorpdfstring {⟨*TEXstring*⟩} {⟨*PDFstring*⟩}

398 \let\texorpdfstring\relax
399 \newcommand{\texorpdfstring}[2]{#1}

{⟨*commands*⟩}

\pdfstringdefDisableCommands 400 \newcommand{\pdfstringdefDisableCommands}[1]{}

\hypercalcbp {⟨*dimen*⟩}   From hyperref.

401 \def\hypercalcbp#1{%
402   \strip@pt\dimexpr 0.99626401\dimexpr(#1)\relax\relax
403 }%

\Acrobatmenu {⟨*menuoption*⟩} {⟨*text*⟩}

404 \newcommand{\Acrobatmenu}[2]{}

\TextField [⟨*parameters*⟩] {⟨*label*⟩}

405 \DeclareRobustCommand{\TextField}[2][]{}

\CheckBox [⟨*parameters*⟩] {⟨*label*⟩}

406 \DeclareRobustCommand{\CheckBox}[2][]{}

\ChoiceMenu [⟨*parameters*⟩] {⟨*label*⟩} {⟨*choices*⟩}

407 \DeclareRobustCommand{\ChoiceMenu}[3][]{}

\PushButton [⟨*parameters*⟩] {⟨*label*⟩}

408 \DeclareRobustCommand{\PushButton}[2][]{}

\Submit   [⟨*parameters*⟩] {⟨*label*⟩}

409 \DeclareRobustCommand{\Submit}[2][]{}

\Reset   [⟨*parameters*⟩] {⟨*label*⟩}

410 \DeclareRobustCommand{\Reset}[2][]{}

\Gauge   [⟨*parameters*⟩] {⟨*label*⟩}

411 \DeclareRobustCommand{\Gauge}[2][]{}

\LayoutTextField   {⟨*label*⟩} {⟨*field*⟩}

412 \newcommand*{\LayoutTextField}[2]{}

\LayoutChoiceField   {⟨*label*⟩} {⟨*field*⟩}

413 \newcommand*{\LayoutChoiceField}[2]{}

\LayoutCheckField   {⟨*label*⟩} {⟨*field*⟩}

414 \newcommand*{\LayoutCheckField}[2]{}

\MakeRadioField   {⟨*width*⟩} {⟨*height*⟩}

415 \newcommand*{\MakeRadioField}[2]{}

\MakeCheckField   {⟨*width*⟩} {⟨*height*⟩}

416 \newcommand*{\MakeCheckField}[2]{}

\MakeTextField   {⟨*width*⟩} {⟨*height*⟩}

417 \newcommand*{\MakeTextField}[2]{}

\MakeChoiceField   {⟨*width*⟩} {⟨*height*⟩}

418 \newcommand*{\MakeChoiceField}[2]{}

\MakeFieldButton   {⟨*text*⟩}

419 \newcommand{\MakeFieldButton}[1]{}

File 229   **`lwarp-hyperxmp.sty`**

§ 341   Package   # hyperxmp

hyperxmp (*Pkg*)   hyperxmp is ignored.

for HTML output: Discard all options for lwarp-hyperxmp:

```
1 \LWR@ProvidesPackageDrop{hyperxmp}[2018/11/27]
2
3 \define@key{LWR@hyperref}{pdfdate}[]{}
4 \define@key{LWR@hyperref}{pdfmetadate}[]{}
5 \define@key{LWR@hyperref}{pdfcopyright}[]{}
6 \define@key{LWR@hyperref}{pdftype}[]{}
7 \define@key{LWR@hyperref}{pdflicenseurl}[]{}
8 \define@key{LWR@hyperref}{pdfauthortitle}[]{}
9 \define@key{LWR@hyperref}{pdfcaptionwriter}[]{}
10 \define@key{LWR@hyperref}{pdfmetalang}[]{}
11 \define@key{LWR@hyperref}{pdfapart}[]{}
12 \define@key{LWR@hyperref}{pdfaconformance}[]{}
13 \define@key{LWR@hyperref}{pdfuapart}[]{}
14 \define@key{LWR@hyperref}{pdfxstandard}[]{}
15 \define@key{LWR@hyperref}{pdfsource}[]{}
16 \define@key{LWR@hyperref}{pdfdocumentid}[]{}
17 \define@key{LWR@hyperref}{pdfinstanceid}[]{}
18 \define@key{LWR@hyperref}{pdfversionid}[]{}
19 \define@key{LWR@hyperref}{pdfrendition}[]{}
20 \define@key{LWR@hyperref}{pdfpublication}[]{}
21 \define@key{LWR@hyperref}{pdfpubtype}[]{}
22 \define@key{LWR@hyperref}{pdfbytes}[]{}
23 \define@key{LWR@hyperref}{pdfnumpages}[]{}
24 \define@key{LWR@hyperref}{pdfissn}[]{}
25 \define@key{LWR@hyperref}{pdfeissn}[]{}
26 \define@key{LWR@hyperref}{pdfisbn}[]{}
27 \define@key{LWR@hyperref}{pdfbookedition}[]{}
28 \define@key{LWR@hyperref}{pdfpublisher}[]{}
29 \define@key{LWR@hyperref}{pdfvolumenum}[]{}
30 \define@key{LWR@hyperref}{pdfissuenum}[]{}
31 \define@key{LWR@hyperref}{pdfpagerange}[]{}
32 \define@key{LWR@hyperref}{pdfdoi}[]{}
33 \define@key{LWR@hyperref}{pdfurl}[]{}
34 \define@key{LWR@hyperref}{pdfidentifier}[]{}
35 \define@key{LWR@hyperref}{pdfsubtitle}[]{}
36 \define@key{LWR@hyperref}{pdfpubstatus}[]{}
37 \define@key{LWR@hyperref}{pdfcontactaddress}[]{}
38 \define@key{LWR@hyperref}{pdfcontactcity}[]{}
39 \define@key{LWR@hyperref}{pdfcontactregion}[]{}
40 \define@key{LWR@hyperref}{pdfcontactpostcode}[]{}
41 \define@key{LWR@hyperref}{pdfcontactcountry}[]{}
42 \define@key{LWR@hyperref}{pdfcontactphone}[]{}
43 \define@key{LWR@hyperref}{pdfcontactemail}[]{}
44 \define@key{LWR@hyperref}{pdfcontacturl}[]{}
45 \define@key{LWR@hyperref}{keeppdfinfo}[]{}
46 \define@key{LWR@hyperref}{pdfauthor}[]{}
47 \define@key{LWR@hyperref}{pdfkeywords}[]{}
```

File 230 **`lwarp-hyphenat.sty`**

§ 342   Package   # hyphenat

hyphenat (*Pkg*)   hyphenat is emulated during HTML output, while the print-mode version is used inside a lateximage.

for HTML output: `1 \LWR@ProvidesPackagePass{hyphenat}[2009/09/02]`

```
2 \LetLtxMacro\LWRHYNAT@origtextnhtt\textnhtt
3 \LetLtxMacro\LWRHYNAT@orignhttfamily\nhttfamily
4 \LetLtxMacro\LWRHYNAT@orignohyphens\nohyphens
5 \LetLtxMacro\LWRHYNAT@origbshyp\bshyp
6 \LetLtxMacro\LWRHYNAT@origfshyp\fshyp
7 \LetLtxMacro\LWRHYNAT@origdothyp\dothyp
8 \LetLtxMacro\LWRHYNAT@origcolonhyp\colonhyp
9 \LetLtxMacro\LWRHYNAT@orighyp\hyp
10
11 \LetLtxMacro\textnhtt\textttt
12 \LetLtxMacro\nhttfamily\ttfamily
13
14 \renewcommand{\nohyphens}[1]{#1}
15 \renewrobustcmd{\bshyp}{%
16    \ifmmode\backslash\else\textbackslash\fi%
17 }
18 \renewrobustcmd{\fshyp}{/}
19 \renewrobustcmd{\dothyp}{.}
20 \renewrobustcmd{\colonhyp}{:}
21 \renewrobustcmd{\hyp}{-}
22
23 \appto\LWR@restoreorigformatting{%
24 \LetLtxMacro\textnhtt\LWRHYNAT@origtextnhtt%
25 \LetLtxMacro\nhttfamily\LWRHYNAT@orignhttfamily%
26 \LetLtxMacro\nohyphens\LWRHYNAT@orignohyphens%
27 \LetLtxMacro\bshyp\LWRHYNAT@origbshyp%
28 \LetLtxMacro\fshyp\LWRHYNAT@origfshyp%
29 \LetLtxMacro\dothyp\LWRHYNAT@origdothyp%
30 \LetLtxMacro\colonhyp\LWRHYNAT@origcolonhyp%
31 \LetLtxMacro\hyp\LWRHYNAT@orighyp%
32 }
```

File 231   `lwarp-idxlayout.sty`

§ 343   Package   # idxlayout

(*Emulates or patches code by* Thomas Titz.)

idxlayout (*Pkg*)   idxlayout is emulated.

for HTML output: Discard all options for lwarp-idxlayout:

```
1 \LWR@ProvidesPackageDrop{idxlayout}[2012/03/30]
```

```
2 \newcommand{\LWR@indexprenote}{}
```

\AtBeginDocument to help with package load order.

```
3 \AtBeginDocument{
4    \preto\printindex{
5
6    \LWR@maybe@orignewpage
7    \LWR@startpars
8
9    \LWR@indexprenote
10
11   }
12 }
```

```
13 \newcommand{\setindexprenote}[1]{\renewcommand{\LWR@indexprenote}{#1}}
14 \newcommand*{\noindexprenote}{\renewcommand{\LWR@indexprenote}{}}
15
16 \newcommand{\idxlayout}[1]{}
17 \newcommand*{\indexfont}{}
18 \newcommand*{\indexjustific}{}
19 \newcommand*{\indexsubsdelim}{}
20 \newcommand*{\indexstheadcase}{}
```

File 232   **lwarp-ifoddpage.sty**

§ 344   Package   **ifoddpage**

(*Emulates or patches code by* MARTIN SCHARRER.)

ifoddpage (*Pkg*)   ifoddpage is emulated.

for HTML output: Discard all options for lwarp-ifoddpage:

```
1 \LWR@ProvidesPackageDrop{ifoddpage}[2016/04/23]
```

```
2 \newif\ifoddpage
3
4 \newif\ifoddpageoroneside
5
6 \DeclareRobustCommand{\checkoddpage}{\oddpagetrue\oddpageoronesidetrue}
7
8 \def\oddpage@page{1}
9
10 \def\@ifoddpage{%
11       \expandafter\@firstoftwo
12 }
13
14 \def\@ifoddpageoroneside{%
15       \expandafter\@firstoftwo
16 }
```

File 233   **lwarp-imakeidx.sty**

§ 345   Package   **imakeidx**

(*Emulates or patches code by* ENRICO GREGORIO.)

imakeidx (*Pkg*)   imakeidx is patched for use by lwarp.

letter headings   When using *makeindex*, to match the print and HTML output's display of index
letter headings, specify the lwarp.ist style:

> \makeindex[options={-s lwarp.ist}]

(For HTML the lwarp.ist style is used automatically, which displays letter headings.
When using *xindy* the default style also displays letter headings.)

index setup   See section 8.6.19 for how to setup *lwarpmk* to process the indexes with imakeidx,
both with and without shell escape.

1 `\LWR@ProvidesPackagePass{imakeidx}[2016/10/15]`

Use the new HTML suffix:

```
2 \catcode`\_=12%
3 \define@key{imki}{name}{\def\imki@name{#1_html}}
4 \catcode`\_=8%
```

`\printindex`   The HTML version of `\printindex`:

```
5 \catcode`\_=12%
6
7 \renewcommand*{\printindex}[1][\imki@jobname]{%
8 \LWR@maybe@orignewpage%
9 \LWR@startpars%
10 \ifstrequal{#1}{\imki@jobname}{%
11   \@ifundefined{#1@idxfile}{%
12         \imki@error{#1}%
13     }{%
14         \imki@putindex{#1}%
15     }%
16 }{%
17   \@ifundefined{#1_html@idxfile}{\imki@error{#1_html}}{\imki@putindex{#1_html}}%
18 }%
19 }
20
21 \catcode`\_=8%
```

`\@index`   The HTML version of `\@index`:

```
22 \VerifyCommand[lwarp][imakeidx]{\@index}{443B697F3326243540BE3FB7665606F6}
23
24 \catcode`\_=12%
25
26 \def\@index[#1]{%
27     \ifstrequal{#1}{\imki@jobname}%
28     {%
29         \@ifundefined{#1@idxfile}%
30         {%
31             \PackageWarning{lwarp-imakeidx}{Undefined index file `#1'}%
32             \begingroup
33             \@sanitize
34             \imki@nowrindex%
35         }%
36         {%
37             \edef\@idxfile{#1}%
38             \begingroup
39             \@sanitize
40             \@wrindex\@idxfile%
41         }%
42     }%
43     {%
44         \@ifundefined{#1_html@idxfile}%
45         {%
46             \PackageWarning{lwarp-imakeidx}{Undefined index file `#1_html'}%
47             \begingroup
48             \@sanitize
49             \imki@nowrindex%
50         }%
```

```
51          {%
52              \edef\@idxfile{#1_html}%
53              \begingroup
54              \@sanitize
55              \@wrindex\@idxfile%
56          }%
57      }%
58 }
59
60 \catcode`\_=8%
```

\item

\subitem

\subsubitem   HTML versions of \item, etc.:

```
61 \appto\theindex{%
62      \LetLtxMacro\item\LWR@indexitem%
63      \LetLtxMacro\subitem\LWR@indexsubitem%
64      \LetLtxMacro\subsubitem\LWR@indexsubsubitem%
65 }
```

\imki@wrindexentrysplit   {⟨file⟩} {⟨entry⟩} {⟨page⟩}

\imki@wrindexentryunique   {⟨file⟩} {⟨entry⟩} {⟨page⟩}

While writing index entries, adds an HTML label, and writes the label's index instead of the page number:

```
66 \VerifyCommand[lwarp][imakeidx]{\imki@wrindexentrysplit}{D8ABE70A4355F52E36723AFAB74F71E7}
67
68 \renewcommand\imki@wrindexentrysplit[3]{%
69      \addtocounter{LWR@autoindex}{1}%
70      \expandafter\protected@write\csname#1@idxfile\endcsname{}%
71          {\string\indexentry{#2}{\arabic{LWR@autoindex}}}%
```

The label is assigned after the file write to avoid conflict with cleveref.

```
72      \label{LWRindex-\arabic{LWR@autoindex}}%
73 }
74
75 \VerifyCommand[lwarp][imakeidx]{\imki@wrindexentryunique}{9131E144394D273F316D03FA91BA0E2B}
76
77 \renewcommand\imki@wrindexentryunique[3]{%
78      \addtocounter{LWR@autoindex}{1}%
79      \protected@write\@indexfile{}%
80          {\string\indexentry[#1]{#2}{\arabic{LWR@autoindex}}}%
```

The label is assigned after the file write to avoid conflict with cleveref.

```
81      \label{LWRindex-\arabic{LWR@autoindex}}%
82 }
```

\LWR@imki@setxdydefopts

Sets the *xindy* HTML options, ignoring the user's settings.

```
83 \newcommand*{\LWR@imki@setxdydefopts}{%
84      \edef\imki@options{ \space %
85          -M \space \LWR@xindyStyle\space %
86          -L \space \LWR@xindyLanguage\space %
87          -C \space \LWR@xindyCodepage\space %
88      }%
```

89 }

\LWR@imki@setdefopts   {⟨*user options*⟩}

Sets the HTML options, added to the user's settings, depending on whether *makeindex* or *xindy* are used.

For *makeindex*, the user's choice is ignored, and only the lwarp version is used. (Only one style at a time is possible.)

For *xindy*, multiple modules may be specified, and the lwarp version is appended.

```
90 \newcommand*{\LWR@imki@setdefopts}[1]{%
91 \ifblank{#1}{%
92     \edef\imki@options{\space -s \space \LWR@makeindexStyle \space}%
93     \ifdefstring{\imki@progdefault}{xindy}{\LWR@imki@setxdydefopts}{}%
94     \ifdefstring{\imki@progdefault}{texindy}{\LWR@imki@setxdydefopts}{}%
95     \ifdefstring{\imki@progdefault}{truexindy}{\LWR@imki@setxdydefopts}{}%
96 }{%
97     \edef\imki@options{\space #1 \space}%
98 }%
99 }
```

\imki@makeindex   Use the new HTML options:

```
100 \VerifyCommand[lwarp][imakeidx]{\imki@makeindex}{83AEF6DF7A13F7D0565457DFB83D42B5}
101
102 \xpatchcmd{\imki@makeindex}
103     {\let\imki@options\space}
104     {\LWR@imki@setdefopts{}}%
105     {}
106     {\LWR@patcherror{imakeidx}{makeindex}}
```

Use the new HTML options.

```
107 \define@key{imki}{options}{\LWR@imki@setdefopts{#1}}
```

\imki@resetdefaults   Use the new HTML options:

```
108 \VerifyCommand[lwarp][imakeidx]{\imki@resetdefaults}{3D417615569AC35F199A3FBD03B640E3}
109
110 \xpatchcmd{\imki@resetdefaults}
111     {\def\imki@options{ }}
112     {\LWR@imki@setdefopts{}}
113     {}
114     {\LWR@patcherror{imakeidx}{resetdefaults}}
```

theindex was already defined \AtBeginDocument by the lwarp core, so it must be redefined here similarly, but patched for imakeidx:

Env   theindex

```
115 \AtBeginDocument{
116 \renewenvironment*{theindex}{%
117     \imki@maybeaddtotoc
118      \imki@indexlevel{\indexname}
119     \LetLtxMacro\item\LWR@indexitem%
120     \LetLtxMacro\subitem\LWR@indexsubitem%
121     \LetLtxMacro\subsubitem\LWR@indexsubsubitem%
122 }{}
123 }% AtBeginDocument
```

Update to the new defaults:

124 \imki@resetdefaults

Update to the new patches:

\AtBeginDocument is because \@wrindex is previously defined as \AtBeginDocument in the lwarp core.

```
125 \ifimki@splitindex
126   \let\imki@startidx\imki@startidxunique
127   \AtBeginDocument{\let\@wrindex\imki@wrindexunique}
128   \let\imki@putindex\imki@putindexunique
129   \let\imki@wrindexentry\imki@wrindexentryunique
130   \let\imki@startidxsplit\@undefined
131   \let\imki@wrindexsplit\@undefined
132   \let\imki@putindexsplit\@undefined
133 \else
134   \let\imki@startidx\imki@startidxsplit
135   \AtBeginDocument{\let\@wrindex\imki@wrindexsplit}
136   \let\imki@putindex\imki@putindexsplit
137   \let\imki@wrindexentry\imki@wrindexentrysplit
138   \let\imki@startidxunique\@undefined
139   \let\imki@wrindexunique\@undefined
140   \let\imki@putindexunique\@undefined
141 \fi
```

File 234   **`lwarp-impnattypo.sty`**

§ 346   Package   # impnattypo

impnattypo (*Pkg*)   impnattypo is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{impnattypo}[2019/03/04]

File 235   **`lwarp-index.sty`**

§ 347   Package   # index

(*Emulates or patches code by* DAVID M. JONES.)

index (*Pkg*)   index is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{index}[2004/01/20]

Use \theLWR@autoindex instead of \thepage. \@tempswatrue is used to force an immediate write to the index file instead of waiting until the end of the page.

```
2 \VerifyCommand[lwarp][index]{\newindex}{F714216FA78BCC8DB70B7BB92BE05F3C}
3
4 \xpatchcmd{\newindex}
5   {\x@newindex[thepage]}
6   {%
```

```
7          \@tempswatrue%
8          \x@newindex[theLWR@autoindex]%
9      }
10     {}
11     {\LWR@patcherror{index}{newindex}}
12
13 \VerifyCommand[lwarp][index]{\renewindex}{B81B08BFE7A2F5CA7D84D4A5A40E7A44}
14
15 \xpatchcmd{\renewindex}
16     {\x@renewindex[thepage]}
17     {%
18         \@tempswatrue%
19         \x@renewindex[theLWR@autoindex]%
20     }
21     {}
22     {\LWR@patcherror{index}{renewindex}}
```

Patched to set a new autoindex:

```
23 \VerifyCommand[lwarp][index]{\@wrindex}{C58C10ACFC42D711D0DA8F4759BA951D}
24
25 \xpatchcmd{\@wrindex}
26     {\begingroup}
27     {%
28         \addtocounter{LWR@autoindex}{1}%                          lwarp
29         \label{LWRindex-\arabic{LWR@autoindex}}%      lwarp
30         \begingroup%
31     }
32     {}
33     {\LWR@patcherror{index}{@wrindex}}
```

\AtBeginDocument lwarp core \lets \@wrindex to \LWR@wrindex. Since the index package has been loaded, \let to its version instead:

```
34 \let\LWR@index@wrindex\@wrindex
35
36 \AtBeginDocument{
37 \let\@wrindex\LWR@index@wrindex
38 }
```

Modified to add \index@prologue:

```
39 \AtBeginDocument{
40 \renewenvironment*{theindex}{%
41     \LWR@indexsection{\indexname}%
42     \ifx\index@prologue\@empty\else
43         \index@prologue
44         \bigskip
45     \fi
46     \LetLtxMacro\item\LWR@indexitem%
47     \LetLtxMacro\subitem\LWR@indexsubitem%
48     \LetLtxMacro\subsubitem\LWR@indexsubsubitem%
49 }{}
50 }% AtBeginDocument
```

Disabled:

```
51 \def\@showidx#1{}
52 \let\@texttop\relax
```

```
53 \renewcommand*{\raggedbottom}{}
54 \renewcommand*{\flushbottom}{}
55 \renewcommand*{\markboth}[2]{}
56 \renewcommand*{\markright}[1]{}
```

File 236   **lwarp-inputtrc.sty**

§ 348   Package   # inputtrc

(*Emulates or patches code by* UWE LÜCK.)

inputtrc (*Pkg*)   inputtrc is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{inputtrc}[2012/10/10]

Patched to remove extraneous spaces, which sometimes showed up in logos inside a lateximage.

```
2 \VerifyCommand[lwarp][inputtrc]{\IT@prim@input}{03F74081468CFB6308896BDEB61D1E23}
3
4 \renewcommand*{\IT@prim@input}[1]{%
5   \typeout{\IT@indent\IT@currfile INPUTTING #1}%
6 %% ... TODO: option to write to `.log' only.
7   \xdef\IT@filestack{{\IT@currfile}\IT@filestack}%
8   \xdef\IT@currfile{#1}%
9   \expandafter \gdef\expandafter \IT@indent\expandafter{%
10    \IT@indent \IT@indent@unit}%                lwarp
11  \@@input#1%                                  lwarp
12  \expandafter\IT@pop@indent\IT@indent   \@nil% lwarp
13  \expandafter\IT@pop@file  \IT@filestack\@nil% lwarp
14  \IT@maybe@returnmessage%% v0.2              lwarp
15 }
```

File 237   **lwarp-intopdf.sty**

§ 349   Package   # intopdf

intopdf (*Pkg*)   intopdf is emulated.

The filespec, MIME type, and description are ignored for now.

for HTML output: 1 \LWR@ProvidesPackageDrop{intopdf}[2019/05/28]

```
2 \NewDocumentCommand{\attachandlink}{o m o m m}{%
3   \LWR@href{#2}{#5}%
4 }
```

File 238   **lwarp-isomath.sty**

§ 350   Package   # isomath

(*Emulates or patches code by* GÜNTER MILDE.)

isomath (*Pkg*)   isomath is used as-is for SVG math, and emulated for MATHJAX.

⚠ MATHJAX sans   MATHJAX does not provide a sans math font, so sans is typeset as roman.

for HTML output: 1 \LWR@ProvidesPackagePass{isomath}[2012/09/04]

```
2 \begin{warpMathJax}
3 \CustomizeMathJax{\let\mathbfit\boldsymbol}
4 \CustomizeMathJax{\let\mathsfbfit\mathbfit}% not sans
5 \CustomizeMathJax{\let\mathsfit\mathit}% not sans
6 \CustomizeMathJax{\let\vectorsym\mathbfit}
7 \CustomizeMathJax{\let\matrixsym\mathbfit}
8 \CustomizeMathJax{\let\tensorsym\mathsfbfit}
9 \CustomizeMathJax{\let\mathboldsans\mathsfbfit}
10 \CustomizeMathJax{\let\mathbold\mathbfit}
11 \CustomizeMathJax{\let\mathsans\mathrm}% not sans
12 \end{warpMathJax}
```

File 239   **`lwarp-isotope.sty`**

§ 351   Package   # isotope

(*Emulates or patches code by* HEIKO BAUKE.)

isotope (*Pkg*)   isotope is patched for use by lwarp with SVG math, and emulated for MATHJAX.

for HTML output: 1 \LWR@ProvidesPackagePass{isotope}[2011/08/26]

```
2 \newcommand{\LWR@HTML@isotope@two}[2][]{%
3     \renewcommand{\isotope@atomicnumber}{#1}%
4     \edef\LWR@isotope@alttag{%
5         \textbackslash(
6         \textbackslash{}isotope
7         [\isotope@nucleonnumber]%
8         [\isotope@atomicnumber]%
9         \{#2\}
10         \textbackslash)%
11     }%
12 \ifbool{mathjax}%
13     {\LWR@isotope@alttag}%
14     {% SVG
15         \m@th%
16         \LWR@subsingledollar*%
17         {% alt tag
18             \LWR@isotope@alttag%
19         }%
20         {isotope}% add'l hashing
21         {% contents
22             \settowidth\@tempdimb{%
23                 \ensuremath{\scriptstyle\isotope@nucleonnumber}%
24             }%
25             \settowidth\@tempdimc{%
26                 \ensuremath{\scriptstyle\isotope@atomicnumber}%
27             }%
28             \ifdim\@tempdimb<\@tempdimc\@tempdimb=\@tempdimc\fi%
29             \ensuremath{
30                 {}%
```

```
31                    ^{\makebox[\@tempdimb][r]{%
32                        \ensuremath{%
33                            \scriptstyle\isotope@nucleonnumber%
34                        }% ensuremath
35                    }}%
36                    _{\makebox[\@tempdimb][r]{%
37                        \ensuremath{%
38                            \scriptstyle\isotope@atomicnumber%
39                        }% ensuremath
40                    }}%
41                    \isotopestyle{#2}%
42                }% ensuremath
43            }% contents
44        }% SVG
45    \endgroup%
46 }%
47 \LWR@formatted{isotope@two}
48
49 \begin{warpMathJax}
50 \CustomizeMathJax{%
51     \newcommand{\LWRisotopetwo}[2][]{%
52        {%
53            \vphantom{\mathrm{#2}}%
54            {}^{\LWRisotopenucleonnumber}_{#1}%
55            \mathrm{#2}%
56        }%
57    }%
58 }
59
60 \CustomizeMathJax{%
61     \newcommand{\isotope}[1][]{%
62        \def\LWRisotopenucleonnumber{#1}%
63        \LWRisotopetwo%
64    }%
65 }
66 \end{warpMathJax}
```

File 240   `lwarp-jurabib.sty`

§ 352   Package   **jurabib**

(*Emulates or patches code by* JENS BERGER.)

jurabib (*Pkg*)   jurabib is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{jurabib}[2004/01/25]

```
2 \renewrobustcmd{\jblangle}{\textless}
3
4 \renewrobustcmd{\jbrangle}{\textgreater}
5
6 \VerifyCommand[lwarp][jurabib]{\jb@biblaw@item}{F93545B67E684787264DA900F185A25A}
7
8 \renewcommand*{\jb@biblaw@item}{%
9     \hspace{0.5em}%
10 %    $\triangleright$
11    \HTMLunicode{25B7}%      lwarp%
```

```
12      \hspace{0.5em}%
13 }
14
15 \VerifyCommand[lwarp][jurabib]{\jbarchsig}{8D821FA370CBD0A61325D5A278E0A369}
16
17 \renewrobustcmd{\jbarchsig}[2]{%
18      \ifjbweareinbib
19          \settowidth{\jb@subarchitemwidth}{\jbsamesubarchindent+#1}%
20      \setlength{\jb@subarchentrywidth}{\textwidth-\jb@subarchitemwidth-4em}%
21 %      \begin{tabular}{@{}p{\jb@subarchitemwidth}@{}j{\jb@subarchentrywidth}@{}}%
22          #1\ifjb@dot\unskip\unskip\unskip.\fi
23 %             &
24           \quad%          lwarp
25           \ifthenelse{\equal{#2}{}}{}{\jbarchnameformat{#2}}%
26 %        \end{tabular}
27      \fi
28 }%
29
30
31 \VerifyCommand[lwarp][jurabib]{\jb@do@post@item}{4FD79AF40E8460C52306C33CF825B63F}
32
33 \xpatchcmd{\jb@do@post@item}
34      {\begin{tabular}{p{\jb@biblaw@item@width}j{\jb@biblaw@entry@width}}}
35      {}
36      {}
37      {\LWR@patcherror{jurabib}{jb@do@post@item 1}}
38
39 \xpatchcmd{\jb@do@post@item}
40      {\multicolumn{2}{p{\columnwidth}}{\jb@@name}}
41      {\jb@@name}
42      {}
43      {\LWR@patcherror{jurabib}{jb@do@post@item 2}}
44
45 \xpatchcmd{\jb@do@post@item}
46      {\jb@biblaw@item & \jb@@fulltitle}
47      {\jb@biblaw@item \quad \jb@@fulltitle}
48      {}
49      {\LWR@patcherror{jurabib}{jb@do@post@item 3}}
50
51 \xpatchcmd{\jb@do@post@item}
52      {\end{tabular}}
53      {}
54      {}
55      {\LWR@patcherror{jurabib}{jb@do@post@item 4}}
56
57 \xpatchcmd{\jb@do@post@item}
58      {\begin{minipage}[t]{\bibnumberwidth}}
59      {}
60      {}
61      {\LWR@patcherror{jurabib}{jb@do@post@item 5}}
62
63 \xpatchcmd{\jb@do@post@item}
64      {\end{minipage}}
65      {\quad}
66      {}
67      {\LWR@patcherror{jurabib}{jb@do@post@item 6}}
```

File 241   **`lwarp-karnaugh-map.sty`**

§ 353   Package   # karnaugh-map

(*Emulates or patches code by* Mattias Jacobsson.)

karnaugh-map (*Pkg*)   karnaugh-map is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{karnaugh-map}[2017/02/20]

This patch is needed only because lwarp changes the definition of \&, and the original uses \ifnum to compare 0 with \&. It is hard to patch this environment, so the entire thing is redefined here, with the lwarp modifications identified in comments.

```
2 \VerifyEnvironment[lwarp][karnaugh-map]{karnaugh-map}
3    {FFA0270032620E79C8344E63AEDBF925}{08A76B622DBB34F033284513743C5F8C}
4
5 \RenewDocumentEnvironment{karnaugh-map}{s O{4} O{4} O{1} O{$X_1X_0$} O{$X_3X_2$} O{$X_5X_4$}} {%
6    \begingroup
7      % store map size {[START]
8        \renewcommand{\@karnaughmap@var@mapsizex@}{#2}%
9        \renewcommand{\@karnaughmap@var@mapsizey@}{#3}%
10       \renewcommand{\@karnaughmap@var@mapsizez@}{#4}%
11     % [END]}
12     % determinate if markings should be color or black and white
13     \IfBooleanTF{#1}{%
14       % should be black and white
15       \renewcommand{\@karnaughmap@var@bw@}{1}%
16     }{%
17       % should be color
18       \renewcommand{\@karnaughmap@var@bw@}{0}%
19     }%
20     %
21     % find matching matrix template and alignment parameters {[START]
22     \newcommand{\@karnaughmap@local@matrixtemplate@}{0}% '0' is considered as missing matrix template
23       \newcommand{\@karnaughmap@local@maprealignmentx@}{0}%
24       \newcommand{\@karnaughmap@local@maprealignmenty@}{0}%
25     \ifnum\@karnaughmap@var@mapsizex@\@karnaughmap@var@mapsizey@\@karnaughmap@var@mapsizez@=221
26       \renewcommand{\@karnaughmap@local@matrixtemplate@}{%
27              \&                  0 \&                 1 \& \phantom{0} \\
28            0 \& |(000000)| \phantom{0} \& |(000001)| \phantom{0} \&            \\
29            1 \& |(000010)| \phantom{0} \& |(000011)| \phantom{0} \&            \\
30        \phantom{0} \&                 \&                 \&            \\
31       }%
32     \fi
33     \ifnum\@karnaughmap@var@mapsizex@\@karnaughmap@var@mapsizey@\@karnaughmap@var@mapsizez@=241
34       \renewcommand{\@karnaughmap@local@matrixtemplate@}{%
35              \&                  0 \&                 1 \& \phantom{00} \\
36          00 \& |(000000)| \phantom{0} \& |(000001)| \phantom{0} \&            \\
37          01 \& |(000010)| \phantom{0} \& |(000011)| \phantom{0} \&            \\
38          11 \& |(000110)| \phantom{0} \& |(000111)| \phantom{0} \&            \\
39          10 \& |(000100)| \phantom{0} \& |(000101)| \phantom{0} \&            \\
40        \phantom{00} \&                 \&                 \&            \\
41       }%
```

```
42      \fi
43    \ifnum\@karnaughmap@var@mapsizex@\@karnaughmap@var@mapsizey@\@karnaughmap@var@mapsizez@=421
44      \renewcommand{\@karnaughmap@local@matrixtemplate@}{%
45              \& 				00 \& 			01 \& 			11 \& 			10 \& \phar
46            0 \& |(000000)| \phantom{0} \& |(000001)| \phantom{0} \& |(000011)| \phantom{0} \& |(000
47            1 \& |(000100)| \phantom{0} \& |(000101)| \phantom{0} \& |(000111)| \phantom{0} \& |(000
48      \phantom{00} \& 				\& 			\& 			\& 			\&
49      }%
50      \fi
51    \ifnum\@karnaughmap@var@mapsizex@\@karnaughmap@var@mapsizey@\@karnaughmap@var@mapsizez@=441
52      \renewcommand{\@karnaughmap@local@matrixtemplate@}{%
53              \& 				00 \& 			01 \& 			11 \& 			10 \& \phar
54           00 \& |(000000)| \phantom{0} \& |(000001)| \phantom{0} \& |(000011)| \phantom{0} \& |(00
55           01 \& |(000100)| \phantom{0} \& |(000101)| \phantom{0} \& |(000111)| \phantom{0} \& |(00
56           11 \& |(001100)| \phantom{0} \& |(001101)| \phantom{0} \& |(001111)| \phantom{0} \& |(00
57           10 \& |(001000)| \phantom{0} \& |(001001)| \phantom{0} \& |(001011)| \phantom{0} \& |(00
58      \phantom{00} \& 				\& 			\& 			\& 			\&
59      }%
60      \fi
61    \ifnum\@karnaughmap@var@mapsizex@\@karnaughmap@var@mapsizey@\@karnaughmap@var@mapsizez@=442
62      \renewcommand{\@karnaughmap@local@matrixtemplate@}{%
63              \& 				00 \& 			01 \& 			11 \& 			10 \& \phar
64           00 \& |(000000)| \phantom{0} \& |(000001)| \phantom{0} \& |(000011)| \phantom{0} \& |(00
65           01 \& |(000100)| \phantom{0} \& |(000101)| \phantom{0} \& |(000111)| \phantom{0} \& |(00
66           11 \& |(001100)| \phantom{0} \& |(001101)| \phantom{0} \& |(001111)| \phantom{0} \& |(00
67           10 \& |(001000)| \phantom{0} \& |(001001)| \phantom{0} \& |(001011)| \phantom{0} \& |(00
68      \phantom{00} \& 				\& 			\& 			\& 			\&
69      }%
70      \renewcommand{\@karnaughmap@local@maprealignmentx@}{2.5}%
71      \fi
72    \ifnum\@karnaughmap@var@mapsizex@\@karnaughmap@var@mapsizey@\@karnaughmap@var@mapsizez@=444
73      \renewcommand{\@karnaughmap@local@matrixtemplate@}{%
74              \& 				00 \& 			01 \& 			11 \& 			10 \& \phar
75           00 \& |(000000)| \phantom{0} \& |(000001)| \phantom{0} \& |(000011)| \phantom{0} \& |(00
76           01 \& |(000100)| \phantom{0} \& |(000101)| \phantom{0} \& |(000111)| \phantom{0} \& |(00
77           11 \& |(001100)| \phantom{0} \& |(001101)| \phantom{0} \& |(001111)| \phantom{0} \& |(00
78           10 \& |(001000)| \phantom{0} \& |(001001)| \phantom{0} \& |(001011)| \phantom{0} \& |(00
79      \phantom{00} \& 				\& 			\& 			\& 			\&
80           00 \& |(100000)| \phantom{0} \& |(100001)| \phantom{0} \& |(100011)| \phantom{0} \& |(10
81           01 \& |(100100)| \phantom{0} \& |(100101)| \phantom{0} \& |(100111)| \phantom{0} \& |(10
82           11 \& |(101100)| \phantom{0} \& |(101101)| \phantom{0} \& |(101111)| \phantom{0} \& |(10
83           10 \& |(101000)| \phantom{0} \& |(101001)| \phantom{0} \& |(101011)| \phantom{0} \& |(10
84      \phantom{00} \& 				\& 			\& 			\& 			\&
85      }%
86      \renewcommand{\@karnaughmap@local@maprealignmentx@}{2.5}%
87      \renewcommand{\@karnaughmap@local@maprealignmenty@}{-2.5}%
88      \fi
89    % [END]}
90    % test if a matrix template is found or not(aka "\@karnaughmap@local@matrixtemplate@" equals to '0')
91    \ifdefstring{\@karnaughmap@local@matrixtemplate@}{0}{%  lwarp
92 %    \ifnum0=\@karnaughmap@local@matrixtemplate@%   original
93      % print error if no template could be found
94      \PackageError{lwarp-karnaugh-map}{%
95        Can not find a template fitting your specification
96      (\@karnaughmap@var@mapsizex@\space x \@karnaughmap@var@mapsizey@\space x
97        \@karnaughmap@var@mapsizez@)%
98      }{%
99        Existing templates have the following dimensions:
100       2x2x1, 2x4x1, 4x2x1, 4x4x1, 4x4x2, and 4x4x4.
101     }%
```

```
102 %      \fi   original
103   }{\relax}%    lwarp
104   \begin{tikzpicture}
105     % grid
106     % for all dimensions
107   \draw[color=black, ultra thin] (0,0) grid (\@karnaughmap@var@mapsizex@,\@karnaughmap@var@mapsizey
108     % when there are 2 sub maps
109     \ifnum\@karnaughmap@var@mapsizez@=2
110       \draw[color=black, ultra thin] (5,0) grid (9,4);
111     \fi
112     % when there are 4 sub maps
113     \ifnum\@karnaughmap@var@mapsizez@=4
114       \draw[color=black, ultra thin] (5,0) grid (9,4);
115       \draw[color=black, ultra thin] (0,-5) grid (4,-1);
116       \draw[color=black, ultra thin] (5,-5) grid (9,-1);
117     \fi
118     % labels
119     % for all dimensions
120   \node[above] at (\@karnaughmap@var@mapsizex@*0.5,\@karnaughmap@var@mapsizey@+0.9) {\small{#5}};
121     \node[left] at (-0.9,\@karnaughmap@var@mapsizey@*0.5) {\small{#6}};
122     % when there are 2 sub maps
123     \ifnum\@karnaughmap@var@mapsizez@=2
124       \node[above] at (7,4.9) {\small{#5}};
125       % extra sub maps labels
126       \node[below] at (2,-0.1) {\small{#7$=0$}};
127       \node[below] at (7,-0.1) {\small{#7$=1$}};
128     \fi
129     % when there are 4 sub maps
130     \ifnum\@karnaughmap@var@mapsizez@=4
131       \node[above] at (7,4.9) {\small{#5}};
132       \node[left] at (-0.9,-3) {\small{#6}};
133       % extra sub maps labels
134       \node[below] at (2,-0.1) {\small{#7$=00$}};
135       \node[below] at (7,-0.1) {\small{#7$=01$}};
136       \node[below] at (2,-5.1) {\small{#7$=10$}};
137       \node[below] at (7,-5.1) {\small{#7$=11$}};
138     \fi
139     % data
140     \matrix[
141       matrix of nodes,
142       ampersand replacement=\&,
143       column sep={1cm,between origins},
144       row sep={1cm,between origins},
145     ] at (\@karnaughmap@var@mapsizex@*0.5+\@karnaughmap@local@maprealignmentx@,\@karnaughmap@var@map
146       \@karnaughmap@local@matrixtemplate@%
147     };
148 }{
149   \end{tikzpicture}
150   \endgroup
151 }
```

File 242   **`lwarp-keyfloat.sty`**

§ 354   Package   **keyfloat**

(*Emulates or patches code by* BRIAN DUNN.)

keyfloat (*Pkg*)   keyfloat is supported with a considerable amount of hacking. (It's a mashup of

lwarp, keyfloat, and tocdata.)

⚠ keywrap   If placing a \keyfig[H] inside a keywrap, use an absolute width for \keyfig, in-
            stead of lw-proportional widths. (The [H] option forces the use of a minipage,
            which internally adjusts for a virtual 6-inch wide minipage, which then corrupts
            the lw option.)

For wrapped figures, overhang and number of lines are ignored.

for HTML output:
```
1 \LWR@ProvidesPackagePass{keyfloat}[2019/09/23]
2
3 \IfPackageAtLeastTF{keyfloat}{2019/09/23}{\relax}{
4     \PackageError{lwarp-keyfloat}
5     {%
6         The keyfloat package is out of date.\MessageBreak
7         Update to keyfloat v2.01 2019/09/23 or later%
8     }
9     {%
10        Please update the keyfloat package.  It's worth it!%
11    }
12 }
```

After keyfloat has loaded:

```
13 \AtBeginDocument{
```

\KFLT@LWR@hook@boxouter   Integration for keyfloat.
(*Hook*) [keyfloat]

```
14 \providecommand*{\KFLT@LWR@hook@boxouter}{}
15
16 \renewcommand*{\KFLT@LWR@hook@boxouter}{%
17     \ifbool{KFLT@keywrap}{%
18     }{%
19         \ifnumequal{\value{KFLT@keyfloatdepth}}{0}{%
20             \setlength{\linewidth}{6in}%
21             \setlength{\textwidth}{6in}%
22             \setlength{\textheight}{9in}%
23         }{}%
24     }%
25     \normalcolor%
26 }
```

\KFLT@LWR@hook@keysubfloats   Integration for keyfloat.
(*Hook*) [keyfloat]

```
27 \LetLtxMacro\KFLT@LWR@hook@keysubfloats\KFLT@LWR@hook@boxouter
```

\KFLT@LWR@hook@keyfloatsminipage   Integration for keyfloat.
(*Hook*) [keyfloat]

```
28 \let\KFLT@LWR@hook@keyfloatsminipage\relax
29 \let\endKFLT@LWR@hook@keyfloatsminipage\relax
30 \newenvironment*{KFLT@LWR@hook@keyfloatsminipage}[1]{}{}
```

\KFLT@LWR@hook@keyfloats   Integration for keyfloat.
(*Hook*) [keyfloat]

```
31 \LetLtxMacro\KFLT@LWR@hook@keyfloats\KFLT@LWR@hook@boxouter
32
33 \VerifyCommand[lwarp][keyfloat]{\KFLT@maybeendfloatrow}{ABD652AC104E3CF79D66B92BC7E4E2D7}
34
```

```
35 \renewcommand*{\KFLT@maybeendfloatrow}{%
36    \ifnumless{\value{KFLT@thiscol}}{\value{KFLT@numcols}}%
37        {}% thiscol < numcols
38        {% >=
39            \defcounter{KFLT@thiscol}{0}%
40        }%
41 }%
42
43 \VerifyCommand[lwarp][keyfloat]{\KFLT@trackrows}{17F751691BBEDD3459F494B072DC2F11}
44
45 \renewcommand{\KFLT@trackrows}%
46 {%
```

If are nested inside a keyfloats or a subfloat:

```
47    \ifboolexpr{%
48        test {\ifnumgreater{\value{KFLT@keyfloatdepth}}{0}} or%
49        bool{KFLT@inkeysubfloats}%
50    }%
51    {% nested
```

Tracks row start and end:

```
52        \KFLT@maybestartfloatrow%
```

Possibly fill space between columns:

```
53        \ifnumgreater{\value{KFLT@thiscol}}{1}%
54            {%
55 %                \hfill%
56            }%
57            {}%
58    }% nested
59    {}% not nested
60 }
```

```
61 \VerifyCommand[lwarp][keyfloat]{\KFLT@onefigureimage}{803E82896F8D49700946B1A6CB132A55}
62
63 \RenewDocumentCommand{\KFLT@onefigureimage}{m}
64 {%
65 \LWR@traceinfo{KFLT@onefigureimage}%
66 % \begin{lrbox}{\KFLT@envbox}%
67 \ifthenelse{\NOT\equal{\KFLT@lw}{}}%
68    {%

69        \ifdimgreater{\KFLT@h}{0pt}%
70        {%
71            \KFLT@frame{%
72                \includegraphics%
73                [%
74                    scale=\KFLT@s,%
75                    width=\KFLT@imagewidth,%
76                    height=\KFLT@h,%
77                    \KFLT@keepaspectratio,%
78                ]{#1}%
79            }%
80        }%
81        {%
```

```
 82              \KFLT@frame{\includegraphics%
 83                  [scale=\KFLT@s,width=\KFLT@imagewidth]{#1}}%
 84          }%
 85      }%
 86      {% not linewidth
 87          \ifthenelse{\dimtest{\KFLT@w}{>}{0pt}}%
 88          {% width is given
 89              \ifthenelse{\dimtest{\KFLT@h}{>}{0pt}}%
 90              {% w and h
 91                  \KFLT@frame{\includegraphics[%
 92                      scale=\KFLT@s,%
 93                      width=\KFLT@imagewidth,%
 94                      height=\KFLT@h,%
 95                      \KFLT@keepaspectratio,%
 96                  ]{#1}}%
 97              }% w and h
 98              {% only w
 99                  \KFLT@frame{\includegraphics%
100                      [scale=\KFLT@s,width=\KFLT@imagewidth]{#1}}%
101              }% only w
102          }% width is given
103          {% width is not given
104              \ifthenelse{\dimtest{\KFLT@h}{>}{0pt}}%
105              {%
106                  \KFLT@frame{\includegraphics%
107                  [scale=\KFLT@s,height=\KFLT@h]{#1}}%
108              }%
109              {%
110                  \KFLT@frame{\includegraphics%
111                  [scale=\KFLT@s]{#1}}%
112              }%
113          }% width is not given
114      }% not linewidth
115 % \end{lrbox}%
116 % \unskip%
117 % \KFLT@findenvboxwidth%
118 % \begin{turn}{\KFLT@r}%
119 % \KFLT@frame{\usebox{\KFLT@envbox}}%
120 % \unskip%
121 % \end{turn}%
122 \LWR@traceinfo{KFLT@onefigureimage: done}%
123 }


124 \VerifyEnvironment[lwarp][keyfloat]{KFLT@boxinner}
125    {44BA9E3F4EA1B3E533F47377BA47F145}{590DE3AADA8DF85EF6E1589B41F0D4F6}
126
127 \RenewDocumentEnvironment{KFLT@boxinner}{}
128 {%
129     \LWR@traceinfo{KFLT@boxinner}%
130     \LWR@stoppars%
131     \minipagefullwidth%
132     \ifboolexpr{bool{KFLT@ft} or bool{KFLT@f}}{%
133         \fminipage{\KFLT@imagewidth}%
134     }{%
135         \minipage{\KFLT@imagewidth}%
136     }%
137 }
138 {%
139     \ifboolexpr{bool{KFLT@ft} or bool{KFLT@f}}{%
140         \endfminipage%
```

```
141     }{%
142         \endminipage%
143     }%
144     \LWR@startpars%
145     \LWR@traceinfo{KFLT@boxinner: done}%
146 }
```

```
147 \newcommand*{\LWR@KFLT@settextalign}[1]{%
148     \def\LWR@KFLT@textalign{justify}%
149     \ifcsstring{KFLT@#1textalign}{\centering}%
150         {\def\LWR@KFLT@textalign{center}}%
151         {}%
152     \ifcsstring{KFLT@#1textalign}{\raggedleft}%
153         {\def\LWR@KFLT@textalign{right}}%
154         {}%
155     \ifcsstring{KFLT@#1textalign}{\raggedright}%
156         {\def\LWR@KFLT@textalign{left}}%
157         {}%
158 }
159
160 \VerifyCommand[lwarp][keyfloat]{\KFLT@addtext}{C086CC818525A9B03EDEACC02609A3BE}
161
162 \renewcommand{\KFLT@addtext}[1]
163 {%
```

Is there text to add?

```
164     \ifcsempty{KFLT@#1t}%
165     {}% no text
166     {% text to add
167         {% local
```

Add some space, then create a <div> to contain the text:

```
168         \addvspace{\smallskipamount}%
169         \LWR@KFLT@settextalign{#1}%
170         \begin{BlockClass}[text-align:\LWR@KFLT@textalign]{floatnotes}%
```

Set the alignment and some text parameters:

```
171 %         \csuse{KFLT@#1textalign}%
172 %         \footnotesize%
173         \setlength{\parskip}{1.5ex}%
174         \setlength{\parindent}{0em}%
```

Typeset the actual text:

```
175         \csuse{KFLT@#1t}%
```

Close it all out with a little more space:

```
176         \end{BlockClass}%
177 %         \par\addvspace{2ex}%
178         }% local
179     }% text to add
180 }
181
182 \IfPackageLoadedTF{tocdata}
```

```
183 {}
184 {% tocdata not loaded
185
186     \newcommand*{\LWR@KFLT@setnamealign}[1]{%
187         \def\LWR@KFLT@textalign{justify}%
188         \ifstrequal{#1}{\centering}%
189             {\def\LWR@KFLT@textalign{center}}%
190             {}%
191         \ifstrequal{#1}{\raggedleft}%
192             {\def\LWR@KFLT@textalign{right}}%
193             {}%
194         \ifstrequal{#1}{\raggedright}%
195             {\def\LWR@KFLT@textalign{left}}%
196             {}%
197     }
198
199     \VerifyCommand[lwarp][keyfloat]{\KFLT@@addartisttext}{35968ED08D9BE09FF1B45E1E40AFE9A7}
200
201     \renewcommand*{\KFLT@@addartisttext}[3]{%
```

Add space and create the name inside a <div>:

```
202 %         \addvspace{\medskipamount}%
203 %     \begin{minipage}{\linewidth}%
204         \LWR@KFLT@setnamealign{#3}%
205         \begin{BlockClass}[text-align:\LWR@KFLT@textalign]{floatnotes}%
```

Text alignment is #3, and depends on artist or author:

```
206 %       #3%
```

#1 is empty or 'subgrp'
#2 is empty for artist, 'u' for author:

```
207         \footnotesize\textsc{%
208             \KFLT@optionalname{\csuse{KFLT@#1a#2p}}%
209             \KFLT@optionalname{\csuse{KFLT@#1a#2f}}%
210             \csuse{KFLT@#1a#2l}%
211             \csuse{KFLT@#1a#2s}%
212         }%
213 %     \end{minipage}%
214         \end{BlockClass}
215 %         \par\addvspace{2ex}%
216     }
217
218 }% tocdata not loaded
```

Env   KFLT@marginfloat   [⟨*offset*⟩] {⟨*type*⟩}

```
219 \DeclareDocumentEnvironment{KFLT@marginfloat}{O{-1.2ex} m}
220 {%
221     \uselengthunit{PT}%
222     \LWR@BlockClassWP%
223         {float:right; width:2in; margin:10pt}%
224         {}%
225         (note)%
226         {marginblock}%
227     \renewcommand*{\@captype}{#2}%
228     \minipage{1.2\LWR@usersmarginparwidth}%
```

```
229     \setlength{\marginparwidth}{.95\LWR@usersmarginparwidth}%
230 }
231 {%
232     \endminipage%
233     \endLWR@BlockClassWP%
234 }


235 \DeclareDocumentEnvironment{marginfigure}{o}
236     {\begin{KFLT@marginfloat}{figure}}
237     {\end{KFLT@marginfloat}}
238
239 \DeclareDocumentEnvironment{margintable}{o}
240     {\begin{KFLT@marginfloat}{table}}
241     {\end{KFLT@marginfloat}}
```

Env keywrap   {⟨*width*⟩} {⟨*keyfloat*⟩}

```
242 \DeclareDocumentEnvironment{keywrap}{m +m}
243 {%
244     \begin{LWR@setvirtualpage}*
245     \setlength{\LWR@templengthone}{#1}%
246     \begin{LWR@BlockClassWP}%
247         {%
248         float:right; width:\LWR@printlength{\LWR@templengthone}; % extra space
249             margin:10pt%
250         }%
251         {}%
252         (note)%
253         {marginblock}%
254     \setlength{\linewidth}{.95\LWR@templengthone}%
255     \booltrue{KFLT@keywrap}%
256     #2%
257     \end{LWR@BlockClassWP}%
258     \end{LWR@setvirtualpage}%
259 }
260 {}


261 }% AtBeginDocument
```

File 243   **`lwarp-keystroke.sty`**

§ 355   Package   # keystroke

(*Emulates or patches code by* WERNER FINK.)

keystroke (*Pkg*)   keystroke is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{keystroke}[2010/04/23]


```
2 \newcommand*{\LWR@HTML@keystroke}[1]{
3     \InlineClass{keystroke}{#1}
4 }
5 \LWR@formatted{keystroke}
6
7
8 \newcommand*{\LWR@HTML@Return}{\keystroke{\HTMLunicode{021A9}}}
```

```
 9 \LWR@formatted{Return}
10
11 \newcommand*{\LWR@HTML@BSpace}{\keystroke{\HTMLunicode{027FB}}}
12 \LWR@formatted{BSpace}
13
14 \newcommand*{\LWR@HTML@Tab}{\keystroke{|\HTMLunicode{021C6}|}}
15 \LWR@formatted{Tab}
16
17 \newcommand*{\LWR@HTML@UArrow}{\keystroke{\HTMLunicode{02191}}}
18 \LWR@formatted{UArrow}
19
20 \newcommand*{\LWR@HTML@DArrow}{\keystroke{\HTMLunicode{02193}}}
21 \LWR@formatted{DArrow}
22
23 \newcommand*{\LWR@HTML@LArrow}{\keystroke{\HTMLunicode{02190}}}
24 \LWR@formatted{LArrow}
25
26 \newcommand*{\LWR@HTML@RArrow}{\keystroke{\HTMLunicode{02192}}}
27 \LWR@formatted{RArrow}
28
29 % Preserves the language options:
30 \VerifyCommand[lwarp][keystroke]{\Shift}{F86359C350A5BC1D264A4997F86C2DC2}
31
32 \LetLtxMacro\LWR@HTML@Shift\Shift
33 \xpatchcmd{\LWR@HTML@Shift}
34     {$\Uparrow$}
35     {\HTMLunicode{21D1}}
36     {}
37     {}
38 \LWR@formatted{Shift}
39
40 \VerifyCommand[lwarp][keystroke]{\PgUp}{CBB81948EFB5940DAD2B51644BB4B2BF}
41
42 \LetLtxMacro\LWR@HTML@PgUp\PgUp
43 \xpatchcmd{\LWR@HTML@PgUp}
44     {$\uparrow$}
45     {\HTMLunicode{2191}}
46     {}
47     {}
48 \LWR@formatted{PgUp}
49
50 \VerifyCommand[lwarp][keystroke]{\PgDown}{B55C849642BE07904975EC7E4D649CAD}
51
52 \LetLtxMacro\LWR@HTML@PgDown\PgDown
53 \xpatchcmd{\LWR@HTML@PgDown}
54     {$\downarrow$}
55     {\HTMLunicode{2193}}
56     {}
57     {}
58 \LWR@formatted{PgDown}
```

File 244   **`lwarp-kpfonts.sty`**

§ 356   Package   # kpfonts

(*Emulates or patches code by* Christophe Caignaert.)

kpfonts (*Pkg*)   kpfonts is used as-is for svg math, and is emulated for MathJax.

⚠ limitations The MATHJAX emulation honors the options uprightRoman for \D only, classicReIm, frenchstyle for Greek only, upright for Greek only, uprightgreeks, slantedGreeks, and mathcalasscript.

The dedicated macros for Greek work correctly.

SVG math should appear the same as the printed output.

for HTML output:

```
1 \LWR@ProvidesPackagePass{kpfonts}[2010/08/20]
2
3 \LWR@infoprocessingmathjax{kpfonts}
4
5 \LWR@origRequirePackage{lwarp-common-mathjax-newpxtxmath}
6
7 \LWR@origRequirePackage{lwarp-common-mathjax-letters}
8
9 \begin{warpMathJax}
10
11 \ifkp@calasscr
12     \CustomizeMathJax{\let\LWRorigmathscr\mathscr}
13     \CustomizeMathJax{\let\LWRorigmathcal\mathcal}
14     \CustomizeMathJax{\let\mathscr\LWRorigmathcal}
15     \CustomizeMathJax{\let\mathcal\LWRorigmathscr}
16 \fi
17
18 \ifkp@upgrk % lowercase
19     \LWR@mathjax@addgreek@l@up{}{}
20     \LWR@mathjax@addgreek@l@it{other}{}
21 \else
22     \LWR@mathjax@addgreek@l@up{other}{}
23 \fi
24
25 \ifkp@slGrk
26     \LWR@mathjax@addgreek@u@it*{}{}
27     \LWR@mathjax@addgreek@u@up*{other}{}
28     \LWR@mathjax@addgreek@u@up*{var}{}
29 \else
30     \LWR@mathjax@addgreek@u@it*{other}{}
31     \LWR@mathjax@addgreek@u@it*{var}{}
32 \fi
33
34 \LWR@mathjax@addgreek@u@up*{}{up}
35 \LWR@mathjax@addgreek@l@up{}{up}
36
37 \LWR@mathjax@addgreek@u@it*{}{sl}
38 \LWR@mathjax@addgreek@l@it{}{sl}
39
40 \CustomizeMathJax{\newcommand{\partialsl}{\mathord{\unicode{x1D715}}}}
41 \CustomizeMathJax{\let\partialup\uppartial}% not upright
42
43 \ifkp@oldReIm
44 \else
45     \CustomizeMathJax{\renewcommand{\Re}{\mathfrak{Re}}}
46     \CustomizeMathJax{\renewcommand{\Im}{\mathfrak{Im}}}
47 \fi
48
49 \ifkp@Dcommand
50     \ifkp@upRm%
51         \CustomizeMathJax{
52             \def\D#1{\mathclose{\,\mathrm{d}}#1}
```

```
53            }
54        \else
55           \CustomizeMathJax{
56                \def\D#1{\mathclose{\,\mathit{d}}#1}
57           }
58        \fi
59 \fi
60
61 \CustomizeMathJax{\let\pounds\mathsterling}
62 \CustomizeMathJax{\let\kppounds\mathsterling}
63
64 \CustomizeMathJax{\newcommand{\mathup}[1]{\mathrm{#1}}}% never sans
65 \CustomizeMathJax{\let\mathupright\mathup}
66
67 \end{warpMathJax}
```

File 245   **`lwarp-kpfonts-otf.sty`**

§ 357   Package   # kpfonts-otf

(*Emulates or patches code by* DANIEL FLIPO.)

kpfonts-otf (*Pkg*)   kpfonts-otf is used as-is for SVG math, and is emulated for MATHJAX.

⚠   limitations   The MATHJAX emulation honors the options fancyReIm, mathcal, frenchstyle for Greek only, and mathcalasscript.

Also see the options for unicode-math, which is loaded by kpfonts-otf.

The unicode-math dedicated macros for Greek work correctly.

⚠   \mathversion   The MATHJAX emulation does not change with the use of \mathversion. Whatever emulation is established at the begin of the document will remain.

SVG math should appear the same as the printed output.

for HTML output:
```
1 \LWR@ProvidesPackagePass{kpfonts-otf}[2020/06/20]
2
3 \LWR@infoprocessingmathjax{kpfonts-otf}
4
5 \LWR@origRequirePackage{lwarp-common-mathjax-nonunicode}
6
7 \LWR@origRequirePackage{lwarp-common-mathjax-letters}
8
9 \begin{warpMathJax}
10
11 \ifkp@calasscr
12     \CustomizeMathJax{\let\mathscr\mathcal}
13 \else
14     \CustomizeMathJax{\let\mathcal\mathscr}
15 \fi
16
17 \ifkp@frenchstyle
18     \LWR@mathjax@addgreek@l@up{}{}
19     \LWR@mathjax@addgreek@u@up*{}{}
20 \fi
21
22 \ifkp@oldReIm
```

```
23      \CustomizeMathJax{\renewcommand{\Re}{\mathfrak{Re}}}
24      \CustomizeMathJax{\renewcommand{\Im}{\mathfrak{Im}}}
25 \else
26 \fi
27
28 \ifkp@Dcommand
29      \CustomizeMathJax{
30          \def\D#1{\mathclose{\,\mathrm{d}}#1}
31      }
32 \fi
33
34 \CustomizeMathJax{\let\varint\int}
35 \CustomizeMathJax{\let\variint\iint}
36 \CustomizeMathJax{\let\variiint\iiint}
37 \CustomizeMathJax{\let\variiiint\iiiint}
38 \CustomizeMathJax{\let\varidotsint\idotsint}
39
40 \CustomizeMathJax{\newcommand{\varointctrclockwise}{%
41      \mathop{\unicode{x2939}\!\!\unicode{x0222E}}%
42 }}
43
44 \CustomizeMathJax{\newcommand{\oiintclockwise}{%
45      \mathop{\unicode{x0222F}\!\!\unicode{x2938}}%
46 }}
47
48 \CustomizeMathJax{\newcommand{\oiintctrclockwise}{%
49      \mathop{\unicode{x2939}\!\!\unicode{x0222F}}%
50 }}
51
52 \CustomizeMathJax{\newcommand{\varoiintclockwise}{%
53      \mathop{\unicode{x0222F}\!\!\unicode{x2938}}%
54 }}
55
56 \CustomizeMathJax{\newcommand{\varoiintctrclockwise}{%
57      \mathop{\unicode{x2939}\!\!\unicode{x0222F}}%
58 }}
59
60 \CustomizeMathJax{\newcommand{\oiiintclockwise}{%
61      \mathop{\unicode{x02230}\!\!\unicode{x2938}}%
62 }}
63
64 \CustomizeMathJax{\newcommand{\oiiintctrclockwise}{%
65      \mathop{\unicode{x2939}\!\!\unicode{x02230}}%
66 }}
67
68 \CustomizeMathJax{\newcommand{\varoiiintclockwise}{%
69      \mathop{\unicode{x02230}\!\!\unicode{x2938}}%
70 }}
71
72 \CustomizeMathJax{\newcommand{\varoiiintctrclockwise}{%
73      \mathop{\unicode{x2939}\!\!\unicode{x02230}}%
74 }}
75
76 \CustomizeMathJax{\newcommand{\sqiint}{%
77      \mathop{\unicode{x2A16}\!\!\unicode{x2A16}}%
78 }}
79
80 \CustomizeMathJax{\newcommand{\sqiiint}{%
81      \mathop{\unicode{x2A16}\!\!\unicode{x2A16}\!\!\unicode{x2A16}}%
82 }}
```

```
83
84 \CustomizeMathJax{\let\widearc\overparen}
85 \CustomizeMathJax{\let\widearcarrow\overrightarrow}
86 \CustomizeMathJax{\let\overrightarc\overrightarrow}
87
88 \end{warpMathJax}
```

File 246   **`lwarp-layaureo.sty`**

§ 358   Package   # layaureo

layaureo (*Pkg*)   layaureo is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{layaureo}[2004/09/16]

File 247   **`lwarp-layout.sty`**

§ 359   Package   # layout

layout (*Pkg*)   layout is ignored.

for HTML output: Discard all options for lwarp-layout:

1 \LWR@ProvidesPackageDrop{layout}[2014/10/28]

2 \NewDocumentCommand{\layout}{s}{}

File 248   **`lwarp-layouts.sty`**

§ 360   Package   # layouts

layouts (*Pkg*)   layouts is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{layouts}[2009/09/02]

```
 2 \newif\ifoddpagelayout
 3   \oddpagelayouttrue
 4 \newif\iftwocolumnlayout
 5   \twocolumnlayoutfalse
 6 \newif\ifdrawmarginpars
 7   \drawmarginparstrue
 8 \newif\ifdrawparameters
 9   \drawparameterstrue
10 \newif\iflistaspara
11   \listasparatrue
12 \newif\ifruninhead
13   \runinheadfalse
14 \newif\ifprintparameters
15   \printparameterstrue
16 \newif\ifdrawdimensions
17   \drawdimensionsfalse
```

```
18 \newif\ifprintheadings
19   \printheadingstrue
20 \newcommand{\testdrawdimensions}{}
21 \newcommand{\testprintparameters}{}
22 \newcommand{\setlabelfont}[1]{}
23 \newcommand{\setparametertextfont}[1]{}
24 \newcommand{\setvaluestextsize}[1]{}
25 \newcommand{\setlayoutscale}[1]{}
26 \newcommand{\setuplayouts}{}
27 \newcommand{\printinunitsof}[1]{}
28 \newcommand{\prntlen}[1]{}
29 \newcommand{\trypaperwidth}[1]{}
30 \newcommand{\trypaperheight}[1]{}
31 \newcommand{\tryhoffset}[1]{}
32 \newcommand{\tryvoffset}[1]{}
33 \newcommand{\trytopmargin}[1]{}
34 \newcommand{\tryheadheight}[1]{}
35 \newcommand{\tryheadsep}[1]{}
36 \newcommand{\trytextheight}[1]{}
37 \newcommand{\tryfootskip}[1]{}
38 \newcommand{\tryoddsidemargin}[1]{}
39 \newcommand{\tryevensidemargin}[1]{}
40 \newcommand{\trytextwidth}[1]{}
41 \newcommand{\trymarginparsep}[1]{}
42 \newcommand{\trymarginparwidth}[1]{}
43 \newcommand{\trymarginparpush}[1]{}
44 \newcommand{\trycolumnsep}[1]{}
45 \newcommand{\trycolumnseprule}[1]{}
46 \newcommand{\setfootbox}[2]{}
47 \newcommand{\currentpage}{}
48 \newcommand{\drawpage}{(draw page)}
49 \newcommand{\pagediagram}{(page diagram)}
50 \newcommand{\pagedesign}{(page design)}
51 \newcommand{\pagevalues}{(page values)}
52 \newcommand{\trystockwidth}[1]{}
53 \newcommand{\trystockheight}[1]{}
54 \newcommand{\trytrimedge}[1]{}
55 \newcommand{\trytrimtop}[1]{}
56 \newcommand{\tryuppermargin}[1]{}
57 \newcommand{\tryspinemargin}[1]{}
58 \newcommand{\currentstock}{}
59 \newcommand{\drawstock}{(draw stock)}
60 \newcommand{\stockdiagram}{(stock diagram)}
61 \newcommand{\stockdesign}{(stock design)}
62 \newcommand{\stockvalues}{(stock values)}
63 \newcommand{\tryitemindent}[1]{}
64 \newcommand{\trylabelwidth}[1]{}
65 \newcommand{\trylabelsep}[1]{}
66 \newcommand{\tryleftmargin}[1]{}
67 \newcommand{\tryrightmargin}[1]{}
68 \newcommand{\trylistparindent}[1]{}
69 \newcommand{\trytopsep}[1]{}
70 \newcommand{\tryparskip}[1]{}
71 \newcommand{\trypartopsep}[1]{}
72 \newcommand{\tryparsep}[1]{}
73 \newcommand{\tryitemsep}[1]{}
74 \newcommand{\currentlist}{}
75 \newcommand{\drawlist}{(draw list)}
76 \newcommand{\listdiagram}{(list diagram)}
77 \newcommand{\listdesign}{(list design)}
```

```
 78 \newcommand{\listvalues}{(list values)}
 79 \newcommand{\tryfootins}[1]{}
 80 \newcommand{\tryfootnotesep}[1]{}
 81 \newcommand{\tryfootnotebaseline}[1]{}
 82 \newcommand{\tryfootruleheight}[1]{}
 83 \newcommand{\tryfootrulefrac}[1]{}
 84 \newcommand{\currentfootnote}{}
 85 \newcommand{\drawfootnote}{(draw footnote)}
 86 \newcommand{\footnotediagram}{(footnote diagram)}
 87 \newcommand{\footnotedesign}{(footnote design)}
 88 \newcommand{\footnotevalues}{(footnote values)}
 89 \newcommand{\tryparindent}[1]{}
 90 \newcommand{\tryparlinewidth}[1]{}
 91 \newcommand{\tryparbaselineskip}[1]{}
 92 \newcommand{\currentparagraph}{}
 93 \newcommand{\drawparagraph}{(draw paragraph)}
 94 \newcommand{\paragraphdiagram}{(paragraph diagram)}
 95 \newcommand{\paragraphdesign}{(paragraph design)}
 96 \newcommand{\paragraphvalues}{(paragraph values)}
 97 \newcommand{\trybeforeskip}[1]{}
 98 \newcommand{\tryafterskip}[1]{}
 99 \newcommand{\tryindent}[1]{}
100 \newcommand{\currentheading}{}
101 \newcommand{\drawheading}[1]{(draw heading)}
102 \newcommand{\headingdiagram}[1]{(heading diagram)}
103 \newcommand{\headingdesign}[1]{(heading design)}
104 \newcommand{\headingvalues}{(heading values)}
105 \newcommand{\trytextfloatsep}[1]{}
106 \newcommand{\tryfloatsep}[1]{}
107 \newcommand{\tryintextsep}[1]{}
108 \newcommand{\trytopfigrule}[1]{}
109 \newcommand{\trybotfigrule}[1]{}
110 \newcommand{\currentfloat}{}
111 \newcommand{\drawfloat}{(draw float)}
112 \newcommand{\floatdiagram}{(float diagram)}
113 \newcommand{\floatdesign}{(float design)}
114 \newcommand{\floatvalues}{(float values)}
115 \newcommand{\trytotalnumber}[1]{}
116 \newcommand{\trytopnumber}[1]{}
117 \newcommand{\trybottomnumber}[1]{}
118 \newcommand{\trytopfraction}[1]{}
119 \newcommand{\trytextfraction}[1]{}
120 \newcommand{\trybottomfraction}[1]{}
121 \newcommand{\currentfloatpage}{}
122 \newcommand{\drawfloatpage}{(draw floatpage)}
123 \newcommand{\floatpagediagram}{(floatpage diagram)}
124 \newcommand{\floatpagedesign}{(floatpage design)}
125 \newcommand{\floatpagevalues}{(floatpage values)}
126 \newcommand{\trytocindent}[1]{}
127 \newcommand{\trytocnumwidth}[1]{}
128 \newcommand{\trytoclinewidth}[1]{}
129 \newcommand{\trytocrmarg}[1]{}
130 \newcommand{\trytocpnumwidth}[1]{}
131 \newcommand{\trytocdotsep}[1]{}
132 \newcommand{\currenttoc}{}
133 \newcommand{\drawtoc}{(draw toc)}
134 \newcommand{\tocdiagram}{(toc diagram)}
135 \newcommand{\tocdesign}{(toc design)}
136 \newcommand{\tocvalues}{(toc values)}
137 \newcommand{\drawaspread}[8][0]{(a spread)}
```

```
138 \newcommand{\drawfontframe}[1]{(font frame)}
139 \newcommand{\drawfontframelabel}[1]{}
```

File 249   **`lwarp-leading.sty`**

§ 361   Package   # leading

leading (*Pkg*)   leading is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{leading}[2008/12/11]`

`2 \newcommand\leading[1]{}`

File 250   **`lwarp-leftidx.sty`**

§ 362   Package   # leftidx

(*Emulates or patches code by* HARALD HARDERS.)

leftidx (*Pkg*)   leftidx works as-is with svg math, and is emulated for MATHJAX.

for HTML output: `1 \LWR@ProvidesPackagePass{leftidx}[2003/09/24]`

```
2 \begin{warpMathJax}
3 \CustomizeMathJax{\newcommand{\leftidx}[3]{{\vphantom{#2}}#1#2#3}}
4 \CustomizeMathJax{\newcommand{\ltrans}[1]{\leftidx{^\mathrm{t}}{\!#1}{}}}
5 \end{warpMathJax}
```

File 251   **`lwarp-letterspace.sty`**

§ 363   Package   # letterspace

(*Emulates or patches code by* R SCHLICHT.)

letterspace (*Pkg*)   letterspace is a subset of microtype, which is pre-loaded by lwarp. All user options and macros are ignored and disabled.

for HTML output: Discard all options for lwarp-letterspace:

`1 \LWR@ProvidesPackageDrop{letterspace}[2018/01/14]`

```
2 \newcommand*\lsstyle{}
3 \newcommand\textls[2][]{}
4 \def\textls#1#{}
5 \newcommand*\lslig[1]{#1}
```

File 252   **`lwarp-lettrine.sty`**

§ 364   Package   # lettrine

(*Emulates or patches code by* DANIEL FLIPO.)

lettrine (*Pkg*)   lettrine is emulated.

Discard all options for lwarp-lettrine:

```
1 \LWR@ProvidesPackageDrop{lettrine}[2018-08-28]
```

The initial letter is in a <span> of class lettrine, and the following text is in a <span> of class lettrinetext. \lettrine [⟨*keys*⟩] {⟨*letter*⟩} {⟨*additional text*⟩}

```
2 \DeclareDocumentCommand{\lettrine}{o m m}{%
3     \InlineClass{lettrine}{#2}\InlineClass{lettrinetext}{#3} % extra space
4 }
5
6 \newcounter{DefaultLines}
7 \setcounter{DefaultLines}{2}
8 \newcounter{DefaultDepth}
9 \newcommand*{\DefaultOptionsFile}{\relax}
10 \newcommand*{\DefaultLoversize}{0}
11 \newcommand*{\DefaultLraise}{0}
12 \newcommand*{\DefaultLhang}{0}
13 \newdimen\DefaultFindent
14 \setlength{\DefaultFindent}{\z@}
15 \newdimen\DefaultNindent
16 \setlength{\DefaultNindent}{0.5em}
17 \newdimen\DefaultSlope
18 \setlength{\DefaultSlope}{\z@}
19 \newdimen\DiscardVskip
20 \setlength{\DiscardVskip}{0.2\p@}
21 \newif\ifLettrineImage
22 \newif\ifLettrineOnGrid
23 \newif\ifLettrineRealHeight
24
25 \newcommand*{\LettrineTextFont}{\scshape}
26 \newcommand*{\LettrineFontHook}{}
27 \newcommand*{\LettrineFont}[1]{\InlineClass{lettrine}{#1}}
28 \newcommand*{\LettrineFontEPS}[1]{\includegraphics[height=1.5ex]{#1}}
```

File 253   **`lwarp-libertinust1math.sty`**

§ 365   Package   # libertinust1math

(*Emulates or patches code by* MICHAEL SHARPE.)

libertinust1math (*Pkg*)   libertinust1math is used as-is for svg math, and is emulated for MATHJAX.

The MATHJAX emulation honors frenchmath for Greek but not Latin characters, and slantedGreek, uprightGreek, and ISO also adjust Greek characters. MATHJAX cannot yet honor options for adjusting Latin characters.

The dedicated macros for upright and italic Greek letters do work correctly.

Some of the symbol font macros such as \mathsfbf do not use a sans font because MATHJAX does not yet have sans Greek.

svg math honors all font choices, and should appear the same as the printed output.

```
 1 \LWR@ProvidesPackagePass{libertinust1math}[2020/06/10]
 2
 3 \LWR@infoprocessingmathjax{libertinust1math}

 4 \LWR@origRequirePackage{lwarp-common-mathjax-letters}
 5
 6 \begin{warpMathJax}
 7
 8 \iflibus@slantedG
 9     \LWR@mathjax@addgreek@u@it*{}{}
10 \else
11     \LWR@mathjax@addgreek@u@up*{}{}
12 \fi
13
14 \LWR@mathjax@addgreek@u@it*{}{it}
15 \LWR@mathjax@addgreek@u@up*{up}{}
16 \LWR@mathjax@addgreek@u@up*{}{up}
17
18 \iflibus@frenchm
19     \LWR@mathjax@addgreek@l@up{}{}
20 \else
21     \LWR@mathjax@addgreek@l@it{}{}
22 \fi
23
24 \LWR@mathjax@addgreek@l@it{}{it}
25 \LWR@mathjax@addgreek@l@up{}{up}
26 \LWR@mathjax@addgreek@l@up{up}{}
27
28 \CustomizeMathJax{\let\uppartial\partial}% not upright


29 \CustomizeMathJax{\let\mathsfbf\mathbf}% not sans
30 % \CustomizeMathJax{\newcommand{\mathsfbf}[1]{%
31 %     \mmlToken{mi}[mathvariant="bold-sans-serif"]{#1}% not greek
32 % }}% not sans
33
34 % \CustomizeMathJax{\newcommand{\mathbfit}[1]{\boldsymbol{#1}}}
35 \CustomizeMathJax{\let\mathbfit\boldsymbol}


36 % \CustomizeMathJax{\newcommand{\mathsfbfit}[1]{\boldsymbol{#1}}}% not sans
37 \CustomizeMathJax{\let\mathsfbfit\mathbfit}% not sans
38 % \CustomizeMathJax{\newcommand{\mathsfbfit}[1]{%
39 %     \mmlToken{mi}[mathvariant="sans-serif-bold-italic"]{#1}% not greek
40 % }}%


41 \CustomizeMathJax{\let\mathsfit\mathit}% not sans
42 % \CustomizeMathJax{\newcommand{\mathsfit}[1]{%
43 %     \mmlToken{mi}[mathvariant="sans-serif-italic"]{#1}% not greek
44 % }}
45
46 \CustomizeMathJax{\let\vectorsym\mathbfit}
47 \CustomizeMathJax{\let\matrixsym\mathbfit}
48 \CustomizeMathJax{\let\tensorsym\mathsfbfit}
49 \CustomizeMathJax{\let\mathboldsans\mathsfbfit}
50 \CustomizeMathJax{\let\mathbold\mathbfit}
```

lwarp_mathjax.txt adds \left/\right support for delimiters.

```
51 \CustomizeMathJax{\let\dlb\lBrack}
52 \CustomizeMathJax{\let\drb\rBrack}
53
54 \CustomizeMathJax{\let\sqrtsign\sqrt}
55
56 \CustomizeMathJax{\let\smallintsl\smallint}
57 \CustomizeMathJax{\newcommand{\smalliintsl}{\mathop{\unicode{x222C}}\limits}}
58 \CustomizeMathJax{\newcommand{\smalliiintsl}{\mathop{\unicode{x222D}}\limits}}
59 \CustomizeMathJax{\newcommand{\smalliiiintsl}{\mathop{\unicode{x2A0C}}\limits}}
60 \CustomizeMathJax{\newcommand{\smallointsl}{\mathop{\unicode{x222E}}\limits}}
61 \CustomizeMathJax{\newcommand{\smalloiintsl}{\mathop{\unicode{x222F}}\limits}}
62
63 \CustomizeMathJax{\let\smallintup\smallint}
64 \CustomizeMathJax{\newcommand{\smalliintup}{\mathop{\unicode{x222C}}\limits}}
65 \CustomizeMathJax{\newcommand{\smalliiintup}{\mathop{\unicode{x222D}}\limits}}
66 \CustomizeMathJax{\newcommand{\smalliiiintup}{\mathop{\unicode{x2A0C}}\limits}}
67 \CustomizeMathJax{\newcommand{\smallointup}{\mathop{\unicode{x222E}}\limits}}
68 \CustomizeMathJax{\newcommand{\smalloiintup}{\mathop{\unicode{x222F}}\limits}}
69
70 \CustomizeMathJax{\let\intslop\int}
71 \CustomizeMathJax{\newcommand{\iintslop}{\mathop{\unicode{x222C}}\limits}}
72 \CustomizeMathJax{\newcommand{\iiintslop}{\mathop{\unicode{x222D}}\limits}}
73 \CustomizeMathJax{\newcommand{\iiiintslop}{\mathop{\unicode{x2A0C}}\limits}}
74 \CustomizeMathJax{\let\ointslop\oint}
75 \CustomizeMathJax{\newcommand{\oiintslop}{\mathop{\unicode{x222F}}\limits}}
76 \CustomizeMathJax{\newcommand{\oiiintslop}{\mathop{\unicode{x2230}}\limits}}
77
78 \CustomizeMathJax{\let\intupop\int}
79 \CustomizeMathJax{\newcommand{\iintupop}{\mathop{\unicode{x222C}}\limits}}
80 \CustomizeMathJax{\newcommand{\iiintupop}{\mathop{\unicode{x222D}}\limits}}
81 \CustomizeMathJax{\newcommand{\iiiintupop}{\mathop{\unicode{x2A0C}}\limits}}
82 \CustomizeMathJax{\let\ointupop\oint}
83 \CustomizeMathJax{\newcommand{\oiintupop}{\mathop{\unicode{x222F}}\limits}}
84 \CustomizeMathJax{\newcommand{\oiiintupop}{\mathop{\unicode{x2230}}\limits}}
85
86 \CustomizeMathJax{\newcommand{\smalliint}{\mathop{\unicode{x222C}}\limits}}
87 \CustomizeMathJax{\newcommand{\smalliiint}{\mathop{\unicode{x222D}}\limits}}
88 \CustomizeMathJax{\newcommand{\smalliiiint}{\mathop{\unicode{x2A0C}}\limits}}
89 \CustomizeMathJax{\newcommand{\smalloint}{\mathop{\unicode{x222E}}\limits}}
90 \CustomizeMathJax{\newcommand{\smalloiint}{\mathop{\unicode{x222F}}\limits}}
91
92 \CustomizeMathJax{\let\intop\int}
93 \CustomizeMathJax{\newcommand{\iintop}{\mathop{\unicode{x222C}}\limits}}
94 \CustomizeMathJax{\newcommand{\iiintop}{\mathop{\unicode{x222D}}\limits}}
95 \CustomizeMathJax{\newcommand{\iiiintop}{\mathop{\unicode{x2A0C}}\limits}}
96 \CustomizeMathJax{\let\ointop\oint}
97 \CustomizeMathJax{\newcommand{\oiintop}{\mathop{\unicode{x222F}}\limits}}
98 \CustomizeMathJax{\newcommand{\oiiintop}{\mathop{\unicode{x2230}}\limits}}
99
100 \CustomizeMathJax{\newcommand{\oiint}{\mathop{\unicode{x222F}}\limits}}
101
102 \CustomizeMathJax{\newcommand{\bigcupdot}{\mathop{\unicode{x2A03}}}}
103 \CustomizeMathJax{\newcommand{\bigsqcap}{\mathop{\unicode{x2A05}}}}
104 \CustomizeMathJax{\newcommand{\xsol}{\mathop{\unicode{x29F8}}}}
105 \CustomizeMathJax{\newcommand{\xbsol}{\mathop{\unicode{x29F9}}}}
106 \CustomizeMathJax{\let\prodop\prod}
107 \CustomizeMathJax{\let\coprodop\coprod}
108 \CustomizeMathJax{\let\sumop\sum}
109 \CustomizeMathJax{\let\bigwedgeop\bigwedge}
110 \CustomizeMathJax{\let\bigveeop\bigvee}
```

```
111 \CustomizeMathJax{\let\bigcapop\bigcap}
112 \CustomizeMathJax{\let\bigcupop\bigcup}
113 \CustomizeMathJax{\let\xsolop\xsol}
114 \CustomizeMathJax{\let\xbsolop\xbsol}
115 \CustomizeMathJax{\let\bigodotop\bigodot}
116 \CustomizeMathJax{\let\bigoplusop\bigoplus}
117 \CustomizeMathJax{\let\bigotimesop\bigotimes}
118 \CustomizeMathJax{\let\bigcupdotop\bigcupdot}
119 \CustomizeMathJax{\let\biguplusop\biguplus}
120 \CustomizeMathJax{\let\bigsqcapop\bigsqcap}
121 \CustomizeMathJax{\let\bigsqcupop\bigsqcup}
122
123 \CustomizeMathJax{\newcommand{\ovhook}[1]{\mathord{#1\unicode{x00309}}}}
124 \CustomizeMathJax{\newcommand{\candra}[1]{\mathord{#1\unicode{x00310}}}}
125 \CustomizeMathJax{\newcommand{\oturnedcomma}[1]{\mathord{#1\unicode{x00312}}}}
126 \CustomizeMathJax{\newcommand{\ocommatopright}[1]{\mathord{#1\unicode{x00315}}}}
127 \CustomizeMathJax{\newcommand{\droang}[1]{\mathord{#1\unicode{x0031A}}}}
128 \CustomizeMathJax{\newcommand{\leftharpoonaccent}[1]{\mathord{#1\unicode{x020D0}}}}
129 \CustomizeMathJax{\newcommand{\rightharpoonaccent}[1]{\mathord{#1\unicode{x020D1}}}}
130 \CustomizeMathJax{\newcommand{\leftarrowaccent}[1]{\mathord{#1\unicode{x020D0}}}}
131 \CustomizeMathJax{\let\rightarrowaccent\vec}
132
133 \CustomizeMathJax{\newcommand{\leftrightarrowaccent}[1]{\mathord{#1\unicode{x020E1}}}}
134 \CustomizeMathJax{\newcommand{\annuity}[1]{\mathord{#1\unicode{x020E7}}}}
135 \CustomizeMathJax{\newcommand{\widebridgeabove}[1]{\mathord{#1\unicode{x020E9}}}}
136 \CustomizeMathJax{\newcommand{\asteraccent}[1]{\mathord{#1\unicode{x020F0}}}}
137
138 % neutralized:
139 \CustomizeMathJax{\newcommand{\braceld}{}}
140 \CustomizeMathJax{\newcommand{\bracerd}{}}
141 \CustomizeMathJax{\newcommand{\bracelu}{}}
142 \CustomizeMathJax{\newcommand{\braceru}{}}
143 \CustomizeMathJax{\newcommand{\braceex}{}}
144 \CustomizeMathJax{\newcommand{\bracemu}{}}
145 \CustomizeMathJax{\newcommand{\bracemd}{}}
146 \CustomizeMathJax{\newcommand{\parenld}{}}
147 \CustomizeMathJax{\newcommand{\parenrd}{}}
148 \CustomizeMathJax{\newcommand{\parenlu}{}}
149 \CustomizeMathJax{\newcommand{\parenru}{}}
150 \CustomizeMathJax{\newcommand{\bracketld}{}}
151 \CustomizeMathJax{\newcommand{\bracketrd}{}}
152 \CustomizeMathJax{\newcommand{\bracketlu}{}}
153 \CustomizeMathJax{\newcommand{\bracketru}{}}
154 \CustomizeMathJax{\newcommand{\bracketex}{}}
155 \CustomizeMathJax{\newcommand{\parenex}{}}
156
157 \CustomizeMathJax{\newcommand{lhook}{~}}
158 \CustomizeMathJax{\newcommand{rhook}{~}}
159 \CustomizeMathJax{\newcommand{relbar}{-}}
160 \CustomizeMathJax{\newcommand{Relbar}{=}}
161
162 \CustomizeMathJax{\newcommand{\mapstochar}{\mathrel{\unicode{x21A6}}}}
163
164 \CustomizeMathJax{\newcommand{\Zbar}{\mathord{\unicode{x0001B5}}}}
165 \CustomizeMathJax{\newcommand{\notchar}{\mathrel{\unicode{x000AC}}}}
166 \CustomizeMathJax{\newcommand{\upbackepsilon}{\mathord{\unicode{x03F6}}}}
167 \CustomizeMathJax{\newcommand{\smblkcircle}{\mathbin{\unicode{x02022}}}}
168 \CustomizeMathJax{\newcommand{\enleadertwodots}{\mathord{\unicode{x02025}}}}
169 \CustomizeMathJax{\newcommand{\unicodeellipsis}{\mathord{\unicode{x02026}}}}
170 \CustomizeMathJax{\newcommand{\mathellipsis}{\mathinner{\unicode{x02026}}}}
```

```
171 \CustomizeMathJax{\newcommand{\dprime}{\mathord{\unicode{x02033}}}}
172 \CustomizeMathJax{\newcommand{\trprime}{\mathord{\unicode{x02034}}}}
173 \CustomizeMathJax{\newcommand{\backdprime}{\mathord{\unicode{x02036}}}}
174 \CustomizeMathJax{\newcommand{\backtrprime}{\mathord{\unicode{x02037}}}}
175 \CustomizeMathJax{\newcommand{\caretinsert}{\mathord{\unicode{x02038}}}}
176 \CustomizeMathJax{\newcommand{\Exclam}{\mathord{\unicode{x0203C}}}}
177
178 \CustomizeMathJax{\newcommand{\hyphenbullet}{\mathord{\unicode{x02043}}}}
179 \CustomizeMathJax{\newcommand{\fracslash}{\mathbin{\unicode{x02044}}}}
180 \CustomizeMathJax{\newcommand{\Question}{\mathord{\unicode{x02047}}}}
181 \CustomizeMathJax{\newcommand{\closure}{\mathrel{\unicode{x02050}}}}
182 \CustomizeMathJax{\newcommand{\qprime}{\mathord{\unicode{x02057}}}}
183 \CustomizeMathJax{\newcommand{\vertoverlay}{\mathrel{\unicode{x020D2}}}}
184 \CustomizeMathJax{\newcommand{\enclosecircle}{\mathord{\unicode{x020DD}}}}
185 \CustomizeMathJax{\newcommand{\enclosesquare}{\mathord{\unicode{x020DE}}}}
186 \CustomizeMathJax{\newcommand{\enclosetriangle}{\mathord{\unicode{x020E4}}}}
187 \CustomizeMathJax{\newcommand{\Eulerconst}{\mathord{\unicode{x02107}}}}
188 \CustomizeMathJax{\newcommand{\turnediota}{\mathord{\unicode{x02129}}}}
189 \CustomizeMathJax{\newcommand{\Angstrom}{\mathord{\unicode{x0212B}}}}
190
191 \CustomizeMathJax{\newcommand{\sansLturned}{\mathord{\unicode{x02142}}}}
192 \CustomizeMathJax{\newcommand{\sansLmirrored}{\mathord{\unicode{x02143}}}}
193 \CustomizeMathJax{\newcommand{\Yup}{\mathord{\unicode{x02144}}}}
194 \CustomizeMathJax{\newcommand{\upand}{\mathbin{\unicode{x0214B}}}}
195 \CustomizeMathJax{\newcommand{\increment}{\mathord{\unicode{x02206}}}}
196 \CustomizeMathJax{\newcommand{\smallin}{\mathrel{\unicode{x0220A}}}}
197 \CustomizeMathJax{\newcommand{\nni}{\mathrel{\unicode{x0220C}}}}
198
199 \CustomizeMathJax{\newcommand{\smallni}{\mathrel{\unicode{x0220D}}}}
200 \CustomizeMathJax{\newcommand{\QED}{\mathord{\unicode{x0220E}}}}
201 \CustomizeMathJax{\newcommand{\vysmwhtcircle}{\mathbin{\unicode{x02218}}}}
202 \CustomizeMathJax{\newcommand{\vysmblkcircle}{\mathbin{\unicode{x02219}}}}
203 \CustomizeMathJax{\newcommand{\rightangle}{\mathord{\unicode{x0221F}}}}
204
205 \CustomizeMathJax{\newcommand{\Colon}{\mathrel{\unicode{x02237}}}}
206 \CustomizeMathJax{\newcommand{\dotminus}{\mathbin{\unicode{x02238}}}}
207 \CustomizeMathJax{\newcommand{\dashcolon}{\mathrel{\unicode{x02239}}}}
208 \CustomizeMathJax{\newcommand{\dotsminusdots}{\mathrel{\unicode{x0223A}}}}
209 \CustomizeMathJax{\newcommand{\kernelcontraction}{\mathrel{\unicode{x0223B}}}}
210 \CustomizeMathJax{\newcommand{\invlazys}{\mathbin{\unicode{x0223E}}}}
211
212 \CustomizeMathJax{\newcommand{\sinewave}{\mathord{\unicode{x0223F}}}}
213 \CustomizeMathJax{\newcommand{\nsime}{\mathrel{\unicode{x02244}}}}
214 \CustomizeMathJax{\newcommand{\simneqq}{\mathrel{\unicode{x02246}}}}
215 \CustomizeMathJax{\newcommand{\napprox}{\mathrel{\unicode{x02249}}}}
216 \CustomizeMathJax{\newcommand{\approxident}{\mathrel{\unicode{x0224B}}}}
217 \CustomizeMathJax{\newcommand{\backcong}{\mathrel{\unicode{x0224C}}}}
218
219 \CustomizeMathJax{\newcommand{\nasymp}{\mathrel{\unicode{x0226D}}}}
220 \CustomizeMathJax{\newcommand{\nlesssim}{\mathrel{\unicode{x02274}}}}
221 \CustomizeMathJax{\newcommand{\ngtrsim}{\mathrel{\unicode{x02275}}}}
222 \CustomizeMathJax{\newcommand{\nlessgtr}{\mathrel{\unicode{x02278}}}}
223 \CustomizeMathJax{\newcommand{\ngtrless}{\mathrel{\unicode{x02279}}}}
224
225 \CustomizeMathJax{\newcommand{\nsubset}{\mathrel{\unicode{x02284}}}}
226 \CustomizeMathJax{\newcommand{\nsupset}{\mathrel{\unicode{x02285}}}}
227
228 \CustomizeMathJax{\newcommand{\cupleftarrow}{\mathbin{\unicode{x0228C}}}}
229 \CustomizeMathJax{\newcommand{\cupdot}{\mathbin{\unicode{x0228D}}}}
230 \CustomizeMathJax{\newcommand{\circledequal}{\mathbin{\unicode{x0229C}}}}
```

231
232 \CustomizeMathJax{\newcommand{\assert}{\mathrel{\unicode{x022A6}}}}
233 \CustomizeMathJax{\newcommand{\VDash}{\mathrel{\unicode{x022AB}}}}
234 \CustomizeMathJax{\newcommand{\prurel}{\mathrel{\unicode{x022B0}}}}
235
236 \CustomizeMathJax{\newcommand{\origof}{\mathrel{\unicode{x022B6}}}}
237 \CustomizeMathJax{\newcommand{\smallprod}{\mathop{\unicode{x0220F}}}}% not small
238 \CustomizeMathJax{\newcommand{\smallcoprod}{\mathop{\unicode{x02210}}}}% not small
239 \CustomizeMathJax{\newcommand{\smallsum}{\mathop{\unicode{x02211}}}}% not small
240 \CustomizeMathJax{\newcommand{\Hfraktur}{\mathord{\unicode{x1D525}}}}
241 \CustomizeMathJax{\newcommand{\dsol}{\mathbin{\unicode{x029F6}}}}
242 \CustomizeMathJax{\newcommand{\rsolbar}{\mathbin{\unicode{x029F7}}}}
243
244 \CustomizeMathJax{\newcommand{\eqless}{\mathrel{\unicode{x022DC}}}}
245 \CustomizeMathJax{\newcommand{\eqgtr}{\mathrel{\unicode{x022DD}}}}
246 \CustomizeMathJax{\newcommand{\npreccurlyeq}{\mathrel{\unicode{x022E0}}}}
247 \CustomizeMathJax{\newcommand{\nsucccurlyeq}{\mathrel{\unicode{x022E1}}}}
248 \CustomizeMathJax{\newcommand{\nsqsubseteq}{\mathrel{\unicode{x022E2}}}}
249 \CustomizeMathJax{\newcommand{\nsqsupseteq}{\mathrel{\unicode{x022E3}}}}
250 \CustomizeMathJax{\newcommand{\sqsubsetneq}{\mathrel{\unicode{x022E4}}}}
251 \CustomizeMathJax{\newcommand{\sqsupsetneq}{\mathrel{\unicode{x022E5}}}}
252 \CustomizeMathJax{\newcommand{\nvartriangleleft}{\mathrel{\unicode{x022EA}}}}
253 \CustomizeMathJax{\newcommand{\nvartriangleright}{\mathrel{\unicode{x022EB}}}}
254
255 \CustomizeMathJax{\newcommand{\vdotsmath}{\mathrel{\unicode{x022EE}}}}
256 \CustomizeMathJax{\newcommand{\unicodecdots}{\mathord{\unicode{x022EF}}}}
257 \CustomizeMathJax{\newcommand{\adots}{\mathrel{\unicode{x022F0}}}}
258 \CustomizeMathJax{\newcommand{\succneq}{\mathrel{\unicode{x02AB2}}}}
259 \CustomizeMathJax{\newcommand{\preceqq}{\mathrel{\unicode{x02AB3}}}}
260 \CustomizeMathJax{\newcommand{\succeqq}{\mathrel{\unicode{x02AB4}}}}
261 \CustomizeMathJax{\newcommand{\precneq}{\mathrel{\unicode{x02AB1}}}}
262
263 \CustomizeMathJax{\newcommand{\mapsfrom}{\mathrel{\unicode{x021A4}}}}
264
265 \CustomizeMathJax{\newcommand{\longmapsfrom}{\mathrel{\unicode{x027FB}}}}
266
267 \CustomizeMathJax{\newcommand{\diameter}{\mathord{\unicode{x02300}}}}
268 \CustomizeMathJax{\newcommand{\coloneq}{\mathrel{\unicode{x02254}}}}
269 \CustomizeMathJax{\newcommand{\eqcolon}{\mathrel{\unicode{x02255}}}}
270 \CustomizeMathJax{\newcommand{\arceq}{\mathrel{\unicode{x02258}}}}
271 \CustomizeMathJax{\newcommand{\wedgeq}{\mathrel{\unicode{x02259}}}}
272 \CustomizeMathJax{\newcommand{\veeeq}{\mathrel{\unicode{x0225A}}}}
273
274 \CustomizeMathJax{\newcommand{\stareq}{\mathrel{\unicode{x0225B}}}}
275 \CustomizeMathJax{\newcommand{\eqdef}{\mathrel{\unicode{x0225D}}}}
276 \CustomizeMathJax{\newcommand{\measeq}{\mathrel{\unicode{x0225E}}}}
277 \CustomizeMathJax{\newcommand{\questeq}{\mathrel{\unicode{x0225F}}}}
278 \CustomizeMathJax{\newcommand{\nequiv}{\mathrel{\unicode{x02262}}}}
279 \CustomizeMathJax{\newcommand{\Equiv}{\mathrel{\unicode{x02263}}}}
280
281 \CustomizeMathJax{\newcommand{\house}{\mathord{\unicode{x02302}}}}
282
283 \CustomizeMathJax{\newcommand{\musicalnote}{\mathord{\unicode{x0266A}}}}
284 \CustomizeMathJax{\newcommand{\degree}{\mathord{\unicode{x000B0}}}}
285 \CustomizeMathJax{\newcommand{\mathsection}{\mathord{\unicode{x000A7}}}}
286 \CustomizeMathJax{\newcommand{\mathparagraph}{\mathord{\unicode{x000B6}}}}
287 \CustomizeMathJax{\newcommand{\checkmarkmath}{\mathord{\unicode{x02713}}}}
288 \CustomizeMathJax{\newcommand{\invnot}{\mathord{\unicode{x02310}}}}
289
290 \CustomizeMathJax{\newcommand{\mathvisiblespace}{\mathord{\unicode{x02423}}}}

```
291 \CustomizeMathJax{\newcommand{\mdlgblksquare}{\mathord{\unicode{x025A0}}}}
292 \CustomizeMathJax{\newcommand{\mdlgwhtsquare}{\mathord{\unicode{x025A1}}}}
293
294 \CustomizeMathJax{\newcommand{\bigblacktriangleup}{\mathord{\unicode{x025B2}}}}
295 \CustomizeMathJax{\newcommand{\varbigtriangleup}{\mathord{\unicode{x025B3}}}}
296
297 \CustomizeMathJax{\newcommand{\bigblacktriangledown}{\mathord{\unicode{x025BC}}}}
298 \CustomizeMathJax{\newcommand{\varbigtriangledown}{\mathord{\unicode{x025BD}}}}
299 \CustomizeMathJax{\newcommand{\Longmapsfrom}{\mathrel{\unicode{x027FD}}}}
300
301 % bug in print font:
302 \CustomizeMathJax{\newcommand{\mdlgblkdiamond}{\mathord{\unicode{x025C6}}}}
303
304 \CustomizeMathJax{\newcommand{\mdlgwhtdiamond}{\mathord{\unicode{x025C7}}}}
305 \CustomizeMathJax{\newcommand{\Longmapsto}{\mathrel{\unicode{x027FE}}}}
306 \CustomizeMathJax{\newcommand{\fisheye}{\mathord{\unicode{x025C9}}}}
307 \CustomizeMathJax{\newcommand{\mdlgwhtlozenge}{\mathord{\unicode{x025CA}}}}
308 \CustomizeMathJax{\newcommand{\mdlgwhtcircle}{\mathbin{\unicode{x025CB}}}}
309 \CustomizeMathJax{\newcommand{\bullseye}{\mathord{\unicode{x025CE}}}}
310 \CustomizeMathJax{\newcommand{\mdlgblkcircle}{\mathord{\unicode{x025CF}}}}
311
312 \CustomizeMathJax{\newcommand{\Nwarrow}{\mathrel{\unicode{x021D6}}}}
313 \CustomizeMathJax{\newcommand{\Nearrow}{\mathrel{\unicode{x021D7}}}}
314 \CustomizeMathJax{\newcommand{\Searrow}{\mathrel{\unicode{x021D8}}}}
315 \CustomizeMathJax{\newcommand{\Swarrow}{\mathrel{\unicode{x021D9}}}}
316
317 \CustomizeMathJax{\newcommand{\Mapsfrom}{\mathord{\unicode{x02906}}}}
318 \CustomizeMathJax{\newcommand{\smwhtcircle}{\mathord{\unicode{x025E6}}}}
319 \CustomizeMathJax{\newcommand{\smwhtdiamond}{\mathbin{\unicode{x022C4}}}}
320 \CustomizeMathJax{\newcommand{\Mapsto}{\mathord{\unicode{x02907}}}}
321
322 \CustomizeMathJax{\let\ngets\nleftarrow}
323 \CustomizeMathJax{\let\nsimeq\nsime}
324 \CustomizeMathJax{\let\nle\nleq}
325 \CustomizeMathJax{\let\nge\ngeq}
326
327 \end{warpMathJax}
```

File 254 **`lwarp-lineno.sty`**

§ 366   Package   # lineno

(*Emulates or patches code by* STEPHAN I. BÖTTCHER.)

lineno (*Pkg*)   lineno is partly emulated, but mostly ignored.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{lineno}[2005/11/02]

2 \newcommand*\resetlinenumber[1][\@ne]{}
3
4 \def\linenumbers{%
5     \@ifnextchar[{\resetlinenumber}%]
6                 {\@ifstar{\resetlinenumber}{}}%
7     }
8
9 \newcommand*{\nolinenumbers}{}
10
```

```
11 \@namedef{linenumbers*}{\par\linenumbers*}
12 \@namedef{runninglinenumbers*}{\par\runninglinenumbers*}
13
14 \def\endlinenumbers{\par}
15 \let\endrunninglinenumbers\endlinenumbers
16 \let\endpagewiselinenumbers\endlinenumbers
17 \expandafter\let\csname endlinenumbers*\endcsname\endlinenumbers
18 \expandafter\let\csname endrunninglinenumbers*\endcsname\endlinenumbers
19 \let\endnolinenumbers\endlinenumbers
20
21 \def\pagewiselinenumbers{\linenumbers\setpagewiselinenumbers}
22
23 \def\runninglinenumbers{\setrunninglinenumbers\linenumbers}
24
25 \def\setpagewiselinenumbers{}
26
27 \def\setrunninglinenumbers{}
28
29 \def\linenomath{}%
30 \@namedef{linenomath*}{}%
31 \def\endlinenomath{}
32 \expandafter\let\csname endlinenomath*\endcsname\endlinenomath
33
34 \let\linelabel\label
35
36 \def\switchlinenumbers{\@ifstar{}{}}
37 \def\setmakelinenumbers#1{\@ifstar{}{}}
38
39 \def\leftlinenumbers{\@ifstar{}{}}
40 \def\rightlinenumbers{\@ifstar{}{}}
41
42 \newcounter{linenumber}
43 \newcount\c@pagewiselinenumber
44 \let\c@runninglinenumber\c@linenumber
45
46 \def\runningpagewiselinenumbers{}
47 \def\realpagewiselinenumbers{}
48
49
50 \NewDocumentCommand\modulolinenumbers{s o}{}
51
52 \chardef\c@linenumbermodulo=5
53 \modulolinenumbers[1]
54
55 \newcommand*\firstlinenumber[1]{}
56
57 \newcommand\internallinenumbers{}
58 \let\endinternallinenumbers\endlinenumbers
59 \@namedef{internallinenumbers*}{\internallinenumbers}
60 \expandafter\let\csname endinternallinenumbers*\endcsname\endlinenumbers
61
62 \newcommand*{\linenoplaceholder}[1]{% redefine per language
63     (line number reference for \detokenize\expandafter{#1})
64 }
65
66 \newcommand*{\lineref}[2][]{\linenoplaceholder{#2}}
67 \newcommand*{\linerefp}[2][]{\linenoplaceholder{#2}}
68 \newcommand*{\linerefr}[2][]{\linenoplaceholder{#2}}
69
70 \newcommand\quotelinenumbers
```

```
71      {\@ifstar\linenumbers{\@ifnextchar[\linenumbers{\linenumbers*}}}
72
73 \newdimen\linenumbersep
74 \newdimen\linenumberwidth
75 \newdimen\quotelinenumbersep
76
77 \quotelinenumbersep=\linenumbersep
78 \let\quotelinenumberfont\linenumberfont
79
80 \def\linenumberfont{\normalfont\tiny\sffamily}
81
82
83 \linenumberwidth=10pt
84 \linenumbersep=10pt
85
86 \def\thelinenumber{}
87
88 \def\LineNumber{}
89 \def\makeLineNumber{}
90 \def\makeLineNumberLeft{}
91 \def\makeLineNumberRight{}
92 \def\makeLineNumberOdd{}
93 \def\makeLineNumberEven{}
94 \def\makeLineNumberRunning{}
95
96
97 \newenvironment{numquote}      {\quote}{\endquote}
98 \newenvironment{numquotation} {\quotation}{\endquotation}
99 \newenvironment{numquote*}     {\quote}{\endquote}
100 \newenvironment{numquotation*}{\quotation}{\endquotation}
101
102 \newdimen\bframerule
103 \bframerule=\fboxrule
104
105 \newdimen\bframesep
106 \bframesep=\fboxsep
107
108 \newenvironment{bframe}
109 {%
110      \LWR@forceminwidth{\bframerule}%
111      \BlockClass[
112          border:\LWR@printlength{\LWR@atleastonept} solid black ; %
113          padding:\LWR@printlength{\bframesep}%
114      ]{bframe}
115 }
116 {\endBlockClass}
```

File 255   **lwarp-lips.sty**

§ 367   Package   **lips**

(*Emulates or patches code by* MATT SWIFT.)

lips (*Pkg*)   lips is emulated.

```
1 % \LWR@ProvidesPackageDrop{lips}
2 \LWR@ProvidesPackageDrop{lips}[2001/08/31]
```

```
3
4 \NewDocumentCommand{\Lips}{}{\textellipsis}
5
6 \NewDocumentCommand{\BracketedLips}{}{[\textellipsis]}
7
8 \let\lips\Lips
9 \let\olips\lips
10
11 \DeclareOption*{}
12 \DeclareOption{mla}{
13 \let\lips\BracketedLips
14 }
15 \ProcessOptions\relax
16
17 \newcommand \LPNobreakList {}
```

File 256   **`lwarp-lipsum.sty`**

§ 368   Package   # lipsum

(*Emulates or patches code by* PATRICK HAPPEL.)

lipsum (*Pkg*)   lipsum is patched for use by lwarp.

for HTML output: 1 `\LWR@ProvidesPackagePass{lipsum}[2021-09-20]`

```
2 \ExplSyntaxOn
3 \xapptocmd{\__lipsum_do:nnnn}
4     {\warpHTMLonly{\LWR@closeparagraph\leavevmode\LWR@orignewline}}
5     {}
6     {\LWR@patcherror{lipsum}{lipsum_do:nnnn}}
7 \ExplSyntaxOff
```

File 257   **`lwarp-listings.sty`**

§ 369   Package   # listings

(*Emulates or patches code by* CARSTEN HEINZ, BROOKS MOSES, JOBST HOFFMANN.)

listings (*Pkg*)   listings is supported with some limitations. Text formatting and escape characters are not yet supported.

1 `\LWR@ProvidesPackagePass{listings}[2024/09/23]`

Force flexible columns. Fixed columns inserts spaces in the PDF output.

2 `\lst@column@flexible`

Patches to embed listings inside pre tags:

```
3 \let\LWR@origlst@Init\lst@Init
4 \let\LWR@origlst@DeInit\lst@DeInit
5
```

```
6 \let\LWR@origlsthkEveryPar\lsthk@EveryPar
7
8 \renewcommand{\l@lstlisting}[2]{\hypertocfloat{1}{lstlisting}{lol}{#1}{#2}}
```

\lstset {⟨*options*⟩}

Use the listings literate option to replace HTML entities:

```
9  \def\lstset@#1{\endgroup%
10 %  \ifx\@empty#1%
11 %       \@empty%
12 %    \else%
13      \setkeys{lst}{%
14         #1%
15         ,literate=%
16         {<}{\HTMLentity{lt}}{4}%
17         {>}{\HTMLentity{gt}}{4}%
18         {'}{\HTMLentity{apos}}{6}%
19         {`}{\HTMLentity{grave}}{7}%
```

The ampersand is not treated here, as the result is inconsistent spacing. It is nevertheless converted to & elsewhere. Sanitizing the double quote interferes with listings' conversion of visible spaces inside strings.

```
20         }%
21 %    \fi%
22 }
```

\lst@Init {⟨*backslash-processing*⟩}   Done at the start of a listing.

```
23 \VerifyCommand[lwarp][listings]{\lst@Init}{A4D103298A6AC8230F525C61F1E1E541}
24
25 \renewcommand{\lst@Init}[1]{%
```

Perform the listings initialization:

```
26 \LWR@traceinfo{lst@Init}%      lwarp
```

\LWR@forcenewpage is moved to the start to avoid a spurious bug with paragraph handling and conditionals.

```
27 \lst@ifdisplaystyle%           lwarp
28     \LWR@forcenewpage%         lwarp
29 \fi%                           lwarp
```

Escapes do not work yet, and are disabled:

```
30 \let\lst@ifmathescape\iffalse%        lwarp
31 \let\lst@DefEsc\relax%                lwarp
32 \def\lst@escapebegin{}%               lwarp
33 \def\lst@escapeend{}%                 lwarp
```

```
34 \renewcommand*{\@captype}{lstlisting}%      lwarp
35     \let\lst@aboveskip\z@\let\lst@belowskip\z@%      lwarp
36     \gdef\lst@boxpos{t}%             lwarp
37     \let\lst@frame\@empty%          lwarp
38     \let\lst@frametshape\@empty%    lwarp
39     \let\lst@framershape\@empty%    lwarp
40     \let\lst@framebshape\@empty%    lwarp
41     \let\lst@framelshape\@empty%    lwarp
42     \lstframe@\lst@frameround ffff\relax%    lwarp
43     \lst@multicols\@empty%  lwarp
44     \begingroup%
```

Inside the listing, temporarily prevent underfull \hbox warnings.

```
45            \hbadness=10000\relax%

46    \ifx\lst@float\relax\else%
47        \edef\@tempa{\noexpand\lst@beginfloat{lstlisting}[\lst@float]}%
48        \expandafter\@tempa%
49    \fi%
50    \ifx\lst@multicols\@empty\else%
51        \edef\lst@next{\noexpand\multicols{\lst@multicols}}%
52        \expandafter\lst@next%
53    \fi%
54    \ifhmode\ifinner \lst@boxtrue \fi\fi%
55    \lst@ifbox%
56        \lsthk@BoxUnsafe%
57        \hbox to\z@\bgroup%
58            $\if t\lst@boxpos \vtop%
59        \else \if b\lst@boxpos \vbox%
60        \else \vcenter \fi\fi%
61        \bgroup \par\noindent%
62    \else%
63        \lst@ifdisplaystyle%
64            \lst@EveryDisplay%
65            \par\penalty-50\relax%
66            \vspace\lst@aboveskip%
67        \fi%
68    \fi%
69    \normalbaselines%
70    \abovecaptionskip\lst@abovecaption\relax%
71    \belowcaptionskip\lst@belowcaption\relax%
72    \lst@MakeCaption t%
```

Use the overall listing label instead of the line number label:

```
73 \LWR@traceinfo{lst@Init: defining current label !\@currentlabel!}%
74    \let\LWR@listings@currentlabel\@currentlabel%              lwarp
75 \LWR@traceinfo{lst@Init: defining current label !\cref@currentlabel!}%
76    \let\LWR@listings@cref@currentlabel\cref@currentlabel%    lwarp

77 \LWR@traceinfo{lst@Init: preinit and init}%
78    \lsthk@PreInit \lsthk@Init%

79    \let\@currentlabel\LWR@listings@currentlabel%              lwarp
80    \let\cref@currentlabel\LWR@listings@cref@currentlabel%    lwarp

81 \LWR@traceinfo{lst@Init: M}%
82    \lst@ifdisplaystyle
83        \global\let\lst@ltxlabel\@empty
84        \if@inlabel
85            \lst@ifresetmargins
86                \leavevmode
87            \else
88                \xdef\lst@ltxlabel{\the\everypar}%
89                \lst@AddTo\lst@ltxlabel{%
90                    \global\let\lst@ltxlabel\@empty
91                    \everypar{\lsthk@EveryLine\lsthk@EveryPar}}%
92            \fi
93        \fi
94        \everypar\expandafter{\lst@ltxlabel
95                        \lsthk@EveryLine\lsthk@EveryPar}%
96    \else
97        \everypar{}%
98        \let\lst@NewLine\@empty
```

```
 99     \fi
100 \LWR@traceinfo{lst@Init: P}%
101     \lsthk@InitVars \lsthk@InitVarsBOL
102     \lst@Let{13}\lst@MProcessListing
103     \let\lst@Backslash#1%
104     \lst@EnterMode{\lst@Pmode}{\lst@SelectCharTable}%
105     \lst@InitFinalize%
106 \LWR@traceinfo{lst@Init: S}%
```

Avoids extra horizontal space:

```
107 \def\lst@framelr{}%      lwarp
```

```
108 \LWR@traceinfo{lst@Init: finished origlst@Init}%
109 \lst@ifdisplaystyle%     lwarp
```

Creating a display.

Disable line numbers, produce the <pre>, then reenable line numbers.

```
110     \LWR@traceinfo{lst@Init: About to create verbatim.}%  lwarp
111     \let\lsthk@EveryPar\relax%                            lwarp
```

```
112     \LWR@atbeginverbatim{programlisting}%        lwarp
113
114     \let\lsthk@EveryPar\LWR@origlsthkEveryPar%   lwarp
115 \else%                                           lwarp
```

Inline, so open a <span>:

```
116     \ifbool{LWR@verbtags}{\LWR@htmltag{%         lwarp
117       span class=\textquotedbl{}inlineprogramlisting\textquotedbl%   lwarp
118     }}{}%                                        lwarp
119 \fi%                                             lwarp
120 \LWR@traceinfo{lst@Init: done}%
121 }
122 \def\LWR@listings@synaxdolloar{$}%  lwarp editor synax highlighting
```

\lst@DeInit   Done at the end of a listing.

```
123 %\renewcommand*{\lst@DeInit}{%
124 \xpretocmd{\lst@DeInit}
125 {%
126 \LWR@traceinfo{lst@DeInit}%
127 \lst@ifdisplaystyle%
```

Creating a display.

Disable line numbers, produce the </pre>, then reenable line numbers:

```
128     \let\lsthk@EveryPar\relax%
129     \LWR@afterendverbatim%
130     \let\lsthk@EveryPar\LWR@origlsthkEveryPar%
131 \else%
```

Inline, so create the closing :

```
132     \ifbool{LWR@verbtags}{\noindent\LWR@htmltag{/span}}{}%
133 \fi%
134 }
135 {}
136 {\LWR@patcherror{listings}{lst@DeInit}}
```

\lst@MakeCaption   {⟨*t/b*⟩}

This is called BOTH at the top and at the bottom of each listing.

Patched for lwarp.

```
137 \VerifyCommand[lwarp][listings]{\lst@MakeCaption}{2B8D898FA33039FBAB7D1772D544153D}
138
139 \def\lst@MakeCaption#1{%
140 \LWR@traceinfo{lst@MakeCaption at #1}%
141   \lst@ifdisplaystyle
142 \LWR@traceinfo{lst@MakeCaption: making a listings display caption}%
143     \ifx #1t%
144 % \allowbreak
145        \ifx\lst@@caption\@empty\expandafter\lst@HRefStepCounter \else
146                              \expandafter\refstepcounter
147      \fi {lstlisting}%
148 % \LWR@traceinfo{About to assign label: !\lst@label!}%
149 %       \ifx\lst@label\@empty\else \label{\lst@label}\fi
150 % \LWR@traceinfo{Finished assigning the label.}%
151        \let\lst@arg\lst@intname \lst@ReplaceIn\lst@arg\lst@filenamerpl
152        \global\let\lst@name\lst@arg \global\let\lstname\lst@name
153        \lst@ifnolol\else
154            \ifx\lst@@caption\@empty
155                \ifx\lst@caption\@empty
156                    \ifx\lst@intname\@empty
157                    \else
158                        \def\lst@temp{ }%
159                        \ifx\lst@intname\lst@temp \else
```

This code places a contents entry for a non-float. This would have to be modified for lwarp:

```
160 \LWR@traceinfo{lst@MakeCaption: addcontents lst@name: -\lst@name-}%
161 %                        \addcontentsline{lol}{lstlisting}{\lst@name}
162                    \fi
163                \fi
164            \fi
165        \else
```

This would have to be modified for lwarp:

```
166 \LWR@traceinfo{lst@MakeCaption: addcontents lst@@caption: -\lst@@caption-}%
167            \addcontentsline{\ext@lstlisting}{lstlisting}%
168 {\protect\numberline{\thelstlisting}%
169 {\protect\ignorespaces \LWR@isolate{\lst@@caption} \protect\relax}}%
170            \fi
171        \fi
172    \fi
173    \ifx\lst@caption\@empty\else
174 \LWR@traceinfo{lst@MakeCaption: lst@caption not empty-}%
175        \lst@IfSubstring #1\lst@captionpos
176            {\begingroup
177 \LWR@traceinfo{lst@MakeCaption: at the selected position}%
```

These space and box commands are not needed for HTML output:

```
178 %            \let\@@vskip\vskip
179 %            \def\vskip{\afterassignment\lst@vskip \@tempskipa}%
180 %            \def\lst@vskip{\nobreak\@@vskip\@tempskipa\nobreak}%
181 %            \par\@parboxrestore\normalsize\normalfont % \noindent (AS)
182 %            \ifx #1t\allowbreak \fi
183            \ifx\lst@title\@empty
```

New lwarp code to create a caption:

```
184                        \LWR@stoppars%        lwarp
185                      \lst@makecaption\fnum@lstlisting{\ignorespaces \lst@caption}
186                  \else
```

New lwarp code to create a title:

```
187 %                      \lst@maketitle\lst@title % (AS)
188 \LWR@traceinfo{lst@MakeCaption: Making title: \lst@title}%
189 \begin{BlockClass}{lstlistingtitle}%    lwarp
190 \lst@maketitle\lst@title%                lwarp
191 \end{BlockClass}%                        lwarp
192          \fi%
193 \LWR@traceinfo{lst@MakeCaption: About to assign label: !\lst@label!}%
194        \ifx\lst@label\@empty\else%      lwarp
195 \leavevmode% gets rid of bad space factor error
196 \GetTitleStringExpand{\lst@caption}%          lwarp
197 \edef\LWR@lntemp{\GetTitleStringResult}%     lwarp
198 \edef\@currentlabelname{\detokenize\expandafter{\LWR@lntemp}}%  lwarp
199 \label{\lst@label}\fi%                               lwarp
200 \LWR@traceinfo{lst@MakeCaption: Finished assigning the label.}% lwarp
```

Not needed for lwarp:

```
201 %                 \ifx #1b\allowbreak \fi
202                  \endgroup}{}%
203     \fi
204 \LWR@traceinfo{lst@MakeCaption: end of making a listings display caption}%
205   \else
206 \LWR@traceinfo{lst@MakeCaption: INLINE}%
207   \fi
208 \LWR@traceinfo{lst@MakeCaption: done at #1}%
209 }
210
211 \renewcommand{\lst@maketitle}[1]{%
212     \LWR@isolate{#1}%
213 }%
214
```

line numbers Patched to keep left line numbers outside of the left margin, and place right line numbers in a field \VerbatimHTMLWidth wide.

```
215 \lst@Key{numbers}{none}{%
216     \let\lst@PlaceNumber\@empty
217     \lstKV@SwitchCases{#1}%
218     {none:\\%
219      left:\def\lst@PlaceNumber{%
```

For now, lwarp places left line numbers inline. Ideally the entire line would be moved to the right, but conflicts with list indenting occurs.

```
220 %          \LWR@origllap{
221             \LWR@orignormalfont%
222             \lst@numberstyle{\thelstnumber}\kern\lst@numbersep%
223 %          }
224     }\\%
225      right:\def\lst@PlaceNumber{\LWR@origrlap{\LWR@orignormalfont
226               \kern 6in \kern\lst@numbersep
227               \lst@numberstyle{\thelstnumber}}}%
228     }{\PackageError{lwarp-listings}{Numbers #1 unknown}\@ehc}}
```

File 258   **lwarp-listliketab.sty**

§ 370   Package   # listliketab

listliketab (*Pkg*)   listliketab is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{listliketab}[2005/01/09]

2 \newcommand*{\storestyleof}[1]{}
3 \newcommand*{\storeliststyle}{}
4 \newenvironment{listliketab}{}{}

File 259   **lwarp-lltjext.sty**

§ 371   Package   # lltjext

(*Emulates or patches code by* The LuaTEX-ja project team.)

lltjext (*Pkg*)   lltjext is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{lltjext}[2018/10/07]

2 \protected\def\yoko{%
3   \directlua{luatexja.direction.set_list_direction(4, 'yoko')}%
4 }
5 \protected\def\tate{\yoko}
6 \protected\def\dtou{\yoko}
7 \protected\def\utod{\yoko}
8
9 \define@key[ltj]{japaram}{direction}{}
10
11 \yoko
12
13 \DeclareExpandableDocumentCommand{\rensuji}{s o m}{#3}
14
15 \DeclareDocumentCommand{\layoutfloat}{d() o m}{}
16
17 \DeclareDocumentCommand{\DeclareLayoutCaption}{m d<> d() o}{}
18
19 \LetLtxMacro\pcaption\caption
20
21 \DeclareDocumentCommand{\layoutcaption}{d<> d() o}{}
22
23 \let\captiondir\relax
24 \RenewDocumentEnvironment{LWR@HTML@minipage}{d<> O{t} O{} O{t} m}
25   {\LWR@HTML@sub@minipage{#2}{#3}{#4}{#5}}
26   {\endLWR@HTML@sub@minipage}
27
28 \RenewDocumentCommand{\LWR@HTML@parbox}{d<> O{t} O{} O{t} m +m}
29 {
30 \LWR@traceinfo{parbox of width #4}%
31 \begin{minipage}[#2][#3][#4]{#5}%

```
32 #6
33 \end{minipage}%
34 }
35
36 \RenewDocumentCommand{\pbox}{d<> O{0pt} O{c} m}{%
37 \global\booltrue{LWR@minipagefullwidth}%
38 \parbox{#2}{#4}%
39 }
```

File 260   **lwarp-lltjp-siunitx.sty**

§ 372   Package   # lltjp-siunitx

(*Emulates or patches code by* THE LUATEX-JA PROJECT TEAM.)

lltjp-siunitx (*Pkg*)   lltjp-siunitx is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{lltjp-siunitx}% 2022-12-14, no date assigned in file

This is the siunitx v3 file, as patched by lltjp-siunitx.

```
 2 \ExplSyntaxOn
 3
 4 \VerifyCommand[lwarp][lltjp-siunitx]{\siunitx_print_text:n}{A248D4314D135CB0AC3E6678F331CFF2}
 5
 6 \cs_set_protected:Npn \siunitx_print_text:n #1
 7   {
 8     \text
 9       {
10         \ltj@allalchar % <--- LuaTeX-ja
11         \bool_if:NT \l__siunitx_print_text_family_bool
12           { \fontfamily { \familydefault } }
13         \bool_if:NT \l__siunitx_print_text_series_bool
14           { \fontseries { \seriesdefault } }
15         \bool_if:NT \l__siunitx_print_text_shape_bool
16           { \fontshape { \shapedefault } }
17         \bool_lazy_any:nT%  lwarp: factors for a single \selectfont
18           {%                                       lwarp
19             { \l__siunitx_print_text_family_bool }% lwarp
20             { \l__siunitx_print_text_series_bool }% lwarp
21             { \l__siunitx_print_text_shape_bool }%  lwarp
22           }%                                       lwarp
23           { \selectfont }%                         lwarp
24         \tl_use:N \l__siunitx_print_text_font_tl%  lwarp
25       \exp_args:NnV \tl_if_head_eq_meaning:nNTF {#1} \l_siunitx_unit_fraction_tl% lwarp
26           {%                                       lwarp
27             \__siunitx_print_text_fraction:Nnn #1%  lwarp
28           }%                                       lwarp
29           {%                                       lwarp
30             \__siunitx_print_text_replace:n {#1}%  ORIGINAL
31           }%                                       lwarp
32       }
33   }
34
35 \ExplSyntaxOff
```

File 261 **lwarp-lltjp-tascmac.sty**

§ 373 Package **lltjp-tascmac**

lltjp-tascmac (*Pkg*) lltjp-tascmac is a patch for tascmac, and is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{lltjp-tascmac}[2020/12/24]

File 262 **lwarp-longtable.sty**

§ 374 Package **longtable**

(*Emulates or patches code by* DAVID CARLISLE.)

longtable (*Pkg*) longtable is emulated.

for HTML output: 1 \LWR@ProvidesPackageDrop{longtable}[2014/10/28]

Use one of either \endhead or \endfirsthead for both print and HTML, and use a
\warpprintonly macro to disable the other head phrase, and also the \endfoot
and \endfirstfoot phrases. (See section 8.10.4 if using threeparttablex.)

```
\begin{longtable}{ [column specifiers] }
[ . . . ] \endfirsthead   % or \endhead, for print and HTML
\warpprintonly{           % not used in HTML
  [ . . . ] \endhead      % or \endfirsthead
  [ . . . ] \endfoot
  [ <lastfoot macros> ] \endlastfoot
}
. . .  table contents . . .
\warpHTMLonly{
  [ <lastfoot macros> ]   % HTML last footer, without \endfoot
                                          % or \endlastfoot.
}
\end{longtable}
```

⚠ Misplaced \noalign Use the \warpprintonly macro instead of the warpprint environment. Doing so
helps avoid "Misplaced \noalign." when using \begin{warpprint}.

⚠ \kill \kill is ignored, place a \kill line inside

\begin{warpprint} . . . \end{warpprint}

or place it inside \warpprintonly.

⚠ lateximage longtable is not supported inside a lateximage.

See:
http://tex.stackexchange.com/questions/43006/
why-is-input-not-expandable

Used to detect more than one of \endhead and \endfirsthead in use for HTML at
the same time.

```
2 \newbool{LWR@longtable@havehead}
3 \boolfalse{LWR@longtable@havehead}
```

longtable (*env.*) * [⟨*horizalignment*⟩] {⟨*colspec*⟩} Emulates the longtable environment.

Per the caption package, the starred version steps the counter per caption. The
unstarred version steps the counter once at the beginning, but not at each caption.

Options [c], [l], and [r] are ignored.

```
 4 \newenvironment{longtable*}[2][]{%
 5     \LWR@floatbegin{table}%
 6     \ifdef{\setcaptiontype}{% caption package:
 7         \setcaptiontype{\LTcaptype}%
 8         \caption@setoptions{longtable}%
 9         \caption@setoptions{@longtable}%
10         \caption@LT@setup%
11     }{% w/o caption package:
12         \renewcommand*{\@captype}{\LTcaptype}%
13     }%
14     \booltrue{LWR@starredlongtable}%
15     \boolfalse{LWR@longtable@havehead}%
16     \let\captionlistentry\LWR@LTcaptionlistentry%
17     \tabular{#2}%
18 }
19 {\endtabular\LWR@floatend}
20
21 \newenvironment{longtable}[2][]{%
22     \LWR@floatbegin{table}%
23     \ifdef{\setcaptiontype}{% caption package:
24         \setcaptiontype{\LTcaptype}%
25         \caption@setoptions{longtable}%
26         \caption@setoptions{@longtable}%
27         \caption@LT@setup%
28     }{% w/o caption package:
29         \renewcommand*{\@captype}{\LTcaptype}%
30     }%
31     \refstepcounter{\LTcaptype}%
32     \boolfalse{LWR@longtable@havehead}%
33     \let\captionlistentry\LWR@LTcaptionlistentry%
34     \tabular{#2}%
35 }
36 {\endtabular\LWR@floatend}
```

Provided for compatibility, but ignored:

```
37 \newcounter{LTchunksize}
```

Error for heads which should have been in \warpprintonly:

```
38 \newcommand*{\LWR@longtable@headerror}{%
39     \PackageError{lwarp-longtable}
40     {For longtable:\MessageBreak
41     1: Keep either one of an \protect\endhead\space or\MessageBreak
42         \space\protect\endfirsthead\space phrase as-is,\MessageBreak
```

```
43        \space to be used by both print and HTML.\MessageBreak
44    2: Place any other \protect\end... phrases inside a\MessageBreak
45        \space\protect\warpprintonly\space macro,
46            to be ignored by HTML.\MessageBreak
47    3: At the end of the table,\MessageBreak
48        \space add a final footer for HTML\MessageBreak
49        \space inside a \protect\warpHTMLonly\space macro.
50            This can be\MessageBreak
51        \space a copy of an \protect\endfoot\space or
52            \protect\endfirstfoot\MessageBreak
53        \space phrase, but without the actual \protect\endfoot\MessageBreak
54        \space or \protect\endfirstfoot\space macros.\MessageBreak
55        \space If using threeparttablex, add\MessageBreak
56        \space \protect\insertTableNotes\space here,
57            optionally with\MessageBreak
58        \space \protect\UseMinipageWidths\space in front.\MessageBreak
59    See the Lwarp documentation regarding\MessageBreak
60    longtables and threeparttablex}
61    {See the Lwarp documentation regading longtables and threeparttablex.}
62 }
```

Error if more than one of \endhead or \endfirsthead is outside of warpprintonly.

```
63 \newcommand*{\LWR@longtable@maybeheaderror}{%
64 \ifbool{LWR@longtable@havehead}%
65    {\LWR@longtable@headerror}%
66    {%
67        \booltrue{LWR@longtable@havehead}
68        \LWR@tabularendofline% throws away options //[dim] and //*
69    }%
70 }
```

Error if more than one of these is outside of warpprint.

```
71 \def\endhead{\LWR@longtable@maybeheaderror}
72 \def\endfirsthead{\LWR@longtable@maybeheaderror}
```

Error if ANY of these is outside of warpprint.

```
73 \def\endfoot{\LWR@longtable@headerror}
74 \def\endlastfoot{\LWR@longtable@headerror}
```

```
75 \let\tabularnewline\\
76 \providecommand*{\LWR@HTML@tabularnewline}{\LWR@tabularendofline}
77 \LWR@formatted{tabularnewline}
```

```
78 \newcommand{\setlongtables}{}% Obsolete command, does nothing.
79 \newlength{\LTleft}
80 \newlength{\LTright}
81 \newlength{\LTpre}
82 \newlength{\LTpost}
83 \newlength{\LTcapwidth}
```

```
84 \LetLtxMacro\LWR@origkill\kill
85 \renewcommand*{\kill}{\LWR@tabularendofline}
86 \appto\LWR@restoreorigformatting{%
87 \LetLtxMacro\kill\LWR@origkill%
```

88 }

File 263 **lwarp-lpic.sty**

§ 375 Package # lpic

(*Emulates or patches code by* R. MATVEYEV.)

lpic (*Pkg*) lpic is patched for use by lwarp.

for HTML output:
```
1 \LWR@ProvidesPackagePass{lpic}[2010/12/23]
```

```
2 \BeforeBeginEnvironment{lpic}{%
3     \begin{lateximage}[-lpic-~\PackageDiagramAltText]?%
4 }
5
6 \AfterEndEnvironment{lpic}{\end{lateximage}}
```

File 264 **lwarp-lscape.sty**

§ 376 Package # lscape

(*Emulates or patches code by* D. P. CARLISLE.)

lscape (*Pkg*) lscape is ignored.

for HTML output: Discard all options for lwarp-lscape.

```
1 \LWR@ProvidesPackageDrop{lscape}[2000/10/22]
```

```
2 \newenvironment*{landscape}{}{}
```

File 265 **lwarp-ltablex.sty**

§ 377 Package # ltablex

(*Emulates or patches code by* ANIL K. GOEL.)

ltablex (*Pkg*) ltablex is emulated by lwarp.

for HTML output: Relies on tabularx.

```
1 \RequirePackage{longtable}
2 \RequirePackage{tabularx}
3
4 \LWR@ProvidesPackageDrop{ltablex}[2014/08/13]
5
6 \DeclareDocumentEnvironment{tabularx}{m o m}
7 {\longtable{#3}}
8 {\endlongtable}
9
10 \DeclareDocumentEnvironment{tabularx*}{m o m}
```

```
11 {\longtable{#3}}
12 {\endlongtable}
13
14 \newcommand*{\keepXColumns}{}
15 \newcommand*{\convertXColumns}{}
```

File 266   **`lwarp-ltcaption.sty`**

§ 378   Package   # ltcaption

(*Emulates or patches code by* AXEL SOMMERFELDT.)

ltcaption (*Pkg*)   ltcaption is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{ltcaption}[2018/08/26]

\LTcaptype is already defined by lwarp.

longtable* is already defined by lwarp-longtable.

```
2 \newlength{\LTcapskip}
3 \newlength{\LTcapleft}
4 \newlength{\LTcapright}
5 \newcommand*{\LTcapmarginsfalse}{}
```

File 267   **`lwarp-ltxgrid.sty`**

§ 379   Package   # ltxgrid

ltxgrid (*Pkg*)   ltxgrid is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{ltxgrid}[2010/07/25]

```
2 \newcommand*{\onecolumngrid}{}
3 \newcommand*{\twocolumngrid}{}
4 \newcommand*{\removestuff}{}
5 \newcommand*{\addstuff}[2]{}
6 \newcommand*{\replacestuff}[2]{}
```

File 268   **`lwarp-ltxtable.sty`**

§ 380   Package   # ltxtable

ltxtable (*Pkg*)   ltxtable is emulated.

⚠   table numbering   The print version does not seem to honor longtable* from the caption package, while lwarp does.

for HTML output: 1 \RequirePackage{tabularx,longtable}
                       2 \LWR@ProvidesPackageDrop{ltxtable}[1995/12/11]

\LTXtable   {⟨*width*⟩} {⟨*file*⟩}

```
3 \newcommand*{\LTXtable}[2]{%
4     \input{#2}%
5 }
```

File 269   `lwarp-lua-check-hyphen.sty`

§ 381   Package   lua-check-hyphen

lua-check-hyphen (*Pkg*)   lua-check-hyphen is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{lua-check-hyphen}[2018/04/19]`

`2 \newcommand*{\LuaCheckHyphen}[1]{}`

File 270   `lwarp-lua-visual-debug.sty`

§ 382   Package   lua-visual-debug

lua-visual-debug (*Pkg*)   lua-visual-debug is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{lua-visual-debug}[2016/05/30]`

File 271   `lwarp-luacolor.sty`

§ 383   Package   luacolor

luacolor (*Pkg*)   luacolor is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{luacolor}[2016/05/16]`

`2 \newcommand{\luacolorProcessBox}[1]{}`

File 272   `lwarp-luamplib.sty`

§ 384   Package   luamplib

(*Emulates or patches code by* HANS HAGEN, TACO HOEKWATER, ELIE ROUX, PHILIPP GESANG, KIM DO-HYUN.)

luamplib (*Pkg*)   luamplib is patched for use by lwarp.

for HTML output: `1 \LWR@ProvidesPackagePass{luamplib}[2020/02/24]`

```
2 \BeforeBeginEnvironment{mplibcode}{%
3     \begin{lateximage}[-mplibcode-~\PackageDiagramAltText]?%
4 }
5 \AfterEndEnvironment{mplibcode}{\end{lateximage}}
```

File 273   **lwarp-luatexko.sty**

§ 385   Package   # luatexko

(*Emulates or patches code by* Dohyun Kim, Soojin Nam.)

luatexko (*Pkg*)   luatexko is patched for use by lwarp.

Modern HTML is used for \dotemph, \ruby, and offset and thickness control for \uline, etc.

for HTML output: 1 \LWR@ProvidesPackagePass{luatexko}[2021/07/10]

```
2 \protected\def\typesetvertical{}
3 \protected\def\typesethorizontal{}
4
5 \def\verticaltypesetting{\BlockClass{verticalrl}}
6 \def\beginverticaltypesetting{\BlockClass{verticalrl}}
7 \def\endverticaltypesetting{\endBlockClass}
8
9 \protected\def\vertical#1{\BlockClass{verticalrl}}
10 \protected\def\endvertical{\endBlockClass}
11 \protected\def\horizontal#1{\BlockClass{horizontaltb}}
12 \protected\def\endhorizontal{\endBlockClass}
13 \DeclareDocumentCommand{\vertlatin}{m}{#1}

14 \newcommand{\LWR@HTML@dotemph}[1]{%
15 %       \uline{#1}%
16     \InlineClass[text-emphasis-style: dot]{dotemph}{#1}%
17 }
18 \LWR@formatted{dotemph}


19 \newcommand{\LWR@HTML@ruby}[2]{%
20     \LWR@htmltagc{ruby}%
21     #1%
22     \LWR@htmltagc{rp}(\LWR@htmltagc{/rp}%
23     \LWR@htmltagc{rt}#2\LWR@htmltagc{/rt}%
24     \LWR@htmltagc{rp})\LWR@htmltagc{/rp}%
25     \LWR@htmltagc{/ruby}%
26 }
27 \LWR@formatted{ruby}
```

The following is modified from lwarp-ulem:

```
28 \NewDocumentCommand{\LWR@HTML@uline}{+m}{%
29     \InlineClass%
30         (text-decoration:underline; text-decoration-skip: auto)%
31         [%
32             text-underline-offset: \ulinedown ;
33             text-decoration-thickness: \ulinewidth%
34         ]%
```

```
35         {uline}{\LWR@isolate{#1}}%
36 }
37 \LWR@formatted{uline}
38
39 \NewDocumentCommand{\LWR@HTML@uuline}{+m}{%
40     \InlineClass%
41         (%
42             text-decoration:underline; text-decoration-skip: auto;%
43             text-decoration-style:double%
44         )%
45         [%
46             text-underline-offset: \ulinedown ;
47             text-decoration-thickness: \ulinewidth%
48         ]%
49         {uuline}{\LWR@isolate{#1}}%
50 }
51 \LWR@formatted{uuline}
52
53 \NewDocumentCommand{\LWR@HTML@uwave}{+m}{%
54     \InlineClass%
55         (%
56             text-decoration:underline; text-decoration-skip: auto;%
57             text-decoration-style:wavy%
58         )%
59         [%
60             text-underline-offset: \ulinedown ;
61             text-decoration-thickness: \ulinewidth%
62         ]%
63         {uwave}{\LWR@isolate{#1}}%
64 }
65 \LWR@formatted{uwave}
66
67 \NewDocumentCommand{\LWR@HTML@sout}{+m}{%
68     \InlineClass%
69         (text-decoration:line-through)%
70         [text-decoration-thickness: \ulinewidth]%
71         {sout}{\LWR@isolate{#1}}%
72 }
73 \LWR@formatted{sout}
74
75 \NewDocumentCommand{\LWR@HTML@xout}{+m}{%
76     \InlineClass%
77         (text-decoration:line-through)%
78         [text-decoration-thickness: \ulinewidth]%
79         {xout}{\LWR@isolate{#1}}%
80 }
81 \LWR@formatted{xout}
82
83 \NewDocumentCommand{\LWR@HTML@dashuline}{+m}{%
84     \InlineClass%
85         (%
86             text-decoration:underline;%
87             text-decoration-skip: auto;%
88             text-decoration-style:dashed%
89         )%
90         [%
91             text-underline-offset: \ulinedown ;
92             text-decoration-thickness: \ulinewidth%
93         ]%
94         {dashuline}{\LWR@isolate{#1}}%
```

```
 95 }
 96 \LWR@formatted{dashuline}
 97
 98 \NewDocumentCommand{\LWR@HTML@dotuline}{+m}{%
 99     \InlineClass%
100        (%
101             text-decoration:underline;%
102             text-decoration-skip: auto;%
103             text-decoration-style: dotted%
104        )%
105        [%
106             text-underline-offset: \ulinedown ;
107             text-decoration-thickness: \ulinewidth%
108        ]%
109        {dotuline}{\LWR@isolate{#1}}%
110 }
111 \LWR@formatted{dotuline}
```

File 274   `lwarp-luatodonotes.sty`

§ 386   Package   **luatodonotes**

(*Emulates or patches code by* FABIAN LIPP.)

luatodonotes (*Pkg*)   luatodonotes is emulated.

The documentation for todonotes and luatodonotes have an example with a todo inside a caption. If this example does not work it will be necessary to move the todo outside of the caption.

for HTML output: 1 \LWR@ProvidesPackagePass{luatodonotes}[2017/09/30]

Nullify options:

```
2 \@todonotes@additionalMarginEnabledfalse

3 \if@todonotes@disabled
4 \else
5
6 \newcommand{\ext@todo}{tdo}
7
8 \renewcommand{\l@todo}[2]{\hypertocfloat{1}{todo}{ldo}{#1}{#2}}

9 \let\LWRTODONOTES@orig@todototoc\todototoc
10
11 \renewcommand*{\todototoc}{%
12 \LWR@phantomsection%
13 \LWRTODONOTES@orig@todototoc%
14 }
15
16
17 \renewcommand{\@todonotes@drawMarginNoteWithLine}{%
18 \fcolorbox
19     {\@todonotes@currentbordercolor}
20     {\@todonotes@currentbackgroundcolor}
21     {\arabic{@todonotes@numberoftodonotes}}}
```

```
22 \marginpar{\@todonotes@drawMarginNote}
23 }
24
25 \renewcommand{\@todonotes@drawInlineNote}{%
26 \fcolorboxBlock%
27     {\@todonotes@currentbordercolor}%
28     {\@todonotes@currentbackgroundcolor}%
29     {%
30         \if@todonotes@authorgiven%
31         {\@todonotes@author:\,}%
32         \fi%
33         \@todonotes@text%
34     }%
35 }
36
37 \newcommand{\@todonotes@drawMarginNote}{%
38     \if@todonotes@authorgiven%
39         \@todonotes@author\par%
40     \fi%
41     \arabic{@todonotes@numberoftodonotes}: %
42     \fcolorbox%
43     {\@todonotes@currentbordercolor}%
44     {\@todonotes@currentbackgroundcolor}%
45     {%
46         \@todonotes@sizecommand%
47         \@todonotes@text %
48     }%
49 }%
50
51 \renewcommand{\missingfigure}[2][]{%
52 \setkeys{todonotes}{#1}%
53 \addcontentsline{tdo}{todo}{\@todonotes@MissingFigureText: #2}%
54 \fcolorboxBlock%
55     {\@todonotes@currentbordercolor}%
56     {\@todonotes@currentfigcolor}%
57     {%
58         \setlength{\fboxrule}{4pt}%
59         \fcolorbox{red}{white}{Missing figure} \quad #2%
60     }
61 }
62
63 \LetLtxMacro\LWRTODONOTES@orig@todocommon\@todocommon
64
65 \RenewDocumentCommand{\@todocommon}{m m}{%
66 \begingroup%
67 \renewcommand*{\phantomsection}{}%
68 \LWRTODONOTES@orig@todocommon{#1}{#2}%
69 \endgroup%
70 }
71
72 \VerifyCommand[lwarp][luatodonotes]{\@todoarea}{3D40C9C729633DA7BB80F7A27E7C2694}
73
74 \renewcommand{\@todoarea}[3][]{%
75     \@todonotes@areaselectedtrue%
76     \@todocommon{#1}{#2}%
77     \todonotes@textmark@highlight{#3}%
78     \zref@label{@todonotes@\arabic{@todonotes@numberoftodonotes}@end}%
79 }%
80
81
```

```
82 \DeclareDocumentCommand{\todonotes@textmark@highlight}{m}{%
83 \InlineClass[background:\LWR@origpound{}B3FFB3]{highlight}{#1}%
84 }
85
86 \fi% \if@todonotes@disabled
```

File 275   **lwarp-luavlna.sty**

§ 387   Package   **luavlna**

(*Emulates or patches code by* MICHAL HOFTICH, MIRO HRONČOK.)

luavlna (*Pkg*)   luavlna is patched for use by lwarp.

The package is disabled for HTML output, due to incompatibilities with lwarp's handling of math SVG images.

for HTML output: 1 \LWR@ProvidesPackagePass{luavlna}[2019/10/30]

```
2 \preventsingleoff
3 \LetLtxMacro\preventsingleon\preventsingleoff
```

File 276   **lwarp-lyluatex.sty**

§ 388   Package   **lyluatex**

(*Emulates or patches code by* FR. JACQUES PERON, URS LISKA, BR. SAMUEL SPRINGUEL.)

lyluatex (*Pkg*)   lyluatex is patched for use by lwarp.

For the first compile, to set *lwarpmk*'s configuration, use:

    lualatex --shell-escape <filename>

⚠ images   After compiling the document with **lwarpmk html**, use **lwarpmk limages** to convert the Lilypond images for HTML.

The option insert=systems results in an image per system. Each music image "system" is placed inside a <span> of class lyluatex, which defaults to display: inline-block.

⚠ insert=fullpage   The option insert=fullpage results in a single image per page of printed output. Each music "fullpage" image is placed inside a <div> of class lyluatex. To match the number of measures per line with the printed version, use the geometry package to select the page geometry, or use the lyluatex options for page and staff sizes.

⚠ options   To use \linewidth or \textwidth inside the package options for lyluatex, use the kvoptions-patch package first:

    \usepackage{kvoptions-patch}
    \usepackage[...,line-width-0.8\linewidth,...]{lyluatex}

⚠ raw-pdf   If using raw-pdf, the resulting PDF images must be converted to SVG:

Enter ⇒   **lwarpmk pdftosvg tmp-ly/*.pdf**

for HTML output:
```
1 \LWR@origRequirePackage{luacode}
2
3 \LWR@ProvidesPackagePass{lyluatex}[2023/04/18]
```

User-redefinable ALT tag:

```
4 \newcommand*{\LyluatexImageAltText}{-lilypond-~\PackageDiagramAltText}
```

`\ly@compilescore`

{⟨*Lilypond object*⟩}

```
5 \VerifyCommand[lwarp][lyluatex]{\ly@compilescore}{31A1EF1F24F22143AFD302A7C6AD29E6}
6
7 \renewcommand*{\ly@compilescore}[1]{%
```

A local group holds a number of changes:

```
8     \begingroup%
```

The user's original geometry and font size are restored to match the print version. This allows for correct spacing in the musical score.

```
9     \LWR@maybe@orignewpage%
10    \LWR@origloadgeometry{LWR@usergeometry}%
11    \LWR@print@normalsize%
```

A local group holds a redefined \includegraphics which is used by *lyluatex.lua* to insert the *Lilypond* score if insert=systems is used. This is now placed inside a lateximage, which itself is placed inside a <span> of class lyluatex.

\LWR@addbaselinemarker preserves the left margins.

```
12    \renewcommand{\includegraphics}[2][]{%
13        \InlineClass{lyluatex}{%
14            \begin{lateximage}[\LyluatexImageAltText]?%
15            \LWR@addbaselinemarker%
16            \LWR@origincludegraphics{##2}%
17            \end{lateximage}%
18        }%
19    }%
```

From the original:

```
20    \ly@setunits%
21    \setluaoption{ly}{currfiledir}{\currfiledir}
22    \setluaoption{ly}{twoside}{\ly@istwosided}
23    \directlua{
24        #1
25        ly.newpage_if_fullpage()
26    }%
27    \ly@resetunits%
28    \ly@currentfonts%
```

The fullpage version is set inside an HTML <div>:

```
29    \directlua{
30        if (ly.score.insert == 'fullpage') then
31            tex.print{[[\string\begin{BlockClass}{lyluatex}]]}
32        end
33    }%
```

Generate the score:

```
34     \directlua{ly.score:process()}%
```

Close the <div>:

```
35     \directlua{
36         if (ly.score.insert == 'fullpage') then
37             tex.print{[[\string\end{BlockClass}]]}
38         end
39     }%
```

Move to a new page and renew the regular page geometry:

```
40     \LWR@maybe@orignewpage%
41     \LWR@origrestoregeometry%
```

End of the local group.

```
42     \endgroup%
43 }
```

In HTML the following generates an error, so is removed:

```
44 \VerifyCommand[lwarp][lyluatex]{\endly@bufferenv}{9B3BA3FC990E03634B57041666E0048D}
45
46 \xpatchcmd{\endly@bufferenv}
47     {\hspace{0pt}\\}
48     {}
49     {}
50     {\LWR@patcherror{lyluatex}{endly@bufferenv}}
```

File 277 **`lwarp-magaz.sty`**

§ 389   Package **magaz**

magaz (*Pkg*)   magaz is emulated.

for HTML output: 1 \LWR@ProvidesPackageDrop{magaz}[2011/11/24]

```
2 \newcommand\FirstLine[1]{%
3     \begingroup%
4     \FirstLineFont{%
5         \LWR@textcurrentcolor{%
6             \LWR@textcurrentfont{%
7                 #1%
8             }%
9         }%
10    }%
11    \endgroup%
12 }
13
14 \providecommand\FirstLineFont{\scshape}
```

File 278   **`lwarp-makeidx.sty`**

§ 390   Package **makeidx**

(*Emulates or patches code by* LATEX PROJECT TEAM.)

makeidx (*Pkg*)   makeidx is patched for use by lwarp.

   1 \LWR@ProvidesPackagePass{makeidx}[2014/09/29]

\@wrindex is redefined \AtBeginDocument by the lwarp core.

\printindex

```
2 \preto\printindex{%
3     \LWR@maybe@orignewpage%
4     \LWR@startpars%
5 }
```

File 279   **`lwarp-manyfoot.sty`**

§ 391   Package   **manyfoot**

manyfoot (*Pkg*)   manyfoot is emulated.

bigfoot, manyfoot   Verbatim footnotes are not yet supported.
⚠   verbatim

If using the bigfoot package, and possibly also manyfoot, problems may occur with counter allocation because lwarp uses many counters, and there is a difference in how counters numbered 256 and up are handled in PDF LATEX. With bigfoot this has been known to show up as an error related to one footnote insert being forbidden inside another. Another problem showed up as a input stack error, and which of these problems occurred depended on how many counters were allocated.

As a possible solution, try creating several new counters before defining bigfoot or manyfoot footnotes, hoping to shift the problematic counter above the 256 threshold. It may instead be necessary to use XƎLATEX or LuaLATEX instead of PDF LATEX.

lwarp's emulation of bigfoot uses manyfoot, so some of the bigfoot enhancements are included here.

The bigfoot "default" footnote is ignored, using the lwarp version instead.

   1 \LWR@ProvidesPackageDrop{manyfoot}[2005/09/11]

```
2 \RequirePackage{nccfoots}
3
4 \newcommand{\extrafootnoterule}{}
5
6 \let\defaultfootnoterule\footnoterule
7
8 \newcommand*{\SelectFootnoteRule}[2][0]{}
9
10 \newcommand{\footnoterulepriority}{1}
11
12 \newcommand{\SetFootnoteHook}[1]{}
13 \@onlypreamble\SetFootnoteHook
14
15 \newcommand{\SplitNote}{}
```

```
16
17 \newcommand*\ExtraParaSkip[1]{}
18
19 \newcommand*{\newfootnote}[2][plain]{%
20     \ifstrequal{#2}{default}{}{% not "default"
21         \expandafter\newbox\csname LWR@footnote#2box\endcsname%
22         \appto{\LWR@printpendingfootnotes}{%
23             \LWR@@printpendingfootnotes{footnote#2}%
24         }
25         \long\csdef{Footnotetext#2}##1##2{%
26             \NCC@makefnmark{##1}%
27             \LWR@@footnotetext{##2}{LWR@footnote#2box}%
28         }%
29         \long\csdef{Footnotetext#2+}##1##2{%
30             \NCC@makefnmark{##1}%
31             \LWR@@footnotetext{##2}{LWR@footnote#2box}%
32         }%
33     }% not "default"
34 }
35 \@onlypreamble\newfootnote
36
37 \newcommand*{\DeclareNewFootnote}[2][plain]{%
38   \@ifnextchar[%
39         {\LWR@manyfoot@declare{#1}{#2}}%
40         {\LWR@manyfoot@declare{#1}{#2}[arabic]}%
41 }
42
43 \def\LWR@manyfoot@declare#1#2[#3]{%
44 \ifstrequal{#2}{default}{}{% not "default"
45   \newfootnote[#1]{#2}%
46   \newcounter{footnote#2}%
47     \newcounter{footnote#2Reset}%
48     \setcounter{footnote#2Reset}{0}%
49     \csdef{thefootnote#2}{%
50       \expandafter\noexpand\csname @#3\endcsname%
51       \expandafter\noexpand\csname c@footnote#2\endcsname%
52     }%
```

For bigfoot, the footnote commands may be appended with one or two plusses or one or two minuses, which are ignored in HTML.

```
53     \expandafter\NewDocumentCommand\csname footnote#2\endcsname{t{+}t{+}t{-}t{-}}{%
54         \stepcounter{footnote#2}%
55         \protected@xdef\@thefnmark{\csname thefootnote#2\endcsname}%
56         \@footnotemark%
57         \csuse{Footnotetext#2}{\@thefnmark}% absorbs the footnote contents
58     }%
59     \csdef{footnotemark#2}{%
60         \stepcounter{footnote#2}%
61         \protected@xdef\@thefnmark{\csname thefootnote#2\endcsname}%
62         \@footnotemark%
63     }%
64     \expandafter\NewDocumentCommand\csname footnotetext#2\endcsname{t{+}t{+}t{-}t{-}}{%
65         \protected@xdef\@thefnmark{\csname thefootnote#2\endcsname}%
66         \csuse{Footnotetext#2}{\@thefnmark}% absorbs the footnote contents
67     }%
68     \csdef{Footnotemark#2}{%
69         \Footnotemark%
70     }%
71     \csdef{Footnote#2}##1{%
```

```
72        \Footnotemark{##1}%
73        \csuse{Footnotetext#2}{##1}%
74    }%
75 }% not "default"
76 }
77 \@onlypreamble\DeclareNewFootnote
```

File 280   **lwarp-marginal.sty**

§ 392   Package   # marginal

marginal (*Pkg*)   marginal is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{marginal}

```
2 \newcommand*{\showlostmarginals}{}
3 \newcommand*{\enlargefreelist}{}
4 \newcommand*{\onesidemarginals}{}
```

File 281   **lwarp-marginfit.sty**

§ 393   Package   # marginfit

marginfit (*Pkg*)   marginfit is ignored.

for HTML output: Discard all options for lwarp-marginfit:

1 \LWR@ProvidesPackageDrop{marginfit}[2018/06/08]

File 282   **lwarp-marginfix.sty**

§ 394   Package   # marginfix

(*Emulates or patches code by* STEPHEN HICKS.)

marginfix (*Pkg*)   marginfix is ignored.

for HTML output: Discard all options for lwarp-marginfix:

1 \LWR@ProvidesPackageDrop{marginfix}[2013/09/08]

```
2 \newcommand*{\marginskip}[1]{}
3 \newcommand*{\clearmargin}{}
4 \newcommand*{\softclearmargin}{}
5 \newcommand*{\extendmargin}[1]{}
6 \newcommand*{\mparshift}[1]{}
7 \newdimen\marginheightadjustment
8 \newdimen\marginposadjustment
9 \newcommand*{\blockmargin}[1][]{}
10 \newcommand*{\unblockmargin}[1][]{}
11 \newcommand*{\marginphantom}[2][]{}
```

File 283   `lwarp-marginnote.sty`

§ 395   Package   # marginnote

(*Emulates or patches code by* Markus Kohm.)

marginnote (*Pkg*)   marginnote is emulated.

for HTML output: Discard all options for lwarp-marginnote:

```
1 \LWR@ProvidesPackageDrop{marginnote}[2018/08/09]
```

```
2 \NewDocumentCommand{\marginnote}{+o +m o}{\marginpar{#2}}
```

```
3 \newcommand*{\marginnoteleftadjust}{}
4 \newcommand*{\marginnoterightadjust}{}
5 \newcommand*{\marginnotetextwidth}{}
6 \let\marginnotetextwidth\textwidth
7 \newcommand*{\marginnotevadjust}{}
8 \newcommand*{\marginfont}{}
9 \newcommand*{\raggedleftmarginnote}{}
10 \newcommand*{\raggedrightmarginnote}{}
```

```
11 \appto\LWR@restoreorigformatting{%
12    \RenewDocumentCommand{\marginnote}{+o +m o}{}
13 }
```

For MathJax:

```
14 \begin{warpMathJax}
15 \CustomizeMathJax{\newcommand{\LWRmarginnote}[1][]{}}
16 \CustomizeMathJax{\newcommand{\marginnote}[2][]{\qquad{\small\textrm{#2}}\LWRmarginnote}}
17 \end{warpMathJax}
```

File 284   `lwarp-marvosym.sty`

§ 396   Package   # marvosym

(*Emulates or patches code by* Thomas Henlich, Mojca Miklavec.)

marvosym (*Pkg*)   marvosym is patched for use by lwarp.

Hashed inline images are used, as there may not be Unicode support for all icons.

for HTML output: ```
1 \LWR@ProvidesPackagePass{marvosym}[2011/07/20]
```

```
2 \renewcommand{\mvchr}[1]{%
3    \begin{lateximage}*[symbol #1]?[marvosym #1]%
4    \mvs\char#1%
5    \end{lateximage}%
6 }
```

```
 7
 8 \renewcommand{\textmvs}[1]{%
 9     \begin{lateximage}%
10     \mvs #1%
11     \end{lateximage}%
12 }
```

File 285 **`lwarp-mathalpha.sty`**

§ 397   Package   # mathalpha

(*Emulates or patches code by* MICHAEL SHARPE.)

mathalpha (*Pkg*)   mathalpha is used as-is for SVG math, and is emulated for MATHJAX.

⚠ limitations   The MATHJAX emulation ignores all package options, and some bold fonts may not be not supported by MATHJAX.

for HTML output:
```
 1 \LWR@ProvidesPackagePass{mathalpha}[2021/11/18]
 2
 3 \begin{warpMathJax}
 4 \CustomizeMathJax{\newcommand{\mathbfbb}[1]{\boldsymbol{\mathbb{#1}}}}% not bold
 5 \CustomizeMathJax{\newcommand{\mathbfcal}[1]{\boldsymbol{\mathcal{#1}}}}
 6 \CustomizeMathJax{\newcommand{\mathbffrak}[1]{\boldsymbol{\mathfrak{#1}}}}
 7 \CustomizeMathJax{\newcommand{\mathbfscr}[1]{\boldsymbol{\mathscr{#1}}}}% not bold
 8
 9 \IfPackageLoadedWithOptionsTF{mathalpha}{oldbold}
10 {
11 \CustomizeMathJax{\newcommand{\mathbbb}[1]{\boldsymbol{\mathbb{#1}}}}% not bold
12 \CustomizeMathJax{\newcommand{\mathbcal}[1]{\boldsymbol{\mathcal{#1}}}}
13 \CustomizeMathJax{\newcommand{\mathbfrak}[1]{\boldsymbol{\mathfrak{#1}}}}
14 \CustomizeMathJax{\newcommand{\mathbscr}[1]{\boldsymbol{\mathscr{#1}}}}% not bold
15 }{}
16 \end{warpMathJax}
```

File 286 **`lwarp-mathastext.sty`**

§ 398   Package   # mathastext

(*Emulates or patches code by* JEAN-FRANÇOIS BURNOL.)

mathastext (*Pkg*)   mathastext is used as-is for SVG math, and emulated for MATHJAX.

for HTML output:
```
 1 \LWR@ProvidesPackagePass{mathastext}[2019/11/16]
 2 \LWR@origRequirePackage{lwarp-common-mathjax-letters}
 3
 4 \begin{warpMathJax}
 5 \ifmst@itgreek
 6 %     \LWR@mathjax@addgreek@l@it{}{}
 7 \else
 8     \ifmst@upgreek
 9         \LWR@mathjax@addgreek@l@up{}{}
10     \else
```

```
11          \ifmst@frenchmath
12              \LWR@mathjax@addgreek@l@up{}{}
13          \else
14              \ifmst@italic
15 %              \LWR@mathjax@addgreek@l@it{}{}
16              \else
17                  \LWR@mathjax@addgreek@l@up{}{}
18              \fi
19          \fi
20      \fi
21 \fi
22
23 \ifcase\mst@greek@select
24      \or{\LWR@mathjax@addgreek@u@it*{}{}}
25 %      \or{\LWR@mathjax@addgreek@u@up*{}{}}
26 \fi
27
28 \CustomizeMathJax{\newcommand{\mathnormalbold}[1]{\boldsymbol{#1}}}
29 \CustomizeMathJax{\newcommand{\MathEulerBold}[1]{\boldsymbol{#1}}}
30 \CustomizeMathJax{\newcommand{\MathEuler}[1]{{#1}}}
31 \CustomizeMathJax{\newcommand{\MathPSymbol}[1]{{#1}}}
32 \CustomizeMathJax{\let\fouriervec\vec}
33 \CustomizeMathJax{\let\pmvec\vec}
34 \CustomizeMathJax{\let\inodot\imath}
35 \CustomizeMathJax{\let\jnodot\jmath}
36 \CustomizeMathJax{\let\shortiff\iff}
37 \CustomizeMathJax{\let\longto\longrightarrow}
38 \CustomizeMathJax{\newcommand{\inftypsy}{\mathord{\unicode{x221E}}}}
39 \CustomizeMathJax{\newcommand{\proptopsy}{\mathrel{\unicode{x221D}}}}
40 \CustomizeMathJax{\let\prodpsy\prod}
41 \CustomizeMathJax{\let\sumpsy\sum}
42 \CustomizeMathJax{\let\MToriginalprod\prod}
43 \CustomizeMathJax{\let\MToriginalsum\sum}
44 \CustomizeMathJax{\newcommand{\DotTriangle}{\mathord{\unicode{x2234}}}}
45 \end{warpMathJax}
```

File 287 **`lwarp-mathcomp.sty`**

§ 399 Package # mathcomp

(*Emulates or patches code by* TILMANN BÖß.)

mathcomp (*Pkg*) mathcomp is supported as-is for SVG math, and is emulated for MATHJAX.

for HTML output:
```
1 \LWR@ProvidesPackagePass{mathcomp}[2001/01/07]

2 \begin{warpMathJax}
3 \CustomizeMathJax{\newcommand{\tcohm}{\mathrm{\Omega}}}
4 \CustomizeMathJax{\newcommand{\tccelsius}{\unicode{x2103}}}
5 \CustomizeMathJax{\newcommand{\tcmu}{\mathrm{\unicode{x00B5}}}}
6 \CustomizeMathJax{\newcommand{\tcperthousand}{\unicode{x2030}}}
7 \CustomizeMathJax{\newcommand{\tcpertenthousand}{\unicode{x2031}}}
8 \CustomizeMathJax{\newcommand{\tcdegree}{\mathrm{^\circ}}}
9 \CustomizeMathJax{\newcommand{\tcdigitoldstyle}[1]{\oldstyle{#1}}}
10 \end{warpMathJax}
```

File 288   `lwarp-mathdesign.sty`

§ 400   Package   # mathdesign

(*Emulates or patches code by* PAUL PICHAUREAU.)

mathdesign (*Pkg*)   **mathdesign** is used as-is for SVG math, and is emulated for MATHJAX.

⚠   limitations   The MATHJAX emulation ignores all package options except `greekuppercase` and `greeklowercase`. The dedicated macros for upright and italic greek letters work correctly, although the user may wish to swap the definitions for epsilon and phi.

SVG math should appear the same as the printed output.

for HTML output: 1 `\LWR@ProvidesPackagePass{mathdesign}[2013/08/29]`

For MATHJAX:

```
2 \LWR@origRequirePackage{lwarp-common-mathjax-letters}
3
4 \LWR@origRequirePackage{lwarp-common-mathjax-overlaysymbols}
5
6 \begin{warpMathJax}
7 \LWR@infoprocessingmathjax{mathdesign}
```

Default greek upright or italicized:

```
8 \if@MD@grupright
9 \LWR@mathjax@addgreek@l@up{}{}
10 \fi
11
12 \if@MD@GRupright
13 \else
14 \LWR@mathjax@addgreek@u@it*{}{}
15 \fi
```

Upright:

```
16 \LWR@mathjax@addgreek@l@up{}{up}
17 \LWR@mathjax@addgreek@u@up*{}{up}
```

Italicized:

```
18 \LWR@mathjax@addgreek@l@it{}{it}
19 \LWR@mathjax@addgreek@u@it*{}{it}
```

Adapt to **mathdesign** inconsistency:

```
20 \CustomizeMathJax{\let\digammaup\Digammaup}
21 \CustomizeMathJax{\renewcommand{\digammait}{\mathit{\digammaup}}}
```

Extra symbols:

```
22 \CustomizeMathJax{\newcommand{\smallin}{\mathrel{\unicode{x220A}}}}
```

23 \CustomizeMathJax{\newcommand{\smallowns}{\mathrel{\unicode{x220D}}}}
24 \CustomizeMathJax{\newcommand{\notsmallin}{\mathrel{\LWRoverlaysymbols{/}{\unicode{x220A}}}}}
25 \CustomizeMathJax{\newcommand{\notsmallowns}{\mathrel{\LWRoverlaysymbols{/}{\unicode{x220D}}}}}
26 \CustomizeMathJax{\newcommand{\rightangle}{\mathord{\unicode{x221F}}}}

Integrals:

27 \CustomizeMathJax{\newcommand{\intclockwise}{\mathop{\unicode{x2231}}\limits}}
28 \CustomizeMathJax{\newcommand{\ointclockwise}{\mathop{\unicode{x2232}}\limits}}
29 \CustomizeMathJax{\newcommand{\ointctrclockwise}{\mathop{\unicode{x2233}}\limits}}
30 \CustomizeMathJax{\newcommand{\oiint}{\mathop{\unicode{x222F}}\limits}}
31 \CustomizeMathJax{\newcommand{\oiiint}{\mathop{\unicode{x2230}}\limits}}

Math and text mode:

32 \CustomizeMathJax{\newcommand{\ddag}{\unicode{x2021}}}
33 \CustomizeMathJax{\newcommand{\P}{\unicode{x00B6}}}
34 \CustomizeMathJax{\newcommand{\copyright}{\unicode{x00A9}}}
35 \CustomizeMathJax{\newcommand{\dag}{\unicode{x2020}}}
36 \CustomizeMathJax{\newcommand{\pounds}{\unicode{x00A3}}}

Extra symbols:

37 \CustomizeMathJax{\newcommand{\iddots}{\mathinner{\unicode{x22F0}}}}
38 \CustomizeMathJax{\newcommand{\utimes}{\mathbin{\overline{\times}}}}
39 \CustomizeMathJax{\newcommand{\dtimes}{\mathbin{\underline{\times}}}}
40 \CustomizeMathJax{\newcommand{\udtimes}{\mathbin{\overline{\underline{\times}}}}}
41 \CustomizeMathJax{\newcommand{\leftwave}{\left\{}}
42 \CustomizeMathJax{\newcommand{\rightwave}{\right\}}}
43
44 \end{warpMathJax}

File 289   **lwarp-mathdots.sty**

§ 401   Package   # mathdots

(*Emulates or patches code by* DAN LUECKING.)

mathdots (*Pkg*)   mathdots is used as-is for svg math, and emulated for MATHJAX.

for HTML output: 1 \LWR@ProvidesPackagePass{mathdots}[2014/06/11]

2 \begin{warpMathJax}
3 \CustomizeMathJax{\newcommand{\iddots}{\mathinner{\unicode{x22F0}}}}
4 \CustomizeMathJax{\let\fixedddots\ddots}
5 \CustomizeMathJax{\let\fixedvdots\vdots}
6 \CustomizeMathJax{\let\fixediddots\iddots}
7 \CustomizeMathJax{\let\originalddots\ddots}
8 \CustomizeMathJax{\let\originalvdots\vdots}
9 \CustomizeMathJax{\let\originaliddots\iddots}
10 \CustomizeMathJax{\let\originaldddot\dddot}
11 \CustomizeMathJax{\let\originaldddddot\ddddot}
12 \end{warpMathJax}

File 290   `lwarp-mathfixs.sty`

§ 402   Package   **mathfixs**

(*Emulates or patches code by* NIKLAS BEISERT.)

mathfixs (*Pkg*)   mathfixs is used as-is for SVG math, and is emulated for MATHJAX.

⚠   Greek letters are unchanged.

for HTML output: 1 `\LWR@ProvidesPackagePass{mathfixs}[2018/12/30]`

```
2 \begin{warpMathJax}
3 \CustomizeMathJax{\newcommand{\rfrac}[2]{\tfrac{#1}{#2}}}
4 \CustomizeMathJax{\newcommand{\vfrac}[2]{\mathinner{{}^{#1}\!/{}_{#2}}}}
5 \CustomizeMathJax{\newcommand{\ProvideMathFix}[1]{}}
6 \CustomizeMathJax{\newcommand{\mathbold}[1]{\boldsymbol{#1}}}
7 \CustomizeMathJax{\newcommand{\.}{\,}}
8 \end{warpMathJax}
```

File 291   `lwarp-mathpazo.sty`

§ 403   Package   **mathpazo**

(*Emulates or patches code by* WALTER SCHMIDT.)

mathpazo (*Pkg*)   mathpazo is used as-is for SVG math, and is emulated for MATHJAX.

⚠   limitations   The MATHJAX emulation ignores all package options. The dedicated macros for upright greek letters do work correctly.

SVG math should appear the same as the printed output.

for HTML output: 1 `\LWR@ProvidesPackagePass{mathpazo}[2020/03/25]`

For MATHJAX:

```
2 \LWR@origRequirePackage{lwarp-common-mathjax-letters}
3
4 \begin{warpMathJax}
5 \LWR@infoprocessingmathjax{mathpazo}
6
7 \ifpazo@slGreek
8 \LWR@mathjax@addgreek@u@it*{}{}
9 \fi
10
11 \LWR@mathjax@addgreek@u@up*{up}{}
12
13 \CustomizeMathJax{\newcommand{\mathbold}[1]{\boldsymbol{#1}}}
14 \end{warpMathJax}
```

File 292   `lwarp-mathptmx.sty`

§ 404   Package   # mathptmx

(*Emulates or patches code by* WALTER SCHMIDT.)

mathptmx (*Pkg*)   mathptmx is used as-is for SVG math, and is emulated for MATHJAX.

⚠ limitations   The MATHJAX emulation ignores all package options. The dedicated macros for upright greek letters do work correctly.

SVG math should appear the same as the printed output.

for HTML output: 1 \LWR@ProvidesPackagePass{mathptmx}[2020/03/25]

For MATHJAX:

```
2 \LWR@origRequirePackage{lwarp-common-mathjax-letters}
3
4 \begin{warpMathJax}
5 \LWR@infoprocessingmathjax{mathptmx}
6
7 \IfPackageLoadedWithOptionsTF{mathptmx}{slantedGreek}
8     {\LWR@mathjax@addgreek@u@it*{}{}}
9     {}
10
11 \LWR@mathjax@addgreek@u@up*{up}{}
12 \end{warpMathJax}
```

File 293   `lwarp-mathspec.sty`

§ 405   Package   # mathspec

(*Emulates or patches code by* ANDREW GILBERT MOSCHOU.)

mathspec (*Pkg*)   mathspec is used as-is with SVG math, and is emulated for MATHJAX.

⚠ quotes   Double quotes (\" and the " character) are removed during MATHJAX emulation, but this also includes inside \text.

for HTML output: 1 \LWR@ProvidesPackagePass{mathspec}[2016/12/22]

```
2 \LWR@origRequirePackage{lwarp-common-mathjax-letters}
3
4 \begin{warpMathJax}
```

Neutralize double quotes (" and \"):

```
5 \booltrue{LWR@MathJax@silentquotes}
```

Sort options for out Greek emulation:

```
 6 \AtBeginDocument{
 7 \ifcase\eu@GreekUppercase@@value %% If Greek Uppercase Regular
 8     \LWR@mathjax@addgreek@u@up*{}{}
 9 \or %% If Greek Uppercase Italic
10     \LWR@mathjax@addgreek@u@it*{}{}
11 \or %% If Greek Uppercase Plain
12     \LWR@mathjax@addgreek@u@up*{}{}
13 \fi
14 \ifcase\eu@GreekLowercase@@value %% If Greek Lowercase Regular
15     \LWR@mathjax@addgreek@l@up{}{}
16 \or %% If Greek Lowercase Italic
17     \LWR@mathjax@addgreek@l@it{}{}
18 \or %% If Greek Lowercase Plain
19     \LWR@mathjax@addgreek@l@it{}{}
20 \fi
21 }
```

Swap definitions according the mathspec conditionals:

```
22 \newcommand*{\LWR@mathspec@varforms}{%
23 \eu@ifbooltrue{GreekLowercase}{
24     \eu@ifbooltrue{exchangebetaforms}{
25         \CustomizeMathJax{\let\LWRorigbeta\beta}
26         \CustomizeMathJax{\let\beta\varbeta}
27         \CustomizeMathJax{\let\varbeta\LWRorigbeta}
28     }
29     \eu@ifbooltrue{exchangeepsilonforms}{
30         \CustomizeMathJax{\let\LWRorigepsilon\epsilon}
31         \CustomizeMathJax{\let\epsilon\varepsilon}
32         \CustomizeMathJax{\let\varepsilon\LWRorigepsilon}
33     }
34     \eu@ifbooltrue{exchangethetaforms}{
35         \CustomizeMathJax{\let\LWRorigtheta\theta}
36         \CustomizeMathJax{\let\theta\vartheta}
37         \CustomizeMathJax{\let\vartheta\LWRorigtheta}
38     }
39     \eu@ifbooltrue{exchangekappaforms}{
40         \CustomizeMathJax{\let\LWRorigkappa\kappa}
41         \CustomizeMathJax{\let\kappa\varkappa}
42         \CustomizeMathJax{\let\varkappa\LWRorigkappa}
43     }
44     \eu@ifbooltrue{exchangepiforms}{
45         \CustomizeMathJax{\let\LWRorigpi\pi}
46         \CustomizeMathJax{\let\pi\varpi}
47         \CustomizeMathJax{\let\varpi\LWRorigpi}
48     }
49     \eu@ifbooltrue{exchangerhoforms}{
50         \CustomizeMathJax{\let\LWRorigrho\rho}
51         \CustomizeMathJax{\let\rho\varrho}
52         \CustomizeMathJax{\let\varrho\LWRorigrho}
53     }
54     \eu@ifbooltrue{exchangephiforms}{
55         \CustomizeMathJax{\let\LWRorigphi\phi}
56         \CustomizeMathJax{\let\phi\varphi}
57         \CustomizeMathJax{\let\varphi\LWRorigphi}
58     }
59 }
60 \eu@ifbooltrue{GreekUppercase}{
61     \eu@ifbooltrue{exhangeThetaforms}{
62         \CustomizeMathJax{\let\LWRorigTheta\Theta}
```

```
63        \CustomizeMathJax{\let\Theta\varTheta}
64        \CustomizeMathJax{\let\varTheta\LWRorigTheta}
65    }
66 }
67 }
```

Append new action to **mathspec**'s \AtBeginDocument code:

```
68 \xapptocmd{\exchangeforms}
69    {\AtBeginDocument{\LWR@mathspec@varforms}}
70    {}
71    {\LWR@patcherror{mathspec}{exchangeforms}}
72
73 \end{warpMathJax}
```

File 294   **lwarp-mathtools.sty**

§ 406   Package   # mathtools

(*Emulates or patches code by* Morten Høgholm, Lars Madsen, LaTeX3 project.)

mathtools (*Pkg*)   **mathtools** is patched for use by **lwarp**. Emulation macros are provided for Math-
⚠ equation numbering   Jax.

showonlyrefs is disabled, as it conflicts with **cleveref**, which is used by **lwarp**.
Equation numbers may not match the print version.

⚠ italic correction   mathic is not emulated for HTML.

⚠ MathJax   If using MathJax:

- Recent changes may not yet be updated in the MathJax extension, which is used by **lwarp**.

- **mathtools** disallowspaces does not work for MathJax. Protect brackets which are not optional arguments, such as:
    ```
    \begin{gathered}{}
    [p]=1 . . .
    \end{gathered}
    ```

- showonlyrefs does not work in MathJax, and will result in a difference in equation numbering compared to the print version.

- alignat in MathJax requires math mode, but in LaTeX it doesn't. It may be required to use warpHTML and warpprint to isolate a version for each mode.

- \DeclarePairedDelimiter and related must be in the preamble before \begin{document}.

- \MakeAboxedCommand uses \Aboxed for all commands made.

for HTML output: 1 \LWR@ProvidesPackagePass{mathtools}[2024/10/04]

2 \RequirePackage{graphicx}

3 \MHInternalSyntaxOn

Forces showonlyrefs off because lwarp uses cleveref, which is not compatible with showonlyrefs.

```
4 \renewcommand*\MT_showonlyrefs_true:{%
5     \PackageWarningNoLine{lwarp}
6     {%
7         Mathtools \space showonlyrefs \space conflicts \space
8         with \space cleveref,\MessageBreak
9         which \space is \space used \space by \space lwarp, \space
10        so \space showonlyrefs \space is\MessageBreak
11        forced \space off. \space\space
12        Equation \space numbers \space may \space not \space match%
13    }
14    \MT_showonlyrefs_false:
15 }
16 \mathtoolsset{showonlyrefs=false}
```

Forces math italic correction off. Not patched for lwarp.

```
17 \renewcommand*{\MT_mathic_true:}{\MT_mathic_false:}
18 \mathtoolsset{mathic=false}
```

```
19 \MHInternalSyntaxOff
```

For MATHJAX.

The MATHJAX package is used, and improvements are added.

```
20 \begin{warpMathJax}
21 \CustomizeMathJax{\require{mathtools}}
22
23 \LWR@infoprocessingmathjax{mathtools}
24
25 \CustomizeMathJax{\newcommand{\vcentcolon}{\mathrel{\unicode{x2236}}}}
26
27 \CustomizeMathJax{\newcommand{\approxcolon}{\approx\vcentcolon}}
28 \CustomizeMathJax{\newcommand{\Approxcolon}{\approx\dblcolon}}
29
30 \CustomizeMathJax{\newcommand{\simcolon}{\sim\vcentcolon}}
31 \CustomizeMathJax{\newcommand{\Simcolon}{\sim\dblcolon}}
32
33 \CustomizeMathJax{\newcommand{\dashcolon}{\mathrel{-}\vcentcolon}}
34 \CustomizeMathJax{\newcommand{\Dashcolon}{\mathrel{-}\dblcolon}}
35
36 \CustomizeMathJax{\newcommand{\colondash}{\vcentcolon\mathrel{-}}}
37 \CustomizeMathJax{\newcommand{\Colondash}{\dblcolon\mathrel{-}}}
38
39 \CustomizeMathJax{\newenvironment{crampedsubarray}[1]{}{}}
40
41 \CustomizeMathJax{\newcommand{\smashoperator}[2][]{#2\limits}}
42
43 \CustomizeMathJax{\newcommand{\SwapAboveDisplaySkip}{}}
44
45 \CustomizeMathJax{\newcommand{\LaTeXunderbrace}[1]{\underbrace{#1}}}
46 \CustomizeMathJax{\newcommand{\LaTeXoverbrace}[1]{\overbrace{#1}}}
47
48 \CustomizeMathJax{\Newextarrow\xLongleftarrow{10,10}{0x21D0}}
49 \CustomizeMathJax{\Newextarrow\xLongrightarrow{10,10}{0x21D2}}
50 \CustomizeMathJax{\let\xlongleftarrow\xleftarrow}
```

```
51 \CustomizeMathJax{\let\xlongrightarrow\xrightarrow}
52
53 \CustomizeMathJax{\newcommand{\LWRmultlined}[1][]{\begin{multline*}}}
54 \CustomizeMathJax{\newenvironment{multlined}[1][]{\LWRmultlined}{\end{multline*}}}
55
56 \CustomizeMathJax{\let\LWRorigshoveleft\shoveleft}
57 \CustomizeMathJax{\renewcommand{\shoveleft}[1][]{\LWRorigshoveleft}}
58 \CustomizeMathJax{\let\LWRorigshoveright\shoveright}
59 \CustomizeMathJax{\renewcommand{\shoveright}[1][]{\LWRorigshoveright}}
60
61 \CustomizeMathJax{\newcommand{\shortintertext}[1]{\text{#1}\notag \\}}
62
63 \LetLtxMacro\LWR@mathtools@orig@DeclarePairedDelimiter\DeclarePairedDelimiter
64 \renewcommand{\DeclarePairedDelimiter}[3]{
65     \LWR@mathtools@orig@DeclarePairedDelimiter{#1}{#2}{#3
66 % starred:
67     \appto\LWR@customizedMathJax{\LWRbackslash(}
68     \appto\LWR@customizedMathJax{%
69      \LWRbackslash{}newcommand\{\LWRbackslash\macrotocsname{#1}LWRsubstar\}%
70     }%
71     \appto\LWR@customizedMathJax{[2][]}%
72     \appto\LWR@customizedMathJax{\{\{}%
73     \LWR@subcustomizedmathjax{##1\left#2##2##1\right#3}%
74     \appto\LWR@customizedMathJax{\}\}}%
75     \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
76 % not starred:
77     \appto\LWR@customizedMathJax{\LWRbackslash(}
78     \appto\LWR@customizedMathJax{%
79      \LWRbackslash{}newcommand\{\LWRbackslash\macrotocsname{#1}LWRsubnostar\}%
80     }%
81     \appto\LWR@customizedMathJax{[2][]}%
82     \appto\LWR@customizedMathJax{\{\{}%
83     \LWR@subcustomizedmathjax{##1#2##2##1#3}%
84     \appto\LWR@customizedMathJax{\}\}}%
85     \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
86 % user macro:
87     \appto\LWR@customizedMathJax{\LWRbackslash(}
88     \appto\LWR@customizedMathJax{%
89         \LWRbackslash{}newcommand\{\LWRbackslash{}\macrotocsname{#1}\}%
90         \{\LWRbackslash{}ifstar%
91             \LWRbackslash{}\macrotocsname{#1}LWRsubstar%
92             \LWRbackslash{}\macrotocsname{#1}LWRsubnostar%
93         \}%
94     }%
95     \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
96 }
97 \@onlypreamble\DeclareParedDelimiter
98
99 % (DeclarePairedDelimiterX is already defined to use \DeclarePairedDelimiterXPP.)
100
101 \LetLtxMacro\LWR@mathtools@orig@DeclarePairedDelimiterXPP\DeclarePairedDelimiterXPP
102 \DeclareDocumentCommand{\DeclarePairedDelimiterXPP}{m O{1} m m m m m}{
103   \LWR@mathtools@orig@DeclarePairedDelimiterXPP{#1}[#2]{#3}{#4}{#5}{#6}{#7}
104 % subsubstar, second opt arg
105     \appto\LWR@customizedMathJax{\LWRbackslash(}%
106     \appto\LWR@customizedMathJax{%
107      \LWRbackslash{}newcommand\{\LWRbackslash\macrotocsname{#1}LWRsubsubstar\}%
108     }%
109     \appto\LWR@customizedMathJax{[#2]}%
110     \appto\LWR@customizedMathJax{\{\{\LWRbackslash{}left}%
```

```
111      \LWR@subcustomizedmathjax{#3#4#7}%
112      \appto\LWR@customizedMathJax{\LWRbackslash{}right}%
113      \LWR@subcustomizedmathjax{#5#6}%
114      \appto\LWR@customizedMathJax{\}\}}%
115      \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
116 % substar, first opt arg
117      \appto\LWR@customizedMathJax{\LWRbackslash(}%
118      \appto\LWR@customizedMathJax{%
119       \LWRbackslash{}newcommand\{\LWRbackslash\macrotocsname{#1}LWRsubstar\}[1][]%
120      }%
121      \appto\LWR@customizedMathJax{%
122          \{
123          \LWRbackslash{}def\LWRbackslash{}delimsize\{\#1\}
124          \LWRbackslash\macrotocsname{#1}LWRsubsubstar
125          \}%
126      }%
127      \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
128 % subsubnostar, second opt arg
129      \appto\LWR@customizedMathJax{\LWRbackslash(}%
130      \appto\LWR@customizedMathJax{%
131       \LWRbackslash{}newcommand\{\LWRbackslash\macrotocsname{#1}LWRsubsubnostar\}%
132      }%
133      \appto\LWR@customizedMathJax{[#2]}%
134      \appto\LWR@customizedMathJax{\{\{\LWRbackslash{}delimsize}%
135      \LWR@subcustomizedmathjax{#3#4#7}%
136      \appto\LWR@customizedMathJax{\LWRbackslash{}delimsize}%
137      \LWR@subcustomizedmathjax{#5#6}%
138      \appto\LWR@customizedMathJax{\}\}}%
139      \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
140 % subnostar, first opt arg
141      \appto\LWR@customizedMathJax{\LWRbackslash(}%
142      \appto\LWR@customizedMathJax{%
143       \LWRbackslash{}newcommand\{\LWRbackslash\macrotocsname{#1}LWRsubnostar\}[1][]%
144      }%
145      \appto\LWR@customizedMathJax{%
146          \{
147          \LWRbackslash{}def\LWRbackslash{}delimsize\{\#1\}
148          \LWRbackslash\macrotocsname{#1}LWRsubsubnostar
149          \}%
150      }%
151      \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
152 % user macro:
153      \appto\LWR@customizedMathJax{\LWRbackslash(}
154      \appto\LWR@customizedMathJax{%
155          \LWRbackslash{}newcommand\{%
156              \LWRbackslash{}\macrotocsname{#1}%
157          \}%
158              \{\LWRbackslash{}ifstar%
159                  \LWRbackslash{}\macrotocsname{#1}LWRsubstar%
160                  \LWRbackslash{}\macrotocsname{#1}LWRsubnostar%
161          \}%
162      }%
163      \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
164 }
165 \@onlypreamble\DeclareParedDelimiterXPP
166 \@onlypreamble\DeclareParedDelimiterX
167
168 \LetLtxMacro\LWR@mathtools@orig@newgathered\newgathered
169 \renewcommand{\newgathered}[4]{%
170      \LWR@mathtools@orig@newgathered{#1}{#2}{#3}{#4}%
```

```
171        \appto\LWR@customizedMathJax{\LWRbackslash(}%
172        \LWR@subcustomizedmathjax{%
173            \newenvironment{#1}{\begin{gathered}}{\end{gathered}}%
174        }%
175        \appto\LWR@customizedMathJax{\LWRbackslash)\LWR@orignewline}%
176 }
177 \@onlypreamble\newgathered
178
179
180 \LetLtxMacro\LWR@mathtools@MakeAboxedCommand\MakeAboxedCommand
181 \renewcommand{\MakeAboxedCommand}[2]{%
182        \LWR@mathtools@MakeAboxedCommand{#1}{#2}%
183        \appto\LWR@customizedMathJax{\LWRbackslash(}
184        \appto\LWR@customizedMathJax{%
185            \LWRbackslash{}let%
186                \LWRbackslash\macrotocsname{#1}%
187                \LWRbackslash{}Aboxed%
188        }%
189        \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
190 }
191
192 \end{warpMathJax}
```

File 295   **`lwarp-mattens.sty`**

§ 407   Package   # mattens

(*Emulates or patches code by* Danie Els.)

mattens (*Pkg*)   mattens is used as-is for svg math, and is emulated for MathJax.

for HTML output: 1 \LWR@ProvidesPackagePass{mattens}[2010/03/26]

```
2 \begin{warpMathJax}
3 \CustomizeMathJax{\newcommand{\LWRmattensnull}{}}
4
5 \CustomizeMathJax{\newcommand{\LWRmattensnostar}[2][]{%
6    {#1{\LWRmattensundercmd{\LWRmattensovercmd{\LWRmattenscross{\boldsymbol{#2}}}}}}%
7 }}
8
9 \CustomizeMathJax{\newcommand{\LWRmattensstar}[2][]{%
10    {#1{\LWRmattensundercmd{\LWRmattensovercmd{\LWRmattenscross{#2}}}}}%
11 }}
12
13 \CustomizeMathJax{\newcommand{\LWRmattens}{
14    \ifstar\LWRmattensstar\LWRmattensnostar%
15 }}
16
17 \CustomizeMathJax{\newcommand{\aS}{%
18    \let\LWRmattenscross\LWRmattensnull%
19    \let\LWRmattensovercmd\overrightarrow%
20    \let\LWRmattensundercmd\LWRmattensnull%
21    \LWRmattens%
22 }}
23
24 \CustomizeMathJax{\newcommand{\Sa}{%
25    \let\LWRmattenscross\LWRmattensnull%
```

```
26      \let\LWRmattensovercmd\underrightarrow%
27      \let\LWRmattensundercmd\LWRmattensnull%
28      \LWRmattens%
29 }}
30
31 \CustomizeMathJax{\newcommand{\bS}{%
32      \let\LWRmattenscross\LWRmattensnull%
33      \let\LWRmattensovercmd\overline%
34      \let\LWRmattensundercmd\LWRmattensnull%
35      \LWRmattens%
36 }}
37
38 \CustomizeMathJax{\newcommand{\Sb}{%
39      \let\LWRmattenscross\LWRmattensnull%
40      \let\LWRmattensovercmd\underline%
41      \let\LWRmattensundercmd\LWRmattensnull%
42      \LWRmattens%
43 }}
44
45 \CustomizeMathJax{\newcommand{\aSa}{%
46      \let\LWRmattenscross\LWRmattensnull%
47      \let\LWRmattensovercmd\overrightarrow%
48      \let\LWRmattensundercmd\underrightarrow%
49      \LWRmattens%
50 }}
51
52 \CustomizeMathJax{\newcommand{\aSb}{%
53      \let\LWRmattenscross\LWRmattensnull%
54      \let\LWRmattensovercmd\overrightarrow%
55      \let\LWRmattensundercmd\underline%
56      \LWRmattens%
57 }}
58
59 \CustomizeMathJax{\newcommand{\bSa}{%
60      \let\LWRmattenscross\LWRmattensnull%
61      \let\LWRmattensovercmd\overline%
62      \let\LWRmattensundercmd\underrightarrow%
63      \LWRmattens%
64 }}
65
66 \CustomizeMathJax{\newcommand{\bSb}{%
67      \let\LWRmattenscross\LWRmattensnull%
68      \let\LWRmattensovercmd\overline%
69      \let\LWRmattensundercmd\underline%
70      \LWRmattens%
71 }}
72
73 \CustomizeMathJax{\newcommand{\aCSa}{%
74      \let\LWRmattenscross\tilde%
75      \let\LWRmattensovercmd\overrightarrow%
76      \let\LWRmattensundercmd\underrightarrow%
77      \LWRmattens%
78 }}
79
80 \CustomizeMathJax{\newcommand{\bCSb}{%
81      \let\LWRmattenscross\tilde%
82      \let\LWRmattensovercmd\overline%
83      \let\LWRmattensundercmd\underline%
84      \LWRmattens%
85 }}
```

86 \end{warpMathJax}

File 296 **`lwarp-maybemath.sty`**

§ 408   Package   # maybemath

(*Emulates or patches code by* ANDY BUCKLEY.)

maybemath (*Pkg*)   maybemath is used as-is for SVG math, and is emulated for MATHJAX.

⚠   no effect   MATHJAX is not able to detect the surrounding text font, so all maybemath macros are ignored.

for HTML output: 1 \LWR@ProvidesPackagePass{maybemath}[2005/2/22]

2 \begin{warpMathJax}
3 \CustomizeMathJax{\newcommand{\mayberm}[1]{{#1}}}
4 \CustomizeMathJax{\let\maybebm\mayberm}
5 \CustomizeMathJax{\let\maybeit\mayberm}
6 \CustomizeMathJax{\let\maybeitrm\mayberm}
7 \CustomizeMathJax{\let\maybeitsubscript\mayberm}
8 \CustomizeMathJax{\let\maybesf\mayberm}
9 \CustomizeMathJax{\let\maybebmsf\mayberm}
10 \end{warpMathJax}

File 297 **`lwarp-mcaption.sty`**

§ 409   Package   # mcaption

(*Emulates or patches code by* STEPHAN HENNIG.)

mcaption (*Pkg*)   mcaption is ignored.

for HTML output: Discard all options for lwarp-mcaption:

1 \LWR@ProvidesPackageDrop{mcaption}[2009/03/13]

2 \newenvironment{margincap}{}{}
3 \newcommand*{\margincapalign}{}
4 \newlength{\margincapsep}

File 298 **`lwarp-mdframed.sty`**

§ 410   Package   # mdframed

(*Emulates or patches code by* MARCO DANIEL, ELKE SCHUBERT.)

mdframed (*Pkg*)   mdframed is loaded with options forced to framemethod=none.

§ 410.1 **Limitations**

support Most basic functionality is supported, including frame background colors and single-border colors and thickness, title and subtitle background colors and borders and thickness, border radius, and shadow. CSS classes are created for mdframed environments and frame titles.

⚠ loading When used, lwarp loads mdframed in HTML with framemethod=none.

font For title font, use

  frametitlefont=\textbf,

instead of

  frametitlefont=\bfseries,

where \textbf must appear just before the comma and will receive the following text as its argument (since the text happens to be between braces in the mdframed source). Since lwarp does not support \bfseries and friends, only one font selection may be made at a time.

theoremtitlefont theoremtitlefont is not supported, since the following text is not in braces in the mdframed source.

ignored options userdefinedwidth and align are currently ignored.

css classes Environments created or encapsulated by mdframed are enclosed in a <div> of class mdframed, and also class md<environmentname> for new environments.

Frame titles are placed in a <div> of class |mdframedtitle|. Subtitles are in a <div> of class |mdframedsubtitle|, and likewise for subsubtitles.

Pre-existing hooks are used to patch extra functions before and after the frames.

§ 410.2 **Package loading**

for HTML output:
```
1 \RequirePackage{xcolor}% for \convertcolorspec
2
3 \LWR@ProvidesPackageDrop{mdframed}[2013/07/01]
```

Do not require TikZ or pstricks:

```
4 \LWR@origRequirePackage[framemethod=none]{mdframed}
```

§ 410.3 **Patches**

Patch to remove PDF formatting and add HTML tags:

```
5 \AtBeginDocument{
6 \def\mdf@trivlist#1{%
7   \edef\mdf@temp{%
8 %     \topsep=\the\topsep\relax%
9 %     \partopsep=\the\partopsep\relax%
10 %     \parsep=\the\parsep\relax%
11   }%
12 %   \setlength{\topsep}{#1}%
13 %   \topskip\z@%
14 %   \partopsep\z@%
15 %   \parsep\z@%
```

```
16 %    \@nmbrlistfalse%
17 %    \@trivlist%
18 %    \labelwidth\z@%
19 %    \leftmargin\z@%
20 %    \itemindent\z@%
21    \let\@itemlabel\@empty%
22    \def\makelabel##1{##1}%
23 %    \item\relax\mdf@temp\relax%
24 }
25
26 \renewcommand*{\endmdf@trivlist}{%
27 \LWR@traceinfo{endmdf@trivlist}%
28 % \endtrivlist%
29 \LWR@listend%
30 }
31 }% AtBeginDocument
```

§ 410.4 Initial setup

To handle css and paragraphs, patch code at start and end of environment and
contents. \LWR@print@raggedright helps avoid hyphenation.

```
32 \mdfsetup{
33 startcode={\LWR@mdframedstart\LWR@print@raggedright},
34 endcode={\LWR@mdframedend},
35 startinnercode={\LWR@startpars\LWR@print@raggedright},
36 endinnercode={\LWR@stoppars},
37 }
```

§ 410.5 Color and length HTML conversion

\LWR@mdfprintcolor   {⟨*mdfcolorkey*⟩}

Given the mdframed key, print the color.

```
38 \newcommand*{\LWR@mdfprintcolor}[1]{%
39 \convertcolorspec{named}{\@nameuse{mdf@#1}}{HTML}\LWR@tempcolor%
40 \LWR@origpound\LWR@tempcolor
41 }
```

\LWR@mdfprintlength   {⟨*mdflengthkey*⟩}

Given the mdframed key, print the length.

```
42 \newcommand*{\LWR@mdfprintlength}[1]{%
43 \LWR@forceminwidth{\@nameuse{mdf@#1@length}}%
44 \LWR@printlength{\LWR@atleastonept}%
45 }
```

§ 410.6 Environment encapsulation

\LWR@mdframedstart   Actions before an mdframe starts.

Encapsulate a frame inside a <div> of the desired class.

```
46 \newcommand*{\LWR@mdframedstart}{%
47 \LWR@traceinfo{LWR@mdframedstart start}%
```

Warn if starting a frame inside a <span>:

```
48       \LWR@spanwarninvalid{mdframe}%
```

Turn off paragraph handling during the generation of the encapsulating tags:

```
49 \LWR@stoppars%
```

Open a <div> and with custom `class` and custom `style`. A BlockClass environment is not used because this <div> is created by the **mdframed** `startcode` and `endcode` settings, which do not properly nest the <div> inside the **mdframed** environment.

```
50 \LWR@htmltagc{div class=\textquotedbl%
51 mdframed%
52 \ifdefstring{\LWR@mdthisenv}{mdframed}{}{ \LWR@mdthisenv}%
53 \textquotedbl \LWR@orignewline
54 style=\textquotedbl\LWR@orignewline
```

Convert and print the background color:

```
55 background: \LWR@mdfprintcolor{backgroundcolor} ; \LWR@orignewline
```

Convert and print the border color and width:

```
56 border: \LWR@mdfprintlength{linewidth} solid
57 \LWR@mdfprintcolor{linecolor} ; \LWR@orignewline
```

Convert and print the border radius:

```
58 border-radius: \LWR@mdfprintlength{roundcorner} ; \LWR@orignewline
```

Convert and print the shadow:

```
59 \ifbool{mdf@shadow}{%
60     box-shadow:
61     \LWR@mdfprintlength{shadowsize}
62     \LWR@mdfprintlength{shadowsize}
63     \LWR@mdfprintlength{shadowsize}
64     \LWR@mdfprintcolor{shadowcolor} ;
65 }
66 {box-shadow: none ;}
67 \LWR@orignewline

68 \textquotedbl}
69 % \LWR@htmldivclass{\LWR@mdthisenv}
```

mdframed environment may not work with the HTML versions of the following, so restore them to their originals while inside mdframed:

```
70 \let\hspace\LWR@print@hspace%
71 \renewcommand*{\rule}{\LWR@print@rule}
72 \LetLtxMacro\makebox\LWR@print@makebox%

73 \LWR@startpars%
74 \LWR@traceinfo{LWR@mdframedstart done}%
75 }
```

\LWR@mdframedend   Actions after an mdframe ends.

After closing the <div>, globally restore to the default environment type:

```
76 \newcommand*{\LWR@mdframedend}{
77 \LWR@traceinfo{LWR@mdframedend start}%
```

Close the custom <div>:

```
78 \LWR@htmldivclassend{\LWR@mdthisenv}
```

Reset future custom class to the default:

```
79 \gdef\LWR@mdthisenv{mdframed}
```

Resume paragraph handling:

```
80 \LWR@startpars%
81 \LWR@traceinfo{LWR@mdframedend done}%
82 }
```

§410.7 **Mdframed environment**

```
83 \renewenvironment{mdframed}[1][]{%
84 \color@begingroup%
85   \mdfsetup{userdefinedwidth=\linewidth,#1}%
86   \mdf@startcode%
87   \mdf@preenvsetting%
88   \ifdefempty{\mdf@firstframetitle}{}%
89        {\let\mdf@frametitlesave\mdf@frametitle%
90         \let\mdf@frametitle\mdf@firstframetitle%
91        }%
92   \ifvmode\nointerlineskip\fi%
93        \ifdefempty{\mdf@frametitle}{}%
94            {\mdfframedtitleenv{\mdf@frametitle}%
95 %            \mdf@@frametitle@use%
96            }%
97   \mdf@trivlist{\mdf@skipabove@length}%%
98   \mdf@settings%
99 %    \mdf@lrbox{\mdf@splitbox@one}%
100 %    \mdf@startinnercode%
101  }%
102  {%
103 %    \mdf@@ignorelastdescenders%
104   \par%
105 %      \unskip\ifvmode\nointerlineskip\hrule \@height\z@ \@width\hsize\fi%%
106   \ifmdf@footnoteinside%
107      \def\mdf@reserveda{%
108        \mdf@footnoteoutput%
109 %        \mdf@endinnercode%
110 %        \endmdf@lrbox%
111 %        \ifdefempty{\mdf@frametitle}{}%
112 %            {\mdfframedtitleenv{\mdf@frametitle}\mdf@@frametitle@use}%
113 %        \detected@mdf@put@frame
114      }%
115   \else%
116      \def\mdf@reserveda{%
117 %        \mdf@endinnercode%
118 %        \endmdf@lrbox%
119 %        \ifdefempty{\mdf@frametitle}{}%
120 %            {\mdfframedtitleenv{\mdf@frametitle}\mdf@@frametitle@use}%
121 %        \detected@mdf@put@frame%
122        \mdf@footnoteoutput%
123      }%
124   \fi%
125   \mdf@reserveda%
126  \aftergroup\endmdf@trivlist%
127 \color@endgroup%
128 \mdf@endcode%
129 }
```

\mdf@footnoteoutput

```
130 \renewrobustcmd*\mdf@footnoteoutput{%
131     \LWR@printpendingmpfootnotes%
```

132 `}`

§ 410.8 Titles and subtitles

\mdfframedtitleenv   {⟨*title*⟩}

Place the title inside a <div> of class mdframedtitle:

133 `\newlength{\LWR@titleroundcorner}`
134
135 `\renewrobustcmd\mdfframedtitleenv[1]{%`
136 `\LWR@traceinfo{LWR@mdframedtitleenv start}%`

Open a <div> with a custom class and custom style:

137 `\begin{BlockClass}[%`

Convert and print the title background color:

138 `background:`
139 `\LWR@mdfprintcolor{frametitlebackgroundcolor}`
140 `; \LWR@orignewline`

Convert and print the title rule:

141 `\ifbool{mdf@frametitlerule}{%`
142 `    border-bottom:`
143 `    \LWR@mdfprintlength{frametitlerulewidth}`
144 `    solid`
145 `    \LWR@mdfprintcolor{frametitlerulecolor}`
146 `    ; \LWR@orignewline`
147 `}{}%`

Finish the custom style and the opening <div> tag:

148 `]{mdframedtitle}%`

Print the title inside the <div>:

149 `\mdf@frametitlefont{\LWR@textcurrentfont{#1}}%`

Close the <div>:

150 `\end{BlockClass}%`
151 `\LWR@traceinfo{LWR@mdframedtitleenv end}%`
152 `}`

\LWR@mdfsubtitlecommon   {⟨*sub — or — subsub*⟩} [⟨*options*⟩] {⟨*title*⟩}

Common code for \LWR@mdfsubtitle and \LWR@mdfsubsubtitle.

Encapsulate the subtitle inside a <div> of class mdframedsubtitle:

153 `\NewDocumentCommand{\LWR@mdfsubtitlecommon}{m o m}`
154 `{% the following empty line is required`
155
156 `\LWR@traceinfo{LWR@mdframedsubtitlecommon start}%`

Open a <div> with a custom class and custom style:

157 `\begin{BlockClass}[%`

Convert and print the background color:

158 `background:`
159 `\LWR@mdfprintcolor{#1titlebackgroundcolor}`
160 `; \LWR@orignewline`

Convert and print the above line:

```
161 \ifbool{mdf@#1titleaboveline}{%
162     border-top:
163     \LWR@mdfprintlength{#1titleabovelinewidth}
164     solid
165     \LWR@mdfprintcolor{#1titleabovelinecolor}
166     ; \LWR@orignewline
167 }{}%
```

Convert and print the below line:

```
168 \ifbool{mdf@#1titlebelowline}{%
169     border-bottom:
170     \LWR@mdfprintlength{#1titlebelowlinewidth}
171     solid
172     \LWR@mdfprintcolor{#1titlebelowlinecolor}
173     ; \LWR@orignewline
174 }{}%
```

Finish the custom style and the opening <div> tag:

```
175 ]{mdframed#1title}%
```

Perform the original subtitle action:

```
176 \IfNoValueTF{#2}
177 {\@nameuse{LWR@origmdf#1title}{\csuse{mdf@#1titlefont}{\LWR@textcurrentfont{#3}}}}%
178 {\@nameuse{LWR@origmdf#1title}[#2]{\csuse{mdf@#1titlefont}{\LWR@textcurrentfont{#3}}}}%
```

Close the <div>:

```
179 \end{BlockClass}%
180 \LWR@traceinfo{LWR@mdframedsubtitlecommon end}%
181 }
```

\LWR@mdfsubtitle   [⟨*options*⟩] {⟨*title*⟩}

```
182 \newcommand*{\LWR@mdfsubtitle}{%
183 \LWR@mdfsubtitlecommon{sub}%
184 }
185 \let\mdfsubtitle\LWR@mdfsubtitle
```

\LWR@mdfsubsubtitle   [⟨*options*⟩] {⟨*title*⟩}

```
186 \newcommand*{\LWR@mdfsubsubtitle}{%
187 \LWR@mdfsubtitlecommon{subsub}%
188 }
189 \let\mdfsubsubtitle\LWR@mdfsubsubtitle
```

§ 410.9 New environments

\LWR@mdthisenv   Stores the environment of the frame about to be created:

```
190 \newcommand*{\LWR@mdthisenv}{mdframed}
```

\newmdenv   [⟨*options*⟩] {⟨*env-name*⟩}

Modified from the original to remember the environment.

```
191 \renewrobustcmd*\newmdenv[2][]{%
192 \newenvironment{#2}%
193 {%
194 \mdfsetup{#1}%
195 \renewcommand*{\LWR@mdthisenv}{md#2}%
196 \begin{mdframed}%
```

```
197 }
198 {\end{mdframed}}%
199 }
```

`\surroundwithmdframed`   [⟨*options*⟩] {⟨*environment*⟩}

Modified from the original to remember the environment.

```
200 \renewrobustcmd*{\surroundwithmdframed}[2][]{%
201 \BeforeBeginEnvironment{#2}{%
202 \renewcommand*{\LWR@mdthisenv}{md#2}%
203 \begin{mdframed}[#1]}%
204 \AfterEndEnvironment{#2}{\end{mdframed}}%
205 }
```

`\mdtheorem`   [⟨*mdframed-options*⟩] {⟨*envname*⟩} [⟨*numberedlike*⟩] {⟨*caption*⟩} [⟨*within*⟩]

Modified from the original to remember the environment.

```
206 \DeclareDocumentCommand{\mdtheorem}{ O{} m o m o }%
207 {\ifcsdef{#2}%
208    {\mdf@PackageWarning{Environment #2 already exits\MessageBreak}}%
209    {%
210     \IfNoValueTF {#3}%
211      {%#3 not given -- number relationship
212       \IfNoValueTF {#5}%
213        {%#3+#5 not given
214        \@definecounter{#2}%
215        \expandafter\xdef\csname the#2\endcsname{\@thmcounter{#2}}%
216        \newenvironment{#2}[1][]{%
217          \refstepcounter{#2}%
218          \ifstrempty{##1}%
219            {\let\@temptitle\relax}%
220            {%
221             \def\@temptitle{\mdf@theoremseparator%
222                             \mdf@theoremspace%
223                             \mdf@theoremtitlefont%
224                             \LWR@textcurrentfont{##1}}%  lwarp
225           \mdf@thm@caption{#2}{{#4}{\csname the#2\endcsname}{##1}}%
226           }%
227         \begin{mdframed}[#1,frametitle={\strut#4\ \csname the#2\endcsname%
228                                        \@temptitle}]}%
229       {\end{mdframed}}%
230     \newenvironment{#2*}[1][]{%
231       \ifstrempty{##1}{\let\@temptitle\relax}{\def\@temptitle{:\ ##1}}%
232       \begin{mdframed}[#1,frametitle={\strut#4\@temptitle}]}%
233       {\end{mdframed}}%
234     }%
235     {%#5 given -- reset counter
236     \@definecounter{#2}\@newctr{#2}[#5]%
237     \expandafter\xdef\csname the#2\endcsname{\@thmcounter{#2}}%
238     \expandafter\xdef\csname the#2\endcsname{%
239            \expandafter\noexpand\csname the#5\endcsname \@thmcountersep%
240              \@thmcounter{#2}}%
241     \newenvironment{#2}[1][]{%
242       \refstepcounter{#2}%
243       \ifstrempty{##1}%
244         {\let\@temptitle\relax}%
245         {%
246          \def\@temptitle{\mdf@theoremseparator%
247                          \mdf@theoremspace%
```

```
248                              \mdf@theoremtitlefont%
249                              \LWR@textcurrentfont{##1}}% lwarp
250             \mdf@thm@caption{#2}{{#4}{\csname the#2\endcsname}{##1}}%
251                }
252           \begin{mdframed}[#1,frametitle={\strut#4\ \csname the#2\endcsname%
253                                           \@temptitle}]}%
254          {\end{mdframed}}%
255         \newenvironment{#2*}[1][]{%
256           \ifstrempty{##1}%
257             {\let\@temptitle\relax}%
258             {%
259              \def\@temptitle{\mdf@theoremseparator%
260                              \mdf@theoremspace%
261                              \mdf@theoremtitlefont%
262                              \LWR@textcurrentfont{##1}}% lwarp
263            \mdf@thm@caption{#2}{{#4}{\csname the#2\endcsname}{##1}}%
264             }%
265            \begin{mdframed}[#1,frametitle={\strut#4\@temptitle}]}%
266           {\end{mdframed}}%
267         }%
268      }%
269      {%#3 given -- number relationship
270        \global\@namedef{the#2}{\@nameuse{the#3}}%
271        \newenvironment{#2}[1][]{%
272          \refstepcounter{#3}%
273          \ifstrempty{##1}%
274            {\let\@temptitle\relax}%
275            {%
276             \def\@temptitle{\mdf@theoremseparator%
277                             \mdf@theoremspace%
278                             \mdf@theoremtitlefont%
279                             \LWR@textcurrentfont{##1}}% lwarp
280           \mdf@thm@caption{#2}{{#4}{\csname the#2\endcsname}{##1}}%
281             }
282           \begin{mdframed}[#1,frametitle={\strut#4\ \csname the#2\endcsname%
283                                           \@temptitle}]}%
284          {\end{mdframed}}%
285        \newenvironment{#2*}[1][]{%
286          \ifstrempty{##1}{\let\@temptitle\relax}{\def\@temptitle{:\ ##1}}%
287          \begin{mdframed}[#1,frametitle={\strut#4\@temptitle}]}%
288         {\end{mdframed}}%
289      }%
290     \BeforeBeginEnvironment{#2}{\renewcommand*{\LWR@mdthisenv}{md#2}}% lwarp
291     \BeforeBeginEnvironment{#2*}{\renewcommand*{\LWR@mdthisenv}{md#2}}% lwarp
292     }%
293  }
```

\newmdtheoremenv   [⟨1: mdframed-options⟩] {⟨2: envname⟩} [⟨3: numberedlike⟩] {⟨4: caption⟩}
                            [⟨5: within⟩]

Modified from the original to remember the environment.

```
294 \DeclareDocumentCommand\newmdtheoremenv{O{} m o m o }{%
295 \ifboolexpr{ test {\IfNoValueTF {#3}} and test {\IfNoValueTF {#5}} }%
296    {\newtheorem{#2}{#4}}%
297    {%
298     \IfValueT{#3}{\newtheorem{#2}[#3]{#4}}%
299     \IfValueT{#5}{\newtheorem{#2}{#4}[#5]}%
300    }%
301 \BeforeBeginEnvironment{#2}{%
```

```
302 \renewcommand*{\LWR@mdthisenv}{md#2}%
303 \begin{mdframed}[#1]}%
304 \AfterEndEnvironment{#2}{%
305 \end{mdframed}}%
306 }
```

File 299   **`lwarp-mdwmath.sty`**

§ 411   Package   # mdwmath

(*Emulates or patches code by* MARK WOODING.)

mdwmath (*Pkg*)   mdwmath is used as-is for SVG math, and is emulated for MATHJAX.

for HTML output: `1 \LWR@ProvidesPackagePass{mdwmath}[1996/04/11]`

```
2 \begin{warpMathJax}
3 \CustomizeMathJax{\let\LWRmdwmathsqrt\sqrt}
4 \CustomizeMathJax{\renewcommand{\sqrt}{\ifstar\LWRmdwmathsqrt\LWRmdwmathsqrt}}
5 \CustomizeMathJax{\newcommand{\bitand}{\mathbin\&}}
6 \CustomizeMathJax{\def\bitor{\mathbin\mid}}
7 \CustomizeMathJax{\def\dblor{\mathbin{\mid\mid}}}
8 \CustomizeMathJax{\def\dbland{\mathbin{\mathrel\bitand\mathrel\bitand}}}
9 \end{warpMathJax}
```

File 300   **`lwarp-media9.sty`**

§ 412   Package   # media9

media9 (*Pkg*)   media9 is emulated.

The packages multimedia, movie15, and media9 are supported.

HTML5 <audio> and <video> objects are created for .mp3 and .mp4 files.

HTML5 <embed> objects are created for http and ftp links.

\href links are created for other media types. (Unfortunately, there is not much overlap between the file types supported for print output and the file types supported by HTML5.)

For media9, a multimedia object is inserted for each addresource=, as well as each flashvars source= and src=. This may result in duplicate objects.

Undesired objects may be nullified by placing them inside \warpprintonly or the warpprint environment.

Each HTML multimedia object includes the poster text, except for <embed> objects. For movie15, the text option is supported to specify the poster text.

The width, height, and totalheight options are supported. The HTML object is scaled according to the display width, correctly compensating for either tall or wide viewports.

Other options are ignored.

media9 \addmediapath is supported. It is assumed that the same path structure will exist for the HTML document.

HTML5 media controls are always specified for each <audio> and <video> object.

media9 slideshows are not supported.

\hyperlinkmovie, \movieref, and \mediabutton are not supported.

3D objects are not supported.

If using a YouTube™ video, use an "embedded" URL with …/embed/… instead of …/v/…

⚠ & in a URL   Many special characters are converted to regular catcode 12 characters for use inside a URL. & is used in the flash variables fields, which are split with xparse \SplitList, which does not seem to work with a catcode 12 divider token, so & is not converted to catcode 12, and will not work in a URL with media9. Using & in a URL in a flashvars field may also cause parsing problems with print output, as well.

for HTML output: 1 \LWR@ProvidesPackageDrop{media9}[2019/02/21]

2 \LWR@origRequirePackage{lwarp-common-multimedia}
3
4 \RequirePackage{xkeyval}

\addmediapath   {⟨*path*⟩}

Supported.

5 \newcommand*{\LWR@medianine@path}{}
6
7 \newcommand*{\addmediapath}[1]{\appto\LWR@medianine@path{{#1}}}

The options and poster text are reused in several places.

8 \newcommand*{\LWR@medianine@postertext}{}
9 \newcommand*{\LWR@medianine@options}{}

Each addresource can generate a multimedia object.

10 \define@key{LWR@medianine}{addresource}{%
11   \expandafter\LWR@multimedia\expandafter[\LWR@medianine@options]
12   {\LWR@medianine@postertext}
13   {#1}
14 }

Each flashvars source can generate a multimedia object.

15 \newcommand*{\LWR@medianine@flashvarsb}[1]{%
16   \IfBeginWith{#1}{source=}{%
17   \StrGobbleLeft{#1}{7}[\LWR@tempone]%
18   \expandafter\LWR@multimedia\expandafter[\LWR@medianine@options]%
19   {\LWR@medianine@postertext}%
20   {\LWR@tempone}%
21   }{}%
22   \IfBeginWith{#1}{src=}{%
23   \StrGobbleLeft{#1}{4}[\LWR@tempone]%

```
24          \expandafter\LWR@multimedia\expandafter[\LWR@medianine@options]%
25              {\LWR@medianine@postertext}%
26              {\LWR@tempone}%
27      }{}%
28 }
29
30 \NewDocumentCommand{\LWR@medianine@flashvars}{ >{\SplitList{&}} m }{%
31      \ProcessList {#1}{\LWR@medianine@flashvarsb}%
32 }
33
34 \define@key{LWR@medianine}{flashvars}{%
35      \LWR@medianine@flashvars{#1}%
36 }
```

\includemedia   [⟨*options*⟩] {⟨*poster text*⟩} {⟨*file or url*⟩}

```
37 \newcommand*{\LWR@includemediab}[3][]{%
38      \let\input@path\LWR@medianine@path%
39      \renewcommand*{\LWR@medianine@options}{#1}%
40      \renewcommand*{\LWR@medianine@postertext}{#2}%
41      \setkeys*{LWR@medianine}{#1}%
42      \IfBeginWith{#3}{http}{\LWR@multimedia[#1]{#2}{#3}}{%
43      \IfBeginWith{#3}{HTTP}{\LWR@multimedia[#1]{#2}{#3}}{%
44      \IfBeginWith{#3}{ftp}{\LWR@multimedia[#1]{#2}{#3}}{%
45      \IfBeginWith{#3}{FTP}{\LWR@multimedia[#1]{#2}{#3}}{%
46      }}}}%
47      \endgroup%
48 }
49
50 \newrobustcmd*{\includemedia}{%
51      \begingroup%
52      \LWR@linkmediacatcodes%
53      \LWR@includemediab%
54 }
```

\mediabutton   [⟨*options*⟩] {⟨*text*⟩}

Ignored.

```
55 \newcommand*{\mediabutton}[2][]{}
```

File 301   `lwarp-memhfixc.sty`

§ 413   Package   **memhfixc**

memhfixc (*Pkg*)   memhfixc is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{memhfixc}[2013/05/30]

File 302   `lwarp-menukeys.sty`

§ 414   Package   **menukeys**

(*Emulates or patches code by* Tobias Weh.)

menukeys (*Pkg*) menukeys is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{menukeys}[2020/12/19]

Patch to use a lateximage whose alt text is the contents of this use of the macro. A hash on these contents allows the reuse of the image for each instance of the same contents.

```
2 \VerifyCommand[lwarp][menukeys]{\tw@define@menu@macro@}{A3C988E47073504556D744EF08443B1D}
3
4 \xpatchcmd{\tw@define@menu@macro@}
5     {\@nameuse{tw@style@#4@pre}}
6     {%
7         \begin{lateximage}*[\detokenize{##2}]?%
8         \@nameuse{tw@style@#4@pre}%
9     }
10    {}
11    {\LWR@patcherror{menukeys}{tw@define@menu@macro@}}
12
13 \xpatchcmd{\tw@define@menu@macro@}
14    {\@nameuse{tw@style@#4@post}}
15    {%
16        \@nameuse{tw@style@#4@post}%
17        \end{lateximage}%
18    }
19    {}
20    {\LWR@patcherror{menukeys}{tw@define@menu@macro@ B}}
```

Patch the existing macros:

```
21 \renewmenumacro{\menu}[>]{menus}
22 \renewmenumacro{\directory}[/]{paths}
23 \renewmenumacro{\keys}[+]{roundedkeys}
```

File 303 **`lwarp-metalogo.sty`**

§ 415 Package **metalogo**

(*Emulates or patches code by* Andrew Gilbert Moschou.)

metalogo (*Pkg*) metalogo is used in print mode, and emulated in HTML.

for HTML output: 1 \LWR@ProvidesPackagePass{metalogo}[2010/05/29]

```
2 \newcommand*{\LWR@HTML@setlogokern}[2]{}
3 \newcommand*{\LWR@HTML@setlogodrop}[2][XeTeX]{}
4 \newcommand*{\LWR@HTML@setLaTeXa}[1]{}
5 \newcommand*{\LWR@HTML@setLaTeXee}[1]{}
6 \newcommand*{\LWR@HTML@seteverylogo}[1]{}
7 \newcommand*{\LWR@HTML@everylogo}[1]{}
8
9 \LWR@formatted{setlogokern}
10 \LWR@formatted{setlogodrop}
11 \LWR@formatted{setLaTeXa}
12 \LWR@formatted{setLaTeXee}
13 \LWR@formatted{seteverylogo}
14 \LWR@formatted{everylogo}
```

File 304   **`lwarp-metalogox.sty`**

§ 416   Package   # metalogox

(*Emulates or patches code by* BRIAN DUNN.)

metalogox (*Pkg*)   metalogox is patched for use by lwarp.

for HTML output: 1 `\LWR@ProvidesPackagePass{metalogox}[2019/01/20]`

`\AtBeginDocument`, adjust the logo setting according to the font which is active at that moment.

```
2 \AtBeginDocument{
3    \let\LWR@metalogox@currentformatting\LWR@formatting
4    \renewcommand*{\LWR@formatting}{print}%
5    \autoadjustlogos*
6    \let\LWR@formatting\LWR@metalogox@currentformatting
7 }
```

File 305   **`lwarp-mhchem.sty`**

§ 417   Package   # mhchem

(*Emulates or patches code by* MARTIN HENSEL.)

mhchem (*Pkg*)   mhchem is patched for use by lwarp.

without MATHJAX   Without MATHJAX, mhchem expressions are converted to SVG math. Inline expressions use hashed filenames to allow reuse, and assume that any mhchem options are global.

MATHJAX with mhchem extension   For MATHJAX, the mhchem extension is used if the mhchem expression is used inside a math expression:

> `$\ce{C6H5-CHO}$`

To force the use of SVG math for an expression which does not work with MATHJAX, place the expression between `\displaymathother` and `\displaymathnormal`:

> `\displaymathother`
> `\[ \ce{ ... } \]`   . . .   `$ \ce { ... } $`
> `\displaymathnormal`

not inside math   If *not* used inside a math expression, lwarp converts standalone mhchem expressions into SVG math images.

⚠ nested math   When producing HTML output without the MATHJAX mhchem extension, lwarp does not support the use of nested dollar signs in mhchem expressions.

For some examples from the mhchem manual, change as follows:

```
$\ce{NaOH(aq,$\infty$)}$              % old
$\ce{NaOH(aq,\infty)}$                % new

$\ce{Fe(CN)_{$\frac{6}{2}}$}$         % old
$\ce{Fe(CN)_{\frac{6}{2}}}$           % new

$\ce{NO_$x$}$                         % old
$\ce{NO_x}$                           % new

$\ce{NO_${x}$}$                       % old
$\ce{NO_{x}}$                         % new

$\ce{$cis${-}[PtCl2(NH3)2]}$          % old
$\ce{\mathit{cis}{-}[PtCl2(NH3)2]}$   % new
```

for HTML output: 1 \LWR@ProvidesPackagePass{mhchem}[2024/01/29]

The original definition of \ce:

2 \LetLtxMacro\LWR@mhchem@origce\ce

The new definition, called from the new \ce after math shift is set. The starred lateximage uses a hashed filename for the svg image. The alt tag is set to the mhchem expression.

```
3 \newcommand{\LWR@mhchem@HTML@ce}[1]{%
4     \LWR@findcurrenttextcolor% sets \LWR@tempcolor
5     \edef\LWR@mhchemalt{\LWR@HTMLsanitizedetokenized{\detokenize{#1}}}%
6     \ifbool{LWR@xfakebold}%
7         {\def\LWR@tempone{Y}}%
8         {\def\LWR@tempone{N}}%
9     \begin{lateximage}%
10        *% hash
11        [% alt
12            \textbackslash{}%
13            ce%
14            \{\LWR@mhchemalt\}%
15        ]%
16        *% no open close tags in alt
17        ?% do not detokenize alt
18        [% add'l alt
19            FM\LWR@f@family%
20            SR\LWR@f@series%
21            SH\LWR@f@shape%
22            SHC\LWR@f@shapecaps%
23            CL\LWR@tempcolor%
24            FB\LWR@tempone% xfakebold
25        ]%
26    \LWR@setcurrentfont%
27    \LWR@mhchem@origce{#1}%
28    \end{lateximage}%
29    \endgroup%
30    \addtocounter{LWR@mhchem@cedepth}{-1}%
31 }
```

Only set math shift if outer depth:

```
32 \newcounter{LWR@mhchem@cedepth}
33 \setcounter{LWR@mhchem@cedepth}{0}
```

The new \ce. Sets math shift then continues.

```
34 \renewcommand{\ce}{%
35     \begingroup%
36     \ifnumequal{\value{LWR@mhchem@cedepth}}{0}{%
37         \catcode`\$=3% math shift
38     }{}%
39     \addtocounter{LWR@mhchem@cedepth}{1}%
40     \LWR@mhchem@HTML@ce%
41 }
```

The original definition of \cesplit:

```
42 \LetLtxMacro\LWR@mhchem@origcesplit\cesplit
```

The new definition, called from the new \cesplit after math shift is set. The
starred lateximage uses a hashed filename for the SVG image. The alt tag is set
to the mhchem expression.

```
43 \newcommand*{\LWR@mhchem@HTML@cesplit}[2]
44 {%
45     \LWR@findcurrenttextcolor% sets \LWR@tempcolor
46     \ifbool{LWR@xfakebold}%
47         {\def\LWR@tempone{Y}}%
48         {\def\LWR@tempone{N}}%
49     \begin{lateximage}%
50         *% hash
51         [% alt
52             \textbackslash{}%
53             cesplit%
54             \{\LWR@HTMLsanitizedetokenized{\detokenize{#2}}\}%
55         ]%
56         *% no open/close tags in alt
57         ?% no detokenize alt
58         [% add'l alt
59             FM\LWR@f@family%
60             SR\LWR@f@series%
61             SH\LWR@f@shape%
62             SHC\LWR@f@shapecaps%
63             CL\LWR@tempcolor%
64             FB\LWR@tempone% xfakebold
65         ]%
66     \LWR@setcurrentfont%
67     \LWR@mhchem@origcesplit{#1}{#2}%
68     \end{lateximage}%
69     \endgroup%
70 }
```

Only set math shift if outer depth:

```
71 \newcounter{LWR@mhchem@cesplitdepth}
72 \setcounter{LWR@mhchem@cesplitdepth}{0}
```

The new \cesplit. Sets math shift then continues.

```
73 \renewcommand{\cesplit}{%
```

```
74    \begingroup%
75    \ifnumequal{\value{LWR@mhchem@cesplitdepth}}{0}{%
76        \catcode`\$=3% math shift
77    }{}%
78    \addtocounter{LWR@mhchem@cesplitdepth}{1}%
79    \LWR@mhchem@HTML@cesplit%
80 }
```

Resore originals inside a lateximage:

```
81 \appto\LWR@restoreorigformatting{%
82 \LetLtxMacro\ce\LWR@mhchem@origce%
83 \LetLtxMacro\cesplit\LWR@mhchem@origcesplit%
84 }
85
86 \begin{warpMathJax}
87 \CustomizeMathJax{\require{mhchem}}
88 \end{warpMathJax}
```

File 306   **`lwarp-microtype.sty`**

§ 418   Package   # microtype

(*Emulates or patches code by* R SCHLICHT.)

microtype (*Pkg*)   microtype is pre-loaded by lwarp. All user options and macros are ignored and disabled.

for HTML output: Discard all options for lwarp-microtype:

```
1 \LWR@ProvidesPackageDrop{microtype}[2018/01/14]

2 \DeclareDocumentCommand{\DeclareMicrotypeSet}{o m m}{}
3 \DeclareDocumentCommand{\UseMicrotypeSet}{o m}{}
4 \DeclareDocumentCommand{\DeclareMicrotypeSetDefault}{o m}{}
5 \DeclareDocumentCommand{\SetProtrusion}{o m m}{}
6 \DeclareDocumentCommand{\SetExpansion}{o m m}{}
7 \DeclareDocumentCommand{\SetTracking}{o m m}{}
8 \DeclareDocumentCommand{\SetExtraKerning}{o m m}{}
9 \DeclareDocumentCommand{\SetExtraSpacing}{o m m}{}
10 \DeclareDocumentCommand{\DisableLigatures}{o m}{}
11 \DeclareDocumentCommand{\DeclareCharacterInheritance}{o m m}{}
12 \DeclareDocumentCommand{\DeclareMicrotypeVariants}{m}{}
13 \DeclareDocumentCommand{\DeclareMicrotypeAlias}{m m}{}
14 \DeclareDocumentCommand{\LoadMicrotypeFile}{m}{}
15 \DeclareDocumentCommand{\DeclareMicrotypeBabelHook}{m m}{}
16 \DeclareDocumentCommand{\microtypesetup}{m}{}
17 \DeclareDocumentCommand{\microtypecontext}{m}{}
18 \DeclareDocumentCommand{\textmicrotypecontext}{m m}{#2}
19 \IfPackageLoadedTF{letterspace}{\let\MT@textls\relax}{%
20 \DeclareDocumentCommand{\lsstyle}{}{}
21 \DeclareDocumentCommand{\textls}{o +m}{}
22 \DeclareDocumentCommand{\lslig}{m}{#1}
23 }
24 \def\DeclareMicrotypeSet#1#{\@gobbletwo}
25 \def\DeclareMicrotypeVariants#1#{\@gobble}
26 \@onlypreamble\DeclareMicrotypeSet
```

```
27 \@onlypreamble\UseMicrotypeSet
28 \@onlypreamble\DeclareMicrotypeSetDefault
29 \@onlypreamble\DisableLigatures
30 \@onlypreamble\DeclareMicrotypeVariants
31 \@onlypreamble\DeclareMicrotypeBabelHook
```

File 307   **`lwarp-midfloat.sty`**

§ 419   Package   # midfloat

(*Emulates or patches code by* Sigitas Tolušis.)

midfloat (*Pkg*)   midfloat is emulated.

for HTML output: `1 \LWR@ProvidesPackageDrop{midfloat}[2012/05/29]`

```
2 \newenvironment{strip}[1][]{}{}
3 \newskip\stripsep
```

File 308   **`lwarp-midpage.sty`**

§ 420   Package   # midpage

midpage (*Pkg*)   midpage is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{midpage}[2009/09/03]`

```
2 \newenvironment{midpage}
3 {\begin{BlockClass}[%
4     \LWR@print@mbox{margin-top:6ex} ; \LWR@print@mbox{margin-bottom:6ex}%
5 ]{midpage}}
6 {\end{BlockClass}}
```

File 309   **`lwarp-minibox.sty`**

§ 421   Package   # minibox

(*Emulates or patches code by* Will Robertson.)

minibox (*Pkg*)   minibox is patched for use by lwarp.

Due to HTML limitations regarding paragraphs and <div>s, miniboxes inline with other text will appear on their own line.

for HTML output: `1 \LWR@ProvidesPackagePass{minibox}[2013/06/21]`

```
2 \ExplSyntaxOn
3 \newcommand\LWR@HTML@minibox[2][]{%
4     \LWR@stoppars%
5     \group_begin:
```

```
6      \keys_set:nn {minibox} {#1}
7      \bool_if:NTF \l_minibox_frame_bool
8      {
9          \setlength\fboxrule{\l_minibox_rule_dim}
10         \setlength\fboxsep{\l_minibox_pad_dim}
11         \fboxBlock{%
12             \begin{tabular}[\l_minibox_tabular_valign_tl]%
13               {\l_minibox_tabular_preamble_tl}
14                 {#2}
15             \end{tabular}
16         }%
17     }
18     {
19         \begin{BlockClass}[display:inline-block]{minibox}
20         \begin{tabular}[\l_minibox_tabular_valign_tl]%
21           {\l_minibox_tabular_preamble_tl}
22             {#2}
23         \end{tabular}
24         \end{BlockClass}
25     }
26     \group_end:
27     \LWR@startpars%
28 }
29 \ExplSyntaxOff
30
31 \LWR@formatted{minibox}
```

File 310   `lwarp-minitoc.sty`

§ 422   Package   # minitoc

minitoc (*Pkg*)   minitoc is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{minitoc}[2018/07/12]

mtcoff disables minitoc.

2 \usepackage{mtcoff}

File 311   `lwarp-minted.sty`

§ 423   Package   # minted

(*Emulates or patches code by* GEOFFREY M. POORE.)

minted (*Pkg*)   minted is patched for use by lwarp.

⚠ limitations   mathescape, texcomments, and highlightlines are disabled. Line numbers on the right will not be aligned. Due to *pdftotext*, extra spaces may appear in broken lines if other formatting is included.

for HTML output: 1 \LWR@ProvidesPackagePass{minted}[2024/11/17]

Several options are forced, since they are unsupported by lwarp.

mathescape and texcomments are disabled because the result might not be linear characters, so might not be interpreted correctly by *pdftotext*, which is used by lwarp to convert from PDF output to HTML text.

breaklines is forced true to use the correctly functioning code in fvextra.

highlightlines is forced false because fvextra processes highlight lines one line at a time, and minted could add a large amount of formatting code, overflowing the line. When set false, fvextra can handle the line breaks which prevent line overflow.

LWR@HTMLsanitize@tmpb@enable is used to turn off HTML sanitization early in the verbatim conversion, otherwise minted would then colorize the sanitized results, breaking the HTML entities in lwarp's HTML output.

```
2 \newcommand*{\LWR@minted@forcekeys}{%
3     \pgfkeys{/minted/cmd/.cd,mathescape=false}%
4     \pgfkeys{/minted/cmd/.cd,texcomments=false}%
5     \pgfkeys{/minted/cmd/.cd,highlightlines=false}%
6     \pgfkeys{/minted/cmd/.cd,breaklines}%
7     \boolfalse{LWR@HTMLsanitize@tmpb@enable}% lwarp
8 }
```

Used by MintedVerbatim. Force breaklines.

```
9 \VerifyCommand[lwarp][minted]{\minted@highlight@i}{B3E47D0B3B8933CCB16FAB0735F408A8}
10
11 \xpretocmd{\minted@highlight@i}
12     {\fvset{breaklines}}
13     {}
14     {\LWR@patcherror{minted}{minted@highlight@i}}

15 \VerifyCommand[lwarp][minted]{\RobustMintInlineProcess@verbatim}
16     {A6C5557F26D2A7E2C8DC542A2ABD892B}
17
18 \xpatchcmd{\RobustMintInlineProcess@verbatim}
19     {\edef\minted@lexer{#2}}
20     {%
21         \LWR@minted@forcekeys%      lwarp
22         \edef\minted@lexer{#2}%
23     }
24     {}
25     {\LWR@patcherror{minted}{RobustMintInlineProcess@verbatim}}
26
27
28 \VerifyCommand[lwarp][minted]{\RobustMintInlineProcess@highlight}
29     {6374066A5D0D1CF9636E667AAFAF81AB}
30
31 \xpatchcmd{\RobustMintInlineProcess@highlight}
32     {\edef\minted@lexer{#2}}
33     {%
34         \LWR@minted@forcekeys%      lwarp
35         \edef\minted@lexer{#2}%
36     }
37     {}
38     {\LWR@patcherror{minted}{RobustMintInlineProcess@highlight}}
39
40
41 \VerifyCommand[lwarp][minted]{\RobustMintProcess@verbatim}
```

```
42     {C1C8E870B5E8E3E8E572CA50ADF69953}
43
44 \xpatchcmd{\RobustMintProcess@verbatim}
45     {\edef\minted@lexer{#2}}
46     {%
47         \LWR@minted@forcekeys%     lwarp
48         \edef\minted@lexer{#2}%
49     }
50     {}
51     {\LWR@patcherror{minted}{RobustMintProcess@verbatim}}
52
53
54 \VerifyCommand[lwarp][minted]{\RobustMintProcess@highlight}
55     {D426304D27561EE1139B30E897A7C2CA}
56
57 \xpatchcmd{\RobustMintProcess@highlight}
58     {\edef\minted@lexer{#2}}
59     {%
60         \LWR@minted@forcekeys%     lwarp
61         \edef\minted@lexer{#2}%
62     }
63     {}
64     {\LWR@patcherror{minted}{RobustMintProcess@highlight}}
65
66
67 \VerifyCommand[lwarp][minted]{\MintedBegin@verbatim}
68     {B16E0A3A51DBF9F39609F2135572D7C2}
69
70 \xpatchcmd{\MintedBegin@verbatim}
71     {\edef\minted@lexer{#2}}
72     {%
73         \LWR@minted@forcekeys%     lwarp
74         \edef\minted@lexer{#2}%
75     }
76     {}
77     {\LWR@patcherror{minted}{MintedBegin@verbatim}}
78
79
80 \VerifyCommand[lwarp][minted]{\MintedBegin@highlight}
81     {A8C3A60BE3C6AF3A401071A9C5FE8B39}
82
83 \xpatchcmd{\MintedBegin@highlight}
84     {\edef\minted@lexer{#2}}
85     {%
86         \LWR@minted@forcekeys%     lwarp
87         \edef\minted@lexer{#2}%
88     }
89     {}
90     {\LWR@patcherror{minted}{MintedBegin@highlight}}
91
92 \ifbool{minted@placeholder}%
93 {\let\MintedBegin\MintedBegin@placeholder
94  \let\MintedEnd\MintedEnd@placeholder}%
95 {\ifbool{minted@verbatim}%
96  {\let\MintedBegin\MintedBegin@verbatim
97   \let\MintedEnd\MintedEnd@verbatim}%
98  {\let\MintedBegin\MintedBegin@highlight
99   \let\MintedEnd\MintedEnd@highlight}}
100
101
```

```
102 \VerifyCommand[lwarp][minted]{\RobustInputMintedProcess@verbatim}
103     {8B475FE3FD41C8026A8447C5C984AE9F}
104
105 \xpatchcmd{\RobustInputMintedProcess@verbatim}
106     {\edef\minted@lexer{#2}}
107     {%
108         \LWR@minted@forcekeys%     lwarp
109         \edef\minted@lexer{#2}%
110     }
111     {}
112     {\LWR@patcherror{minted}{RobustInputMintedProcess@verbatim}}
113
114
115 \VerifyCommand[lwarp][minted]{\RobustInputMintedProcess@highlight}
116     {A3D22F5B26557361E26D55FD0E8701CD}
117
118 \xpatchcmd{\RobustInputMintedProcess@highlight}
119     {\edef\minted@lexer{#2}}
120     {%
121         \LWR@minted@forcekeys%     lwarp
122         \edef\minted@lexer{#2}%
123     }
124     {}
125     {\LWR@patcherror{minted}{RobustInputMintedProcess@highlight}}
```

To add sanitization during the final output, adjust several characters to use HTML entities when loading the pygmentized results,

Not using \VerifyCommand here because this is a simple patch, not likely to be affected by other changes to the original.

```
126 \xpatchcmd{\minted@highlight@i}
127     {\input{\minted@highlightfilepath}}
128     {%
129         \LWR@minted@overrides%  lwarp
130         \input{\minted@highlightfilepath}%
131     }
132     {}
133     {\LWR@patcherror{minted}{minted@highlight@i}}
134
135 \xpatchcmd{\minted@highlight@create}
136     {\input{\minted@highlightfilepath}}
137     {%
138         \LWR@minted@overrides%  lwarp
139         \input{\minted@highlightfilepath}%
140     }
141     {}
142     {\LWR@patcherror{minted}{minted@highlight@create}}
```

These macros are used inside the *.pygtex files to format several individual characters. These are revised to use HTML entities. The backquote grave is not supported by pygments.

```
143 \newcommand*{\LWR@minted@overrides}{
144 \def\PYGZam{\char`\&amp;}
145 \def\PYGZlt{\char`\&lt;}
146 \def\PYGZgt{\char`\&gt;}
147 \def\PYGZsq{\char`\&apos;}
148 }
```

File 312 **`lwarp-mismath.sty`**

§ 424 Package **mismath**

(*Emulates or patches code by* ANTOINE MISSIER.)

mismath (*Pkg*) mismath is patched for svg math, and emulated for MATHJAX.

⚠ MATHJAX \enumber, \inumber, \jnumber, and \pinumber are ignored for MATHJAX, except that \itpi is made available as a clone of \pi.

\MathUp, \MathIt, \MathNumbers, and \MathNormal are ignored in MATHJAX.

For MATHJAX, \boldvect and \arrowvect are honored if in the preamble.

If \boldvectcommand is set to \mathbf in the preamble, it will be used for MATH-JAX, otherwise it will default to \mathit. \boldvectcommand may also be set with \CustomizeMathJax in the preamble. See section 8.7.7. Note that as of this writing there is not a bold italic font across all MATHJAX fonts.

If \probastyle is set to \mathbb in the preamble, it will be used for MATHJAX, otherwise it will default to \mathrm. \probastyle may be set with \CustomizeMathJax in the preamble.

If \mathset is set to \mathbb in the preamble, it will be used for MATHJAX, otherwise it will default to \mathbf. \mathset may be set with \CustomizeMathJax in the preamble.

for HTML output:
```
1 \LWR@ProvidesPackagePass{mismath}[2024/06/16]
```

For MATHJAX, used in the HTML comment before the environment.

```
2 \ifbool{mathjax}{
3     \RenewEnviron{mathcols}{%
4         \preto\BODY{\begin{aligned}\displaystyle}
5         \appto\BODY{\end{aligned}}
6         \expandafter\(\BODY\)
7     }
8 }% mathjax
```

For svg math. The lateximage restores the original defintion of the math environment.

```
9 {% svg
10     \renewenvironment{mathcols}{
11         \begin{lateximage}
12         \begin{math}
13         \begin{aligned}\displaystyle
14     }{
15         \end{aligned}%
16         \end{math}
17         \end{lateximage}
18     }
19 }% svg
20
```

```
21 \renewcommand{\changecol}{
22     \end{aligned}    \qquad
23     \begin{aligned}\displaystyle
24 }
25
26 \begin{warpMathJax}
27 \CustomizeMathJax{\require{upgreek}}% for \Updelta
28
29 \CustomizeMathJax{\newcommand{\mathup}[1]{\mathrm{#1}}}
30 \CustomizeMathJax{\newcommand{\e}{\mathrm{e}}}
31 \CustomizeMathJax{\newcommand{\i}{\mathrm{i}}}
32 \CustomizeMathJax{\newcommand{\j}{\mathrm{j}}}
33
34 \CustomizeMathJax{\let\mathbfsfit\mathbfit}% not sans
35 \CustomizeMathJax{\let\tensor\mathbfsfit}
36
37 \CustomizeMathJax{\newcommand{\boldvect}{}}
38 \CustomizeMathJax{\newcommand{\arrowvect}{}}
39 \CustomizeMathJax{\newcommand{\pinumber}[1][]{}}
40 \CustomizeMathJax{\newcommand{\hvect}[1]{\vec{\vphantom{h}#1}}}
41 \CustomizeMathJax{\newcommand{\hvec}[1]{\vec{\vphantom{t}#1}}}
42 \CustomizeMathJax{%
43     \newcommand{\norm}[1]{\left\vert\left\vert#1\right\vert\right\vert}
44 }
45 \CustomizeMathJax{\newcommand{\di}{\mathop{}\!\mathrm{d}}}
46
47 \CustomizeMathJax{\newcommand{\upDelta}{\Updelta}}
48 \CustomizeMathJax{\newcommand{\opDelta}{\Updelta}}
49 \CustomizeMathJax{\newcommand{\opdelta}{\updelta}}
50
51 \CustomizeMathJax{\newcommand{\P}{\operatorname{\probastyle{P}}}}
52 \CustomizeMathJax{\newcommand{\E}{\operatorname{\probastyle{E}}}}
53 \CustomizeMathJax{\newcommand{\V}{\operatorname{\probastyle{V}}}}
54 \CustomizeMathJax{\newcommand{\Par}{\unicode{x00B6}}}
55
56 \CustomizeMathJax{\DeclareMathOperator{\adj}{adj}}
57 \CustomizeMathJax{\DeclareMathOperator{\Aut}{Aut}}
58 \CustomizeMathJax{\DeclareMathOperator{\codim}{codim}}
59 \CustomizeMathJax{\DeclareMathOperator{\Conv}{Conv}}
60 \CustomizeMathJax{\DeclareMathOperator{\cov}{cov}}
61 \CustomizeMathJax{\DeclareMathOperator{\Cov}{Cov}}
62 \CustomizeMathJax{\newcommand{\curl}{\operatorname{\vect{\mathrm{curl}}}}}
63 \CustomizeMathJax{\DeclareMathOperator{\divg}{div}}
64 \CustomizeMathJax{\DeclareMathOperator{\End}{End}}
65
66 \CustomizeMathJax{\DeclareMathOperator{\erf}{erf}}
67 \CustomizeMathJax{\newcommand{\grad}{\operatorname{\vect{\mathrm{grad}}}}}
68 \CustomizeMathJax{\DeclareMathOperator{\id}{id}}
69 \CustomizeMathJax{\DeclareMathOperator{\Id}{Id}}
70 \CustomizeMathJax{\DeclareMathOperator{\im}{im}}
71 \CustomizeMathJax{\let\oldIm\Im}
72 \CustomizeMathJax{\renewcommand{\Im}{\operatorname{Im}}}
73 \CustomizeMathJax{\DeclareMathOperator{\lb}{lb}}
74 \CustomizeMathJax{\DeclareMathOperator{\lcm}{lcm}}
75
76 \CustomizeMathJax{\DeclareMathOperator{\rank}{rank}}
77 \CustomizeMathJax{\let\oldRe\Re}
78 \CustomizeMathJax{\renewcommand{\Re}{\operatorname{Re}}}
79 \CustomizeMathJax{\newcommand{\rot}{\operatorname{\vect{\mathrm{rot}}}}}
80 \CustomizeMathJax{\DeclareMathOperator{\sgn}{sgn}}
```

```
81 \CustomizeMathJax{\DeclareMathOperator{\sinc}{sinc}}
82 \CustomizeMathJax{\DeclareMathOperator{\spa}{span}}
83 \CustomizeMathJax{\DeclareMathOperator{\tr}{tr}}
84 \CustomizeMathJax{\DeclareMathOperator{\var}{var}}
85 \CustomizeMathJax{\DeclareMathOperator{\Var}{Var}}
86 \CustomizeMathJax{\DeclareMathOperator{\Zu}{Z}}
87
88 \CustomizeMathJax{\DeclareMathOperator{\arccot}{arccot}}
89 \CustomizeMathJax{\DeclareMathOperator{\sech}{sech}}
90 \CustomizeMathJax{\DeclareMathOperator{\csch}{csch}}
91 \CustomizeMathJax{\DeclareMathOperator{\arsinh}{arsinh}}
92 \CustomizeMathJax{\DeclareMathOperator{\arcosh}{arcosh}}
93 \CustomizeMathJax{\DeclareMathOperator{\artanh}{artanh}}
94 \CustomizeMathJax{\DeclareMathOperator{\arcoth}{arcoth}}
95 \CustomizeMathJax{\DeclareMathOperator{\arsech}{arsech}}
96 \CustomizeMathJax{\DeclareMathOperator{\arcsch}{arcsch}}
97
98 \CustomizeMathJax{\DeclareMathOperator{\bigO}{\mathcal{O}}}
99 \CustomizeMathJax{\DeclareMathOperator{\bigo}{O}}
100 \CustomizeMathJax{\DeclareMathOperator{\lito}{o}}
101
102 \CustomizeMathJax{\newcommand{\R}{\mathset{R}}}
103 \CustomizeMathJax{\newcommand{\C}{\mathset{C}}}
104 \CustomizeMathJax{\newcommand{\N}{\mathset{N}}}
105 \CustomizeMathJax{\newcommand{\Z}{\mathset{Z}}}
106 \CustomizeMathJax{\newcommand{\Q}{\mathset{Q}}}
107 \CustomizeMathJax{\newcommand{\F}{\mathset{F}}}
108 \CustomizeMathJax{\newcommand{\K}{\mathset{K}}}
109
110 \CustomizeMathJax{\newcommand{\ds}{\displaystyle}}
111 \CustomizeMathJax{\newcommand{\dlim}{\lim\limits}}
112 \CustomizeMathJax{\newcommand{\dsum}{\sum\limits}}
113 \CustomizeMathJax{\newcommand{\dprod}{\prod\limits}}
114 \CustomizeMathJax{\newcommand{\dcup}{\bigcup\limits}}
115 \CustomizeMathJax{\newcommand{\dcap}{\bigcap\limits}}
116 \CustomizeMathJax{\newcommand{\lbar}{\overline}}
117 \CustomizeMathJax{\newcommand{\hlbar}[1]{\overline{\vphantom{h}#1}}}
118 \CustomizeMathJax{\newcommand{\LWReqdefstar}{\stackrel{\Delta}{=}}}
119 \CustomizeMathJax{\newcommand{\LWReqdefnostar}{\stackrel{\mathrm{def}}{=}}}
120 \CustomizeMathJax{\newcommand{\eqdef}{\ifstar\LWReqdefstar\LWReqdefnostar}}
121 \CustomizeMathJax{\newcommand{\unbr}{\underbrace}}
122 \CustomizeMathJax{\newcommand{\iif}{if and only if }}
123
124 \CustomizeMathJax{\newcommand{\mul}{\mathord{\times}}}
125 \CustomizeMathJax{\newcommand{\then}{\ \Longrightarrow \ \mbox{} }}
126 \CustomizeMathJax{\newcommand{\txt}[1]{\quad\text{#1}\quad}}
127 \CustomizeMathJax{\newcommand{\pow}[2]{\left( #1 \right)^{\!#2}}}
128 \CustomizeMathJax{\newcommand{\abs}[1]{\left\vert#1\right\vert}}
129 \CustomizeMathJax{\newcommand{\lfrac}[2]{\frac{\:#1\:}{\:#2\:}}}
130
131 \CustomizeMathJax{\newenvironment{system}[1][l]%
132     {\left\{\begin{array}{@{.15em}#1@{}}}
133     {\end{array}\right.}
134 }
135
136 \CustomizeMathJax{\newenvironment{spmatrix}
137     {\left(\begin{smallmatrix}}
138     {\end{smallmatrix}\right)}
139 }
140
```

```
141 \CustomizeMathJax{%
142     \newenvironment{mathcols}
143         {\begin{aligned}\displaystyle}
144         {\end{aligned}}
145 }
146 \CustomizeMathJax{\newcommand{\changecol}{\end{aligned}\qquad\begin{aligned}}}
```

User-adjustable settings, detected if in the preamble.

```
147 \AtBeginDocument{
148 \ifdef{\itpi}{
149     \CustomizeMathJax{\let\itpi\pi}
150 }{}
151 \ifdefstring{\boldvectcommand}{\mathbf}{
152     \CustomizeMathJax{\newcommand{\boldvectcommand}[1]{\mathbf{#1}}}
153 }{
154     \CustomizeMathJax{\newcommand{\boldvectcommand}[1]{\boldsymbol{#1}}}
155 }
156 \ifbool{arrowvect}{
157     \CustomizeMathJax{\newcommand{\vect}[1]{\overrightarrow{#1}}}
158 }{
159     \CustomizeMathJax{\newcommand{\vect}[1]{\boldvectcommand{#1}}}
160 }
161 \ifdefstring{\probastyle}{\mathbb}{
162     \CustomizeMathJax{\newcommand{\probastyle}[1]{\mathbb{#1}}}
163 }{
164     \CustomizeMathJax{\newcommand{\probastyle}[1]{\mathrm{#1}}}
165 }
166 \ifdefstring{\mathset}{\mathbb}{
167     \CustomizeMathJax{\newcommand{\mathset}[1]{\mathbb{#1}}}
168 }{
169     \CustomizeMathJax{\newcommand{\mathset}[1]{\mathbf{#1}}}
170 }
171 }
172 \end{warpMathJax}
```

File 313   **`lwarp-mleftright.sty`**

§ 425   Package   # mleftright

(*Emulates or patches code by* Heiko Oberdiek.)

mleftright (*Pkg*)   mleftright is used as-is, and is emulated for MathJax.

for HTML output: 1 \LWR@ProvidesPackagePass{mleftright}[2019/12/03]

```
2 \begin{warpMathJax}
3 \CustomizeMathJax{\newcommand{\mleft}{\left}}
4 \CustomizeMathJax{\newcommand{\mright}{\right}}
5 \CustomizeMathJax{\newcommand{\mleftright}{}}
6 \CustomizeMathJax{\newcommand{\mleftrightrestore}{}}
7 \end{warpMathJax}
```

File 314 **`lwarp-morefloats.sty`**

§ 426 Package **morefloats**

morefloats (*Pkg*) **morefloats** is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{morefloats}[2015/07/22]

File 315 **`lwarp-moreverb.sty`**

§ 427 Package **moreverb**

(*Emulates or patches code by* ROBIN FAIRBAIRNS.)

moreverb (*Pkg*) **moreverb** is supported with some patches.

```
1 \LWR@ProvidesPackagePass{moreverb}[2008/06/03]

2 \BeforeBeginEnvironment{verbatimtab}{%
3 \LWR@forcenewpage
4 \LWR@atbeginverbatim{Verbatim}%
5 }
6 \AfterEndEnvironment{verbatimtab}{%
7 \LWR@afterendverbatim%
8 }
9
10
11 \LetLtxMacro\LWRMV@orig@verbatimtabinput\@verbatimtabinput
12
13 \renewcommand{\@verbatimtabinput}[2][]{%
14 \LWR@forcenewpage
15 \LWR@atbeginverbatim{Verbatim}%
16 \LWRMV@orig@verbatimtabinput[#1]{#2}%
17 \LWR@afterendverbatim%
18 }
19
20 \BeforeBeginEnvironment{listing}{%
21 \LWR@forcenewpage
22 \LWR@atbeginverbatim{programlisting}%
23 }
24
25 \AfterEndEnvironment{listing}{%
26 \LWR@afterendverbatim%
27 }
28
29 \BeforeBeginEnvironment{listingcont}{%
30 \LWR@forcenewpage
31 \LWR@atbeginverbatim{programlisting}%
32 }
33
34 \AfterEndEnvironment{listingcont}{%
35 \LWR@afterendverbatim%
36 }
```

```
37 \LetLtxMacro\LWRMV@@listinginput\@listinginput
38
39 \renewcommand{\@listinginput}[3][]{
40 \LWR@forcenewpage
41 \LWR@atbeginverbatim{programlisting}%
42 \LWRMV@@listinginput[#1]{#2}{#3}%
43 \LWR@afterendverbatim%
44 }
45
46
47 \renewenvironment*{boxedverbatim}
48 {
49 \LWR@forcenewpage
50 \LWR@atbeginverbatim{boxedverbatim}%
51 \verbatim%
52 }
53 {
54 \endverbatim%
55 \LWR@afterendverbatim%
56 }
```

File 316 **`lwarp-movie15.sty`**

§ 428   Package # movie15

movie15 (*Pkg*)   movie15 is emualted.

The packages multimedia, movie15, and media9 are supported.

HTML5 <audio> and <video> objects are created for .mp3 and .mp4 files.

HTML5 <embed> objects are created for http and ftp links.

\href links are created for other media types. (Unfortunately, there is not much overlap between the file types supported for print output and the file types supported by HTML5.)

For media9, a multimedia object is inserted for each addresource=, as well as each flashvars source= and src=. This may result in duplicate objects.

Undesired objects may be nullified by placing them inside \warpprintonly or the warpprint environment.

Each HTML multimedia object includes the poster text, except for <embed> objects. For movie15, the text option is supported to specify the poster text.

The width, height, and totalheight options are supported. The HTML object is scaled according to the display width, correctly compensating for either tall or wide viewports.

Other options are ignored.

media9 \addmediapath is supported. It is assumed that the same path structure will exist for the HTML document.

HTML5 media controls are always specified for each <audio> and <video> object.

media9 slideshows are not supported.

\hyperlinkmovie, \movieref, and \mediabutton are not supported.

3D objects are not supported.

If using a YouTube™ video, use an "embedded" URL with .../embed/... instead of .../v/...

for HTML output:

```
1 \LWR@ProvidesPackageDrop{movie15}[2012/05/16]

2 \LWR@origRequirePackage{lwarp-common-multimedia}
3
4 \RequirePackage{xkeyval}
5
6 \newcommand*{\LWR@moviefifteen@text}{}
7
8 \define@key{LWR@moviefifteen}{text}{\renewcommand{\LWR@moviefifteen@text}{#1}}
9
10 \newcommand*{\LWR@includemovieb}[4][]{%
11     \renewcommand{\LWR@moviefifteen@text}{(multimedia)}
12     \setkeys*{LWR@moviefifteen}{#1}%
13     \LWR@multimediab[#1,width=#2,height=#3]{\LWR@moviefifteen@text}{#4}%
14 }
15
16 \newrobustcmd*{\includemovie}{%
17     \begingroup%
18     \LWR@linkmediacatcodes%
19     \LWR@includemovieb%
20 }
21
22
23 \newcommand*{\movieref}[3][]{}
24
25 \LetLtxMacro\movie\LWR@multimedia
26 % \LetLtxMacro\sound\LWR@multimedia% not in media15
27
28 \newcommand{\hyperlinkmovie}[3][]{}
```

File 317 **`lwarp-mparhack.sty`**

§ 429   Package **mparhack**

mparhack (*Pkg*)   mparhack is ignored.

for HTML output: Discard all options for lwarp-mparhack:

```
1 \LWR@ProvidesPackageDrop{mparhack}[2005/04/17]
```

File 318   **`lwarp-multibib.sty`**

§ 430   Package **multibib**

(*Emulates or patches code by* THORSTEN HANSEN.)

multibib (*Pkg*)   multibib is patched for use by lwarp.

for HTML output:

```
1 \LWR@ProvidesPackagePass{multibib}[2008/12/10]
```

```
2 \VerifyCommand[lwarp][multibib]{\newcites}{77893319F9935670F2FF2E524075CB71}
3
4 \xpatchcmd\newcites
5     {{\@suffix}}
6     {{\@suffix_html}}
7     {}
8     {\LWR@patcherror{multibib}{newcites}}
```

File 319   **`lwarp-multicap.sty`**

§ 431   Package   # multicap

multicap (*Pkg*)   multicap is emualted.

for HTML output: `1 \LWR@ProvidesPackageDrop{multicap}[2002/05/04]`

```
2 \newcommand*{\mfcaption}{\captionof{figure}}
3 \newcommand*{\mtcaption}{\captionof{table}}
4 \newcounter{mcapsize}
5 \newcounter{mcapskip}
6 \newlength{\abvmcapskip}
7 \newlength{\blwmcapskip}
```

File 320   **`lwarp-multicol.sty`**

§ 432   Package   # multicol

(*Emulates or patches code by* Frank Mittelbach.)

multicol (*Pkg*)   multicol is emulated.

for HTML output: `1 \LWR@ProvidesPackageDrop{multicol}[2021/10/28]`

Multicols are converted into a 1–3 column display, browser-supported.

The optional multicols heading is placed inside a <div> of class multicolsheading.

The content is placed inside a <div> of class multicols.

Env   multicols   * {⟨*numcols*⟩} [⟨*heading*⟩]

```
2 \NewDocumentEnvironment{multicols}{s m o}
```

HTML <div> class to contain everything:

```
3 {
4     \LWR@forcenewpage
5     \BlockClass{multicols}
```

Optional HTML <div> class for the heading:

```
6     \IfValueT{#3}{\begin{BlockClass}{multicolsheading}#3\end{BlockClass}}%
```

Change \linewidth to compensate for expected size:

```
7    \setlength{\linewidth}{\linewidth/#2}
```

Locally force any minipages to be fullwidth:

```
8    \booltrue{LWR@forceminipagefullwidth}
9 }
```

When done with the environment, close the <div>:

```
10 {\endBlockClass}
```

Emulated null functions which are not used in HTML:

```
11 \newcommand*{\columnbreak}{}
12 \newcommand*{\newcolumn}{}
13 \newcommand*{\RLmulticolcolumns}{}
14 \newcommand*{\LRmulticolcolumns}{}
15
16 \newlength{\premulticols}
17 \newlength{\postmulticols}
18 \newlength{\multicolsep}
19 \newlength{\multicolbaselineskip}
20 \newlength{\multicoltolerance}
21 \newlength{\multicolpretolerance}
22 \newcommand*{\columnseprulecolor}{\normalcolor}
23 \newcounter{columnbadness}
24 \newcounter{finalcolumnbadness}
25 \newcounter{collectmore}
26 \newcounter{unbalance}
27 \newlength{\multicolovershoot}
28 \newlength{\multicolundershoot}


29 \NewDocumentCommand{\docolaction}{s o m m m}{%
30    \IfValueTF{#2}{#2}{#3}%
31 }
```

File 321   `lwarp-multicolrule.sty`

§ 433   Package   # multicolrule

multicolrule (*Pkg*)   multicolrule is ignored.

for HTML output:

```
1 \RequirePackage{multicol}
2
3 \LWR@ProvidesPackageDrop{multicolrule}[2019/01/01]

4 \newcommand*{\SetMCRule}[1]{}
5 \NewDocumentCommand{\DeclareMCRulePattern}{m m}{}
```

File 322   `lwarp-multimedia.sty`

§ 434   Package   # multimedia

multimedia (*Pkg*)   multimedia is emulated.

The packages multimedia, movie15, and media9 are supported.

HTML5 <audio> and <video> objects are created for .mp3 and .mp4 files.

HTML5 <embed> objects are created for http and ftp links.

\href links are created for other media types. (Unfortunately, there is not much overlap between the file types supported for print output and the file types supported by HTML5.)

For media9, a multimedia object is inserted for each addresource=, as well as each flashvars source= and src=. This may result in duplicate objects.

Undesired objects may be nullified by placing them inside \warpprintonly or the warpprint environment.

Each HTML multimedia object includes the poster text, except for <embed> objects. For movie15, the text option is supported to specify the poster text.

The width, height, and totalheight options are supported. The HTML object is scaled according to the display width, correctly compensating for either tall or wide viewports.

Other options are ignored.

media9 \addmediapath is supported. It is assumed that the same path structure will exist for the HTML document.

HTML5 media controls are always specified for each <audio> and <video> object.

media9 slideshows are not supported.

\hyperlinkmovie, \movieref, and \mediabutton are not supported.

3D objects are not supported.

If using a YOUTUBE™ video, use an "embedded" URL with …/embed/… instead of …/v/…

```
1 \LWR@ProvidesPackageDrop{multimedia}[2012/05/02]

2 \LWR@origRequirePackage{lwarp-common-multimedia}
3
4 \LetLtxMacro\movie\LWR@multimedia
5 \LetLtxMacro\sound\LWR@multimedia
6
7 \newcommand{\hyperlinkmovie}[3][]{}
8
9 \newcommand{\hyperlinksound}[3][]{}
10
11 \newcommand{\hyperlinkmute}
```

File 323 **lwarp-multiobjective.sty**

§ 435   Package **multiobjective**

(*Emulates or patches code by* LUIS MARTÍ.)

multiobjective (*Pkg*)   multiobjective is used as-is for svg math, and is emulated for MathJax.

1 \LWR@ProvidesPackagePass{multiobjective}[2008/08/19]

2 \begin{warpMathJax}
3 \CustomizeMathJax{\newcommand{\dom}{\prec}}
4 \CustomizeMathJax{\newcommand{\negdom}{\not\prec}}
5 \CustomizeMathJax{\newcommand{\weakdom}{\preccurlyeq}}
6 \CustomizeMathJax{\newcommand{\negweakdom}{\not\preccurlyeq}}

7 \CustomizeMathJax{\newcommand{\strictdom}{\mathord{\prec}\!\!\!\mathord{\prec}}}
8 \CustomizeMathJax{\newcommand{\negstrictdom}{\mathord{\not\prec}\!\!\!\mathord{\prec}}}
9 \CustomizeMathJax{\newcommand{\multepsilondom}{\preccurlyeq_{\epsilon\cdot}}}
10 \CustomizeMathJax{\newcommand{\addiepsilondom}{\preccurlyeq_{\epsilon +}}}
11 \CustomizeMathJax{\newcommand{\better}{\triangleleft}}
12 \CustomizeMathJax{\def\vec#1{%
13   \mathchoice%
14   {{\displaystyle\boldsymbol{#1}}}%
15   {{\textstyle\boldsymbol{#1}}}%
16   {{\scriptstyle\boldsymbol{#1}}}%
17   {{\scriptscriptstyle\boldsymbol{#1}}}%
18 }}
19
20 \CustomizeMathJax{\newcommand{\set}[1]{%
21   \mathchoice%
22   {{\displaystyle\mathcal{#1}}}%
23   {{\textstyle\mathcal{#1}}}%
24   {{\scriptstyle\mathcal{#1}}}%
25   {{\scriptscriptstyle\mathcal{#1}}}%
26 }}
27 \CustomizeMathJax{\def\argmax{\mathop{{\mathrm{arg}}\,\max}}}
28 \CustomizeMathJax{\def\argmin{\mathop{{\mathrm{arg}}\,\min}}
29 }}
30 \end{warpMathJax}

File 324   **`lwarp-multirow.sty`**

§ 436   Package   # multirow

(*Emulates or patches code by* PIET VAN OOSTRUM, ØYSTEIN BACHE, JERRY LEICHTER.)

multirow (*Pkg*)   multirow is emulated during HTML output, and used as-is while inside a lateximage.

vposn   • Note that recent versions of multirow include a new optional vposn argument.

multirow cells   • For multirow, insert \mrowcell into any empty multi-row cells. This will be a null function for the print output, and is a placeholder for parsing the table for HTML output. An error is generated if this is missed.

```
. . . &   \multirow{2}{.5in}{text}   & . . .
. . . &   \mrowcell                  & . . .
```

colored cells   • The multirow documentation regarding colored cells recommends using a negative number of rows. This will not work with lwarp, so \warpprintonly and \warpHTMLonly must be used to make versions for print and HTML.

with \multicolumn

⚠ \multicolumn & \multirow

⚠ skipped cells

⚠ empty cells

⚠ MathJax

- See section 436.2 for \multicolumrow.

lwarp does not support directly combining \multicolumn and \multirow. Use \multicolumnrow instead. To create a 2 column, 3 row cell:

> \multicolumnrow{2}{c}[c]{3}[0]{1in}[0pt]{Text}

The two arguments for \multicolumn come first, followed by the five arguments for \multirow, many of which are optional, followed by the contents.

As per \multirow, skipped cells to the right of the \multicolumnrow statement are not included in the source code on the same line. On the following lines, \mcolrowcell must be used for each cell of each column and each row to be skipped. An error is generated if this is missed.

> ... & \multicolumnrow{2}{c}[c]{3}[0]{1in}[0pt]{Text} & ...
>
> ... & \mcolrowcell   &   \mcolrowcell & ...
>
> ... & \mcolrowcell   &   \mcolrowcell & ...

- MATHJAX does not support **multirow**, so it is emulated to only print its text on the first row. \multirow works as expected in text tabulars or svg math.

In a lateximage, the print versions are restored.

See section 77.24 for the print-mode versions.

for HTML output: Remove the placeholder macro which was used if **multirow** was not loaded:

```
1 \LetLtxMacro\multirow\relax
```

```
2 \LWR@ProvidesPackagePass{multirow}[2021/03/15]
```

\LWR@multirowborder Set to left or right to create a thick border for the cell, for use by **bigdelim**:

```
3 \newcommand{\LWR@multirowborder}{}
```

§ 436.1 **Multirow**

\LWR@multirow@par \par inside a \multirow.

```
4 \newcommand*{\LWR@multirow@par}{%
5     \LWR@htmltag{br /}%
6 }%
```

\multirow [⟨1: vpos⟩] {⟨2: numrows⟩} [⟨3:bigstruts⟩] {⟨4: width⟩} [⟨5: vmove⟩] {⟨6: text⟩}

```
7 \NewDocumentCommand{\LWR@HTML@multirow}{O{c} m o m o +m}%
8 {%
9     \LWR@traceinfo{LWR@HTML@multirow #1 #2 #4}%

10    \booltrue{LWR@usedmultirow}%

11    \LWR@maybenewtablerow%
12    \LWR@tabularleftedge%
```

Print the start of a new table data cell:

```
13    \LWR@htmltag{%
14        td rowspan=\textquotedbl#2\textquotedbl\ %
```

A class adds the column spec and the rule:

```
15        class=\textquotedbl{}td%
```

Append this column's spec:

```
16        \LWR@getexparray{LWR@tablecolspec}{\arabic{LWR@tableLaTeXcolindex}}%
```

If this column has a cmidrule, add "rule" to the end of the HTML class tag. Also add the vertical bar class.

```
17        \LWR@addcmidruletrim%
18        \LWR@addleftmostbartag%
19        \LWR@printbartag{\arabic{LWR@tableLaTeXcolindex}}%
20        \textquotedbl%


21        \LWR@tdstartstyles%
```

The vertical alignment, if given:

```
22        \ifstrequal{#1}{c}{\LWR@tdaddstyle\LWR@print@mbox{vertical-align:middle}}{}%
23        \ifstrequal{#1}{b}{\LWR@tdaddstyle\LWR@print@mbox{vertical-align:bottom}}{}%
24        \ifstrequal{#1}{t}{\LWR@tdaddstyle\LWR@print@mbox{vertical-align:top}}{}%
```

The left/right border, if given:

```
25        \ifdefvoid{\LWR@multirowborder}{}{%
26          \LWR@tdaddstyle%
27         \LWR@print@mbox{border-\LWR@multirowborder:} 2px dotted black ; %
28          \LWR@print@mbox{padding-\LWR@multirowborder:} 2px%
29        }%
```

Additional style elements:

```
30        \LWR@addcmidrulewidth%
31        \LWR@addcdashline%
32        \LWR@addtabularrulecolors%
33        \LWR@tdendstyles%
34     }%
```

The column's < spec:

```
35     \LWR@getexparray{LWR@colbeforespec}{\arabic{LWR@tableLaTeXcolindex}}%
```

While printing the text, redefine \\ to generate a new line. If a nested tabular occurs, \\ is redefined to \LWR@tabularendofline at the start of the tabular, then \LWR@endofline before again printing any \multirow contents inside the nested tabular.

\par is redefined to insert an HTML break, and if tabular is nested, it is redefined at the start of tabular.

```
36     \begingroup%
37        \LetLtxMacro{\\}{\LWR@endofline}%
38        \booltrue{LWR@in@multirow@par}%
39        #6%
40     \endgroup%
```

```
41        \LWR@stoppars%
42        \boolfalse{LWR@intabularmetadata}%
43        \renewcommand{\LWR@multirowborder}{}%
44        \LWR@traceinfo{LWR@HTML@multirow done}%
45 }%
46
47 \LWR@formatted{multirow}
```

§ 436.2 **Combined multicolumn and multirow**

\multicolumnrow {⟨*1:cols*⟩} {⟨*2:halign*⟩} [⟨*3:vpos*⟩] {⟨*4:numrows*⟩} [⟨*5:bigstruts*⟩] {⟨*6:width*⟩} [⟨*7:fixup*⟩]
{⟨*8:text*⟩}

\IfPackageLoadedTF{multirow} determines if v2.0 or later of multirow was used,
which included the \ProvidesPackage macro.

The HTML version follows.

\AtBeginDocument because the print version had to see if multirow was loaded
before determining how to define \LWR@print@multicolumnrow.

```
48 \AtBeginDocument{
49
50 \NewExpandableDocumentCommand{\LWR@HTML@multicolumnrow}{m m O{} m O{} m O{} +m}{%
```

```
51 \booltrue{LWR@usedmultirow}%
```

Figure out how many extra HTML columns to add for @ and ! columns:

```
52 \LWR@tabularhtmlcolumns{\arabic{LWR@tableLaTeXcolindex}}{#1}
```

Create the multicolumn/multirow tag, temporarily redefining the end of line. (Us-
ing a group caused problems with a nested tabular.

```
53 \LetLtxMacro{\\}{\LWR@endofline}%
54 \LWR@domulticolumn[#3][#4]{#1}{\arabic{LWR@tabhtmlcoltotal}}{#2}{#8}%
55 \LetLtxMacro{\\}{\LWR@tabularendofline}%
```

Move to the next LaTeX column:

```
56 \defaddtocounter{LWR@tableLaTeXcolindex}{#1}%
57 \defaddtocounter{LWR@tableLaTeXcolindex}{-1}%
```

Skip any trailing @ or ! columns for this cell:

```
58 \booltrue{LWR@skipatbang}%
59 }
60
61 \LWR@expandableformatted{multicolumnrow}
62
63 }% \AtBeginDocument
```

For MATHJAX. Only the text is used. All other parameters are ignored.

```
64 \begin{warpMathJax}
65 % \multirow[vpos]{num}[bigstruts]{width}[vmove]{text}
66 \CustomizeMathJax{\newcommand{\LWRsubmultirow}[2][]{#2}}
```

```
67 \CustomizeMathJax{\newcommand{\LWRmultirow}[2][]{\LWRsubmultirow}}
68 \CustomizeMathJax{\newcommand{\multirow}[2][]{\LWRmultirow}}
69 %
70 \CustomizeMathJax{\newcommand{\mrowcell}{}}
71 \CustomizeMathJax{\newcommand{\mcolrowcell}{}}
72 \CustomizeMathJax{\newcommand{\STneed}[1]{}}
73 \end{warpMathJax}
```

File 325   **`lwarp-multitoc.sty`**

§ 437   Package   # multitoc

multitoc (*Pkg*)   multitoc is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{multitoc}[1999/06/08]`

```
2 \newcommand{\multicolumntoc}{2}
3 \newcommand{\multicolumnlot}{2}
4 \newcommand{\multicolumnlof}{2}
5 \newcommand*{\immediateaddtocontents}[2]{}
```

File 326   **`lwarp-musicography.sty`**

§ 438   Package   # musicography

(*Emulates or patches code by* ANDREW A. CASHNER.)

musicography (*Pkg*)   musicography is patched for use by lwarp.

Images are used for the meter symbols and fingered bass, since the HTML fonts tend not to be the correct size and HTML cannot stack items. The HTML alt tag copies C and 3/2, etc. Hashes are used for the meter images, which are then reused as necessary.

⚠   Note that browser support for musical symbols may be buggy. ALT text and copy/paste into a text editor work well.

for HTML output: `1 \LWR@ProvidesPackagePass{musicography}[2023/09/08]`

```
2 \NewDocumentCommand{\LWR@HTML@musSymbol}{ O{\musFont} m m m m }{%
3 \begin{lateximage}%
4 {#1\kern#2\raisebox{#3}{#5}\kern#4}%
5 \end{lateximage}%
6 }
7
8 \LWR@formatted{musSymbol}
9
10 \NewDocumentCommand{\LWR@HTML@musStemmedNote}{ m }{%
11 \begin{lateximage}%
12 \musSymbol{0.05em}{0.5ex}{0.2em}{#1\musStem}%
13 \end{lateximage}%
14 }
15
16 \LWR@formatted{musStemmedNote}
```

```
17
18 \NewDocumentCommand{\LWR@HTML@musFlaggedNote}{ m m }{%
19 \begin{lateximage}%
20 \musSymbol{0.05em}{0.5ex}{0pt}{#1\musStem}%
21 \musSymbol{0pt}{0pt}{0.9em}{#2}%
22 \end{lateximage}%
23 }
24
25 \LWR@formatted{musFlaggedNote}
26
27 \NewDocumentCommand{\LWR@HTML@musDottedNote}{ m }{%
28 \begin{lateximage}%
29 #1\musDot%
30 \end{lateximage}%
31 }
32
33 \LWR@formatted{musDottedNote}
34
35 \NewDocumentCommand{\LWR@HTML@musMeter}{ O{\musNumFont} m m }{%
36 \begin{lateximage}*[#2/#3]*?[#1#2#3]%
37 \musStack{#2 #3}\kern0.05em%
38 \end{lateximage}%
39 }
40
41 \LWR@formatted{musMeter}
42
43 \NewDocumentCommand{\LWR@HTML@meterCplus}{ m }{%
44 \begin{lateximage}*[C#1]*?%
45     \meterC{}\kern-0.7pt#1%
46 \end{lateximage}%
47 }
48
49 \LWR@formatted{meterCplus}
50
51 \NewDocumentCommand{\LWR@HTML@meterC}{}{%
52 \begin{lateximage}*[C]*%
53 \musSymbolMeter{\symbol{83}}%
54 \end{lateximage}%
55 }
56
57 \LWR@formatted{meterC}
58
59 \NewDocumentCommand{\LWR@HTML@meterCutC}{}{%
60 \begin{lateximage}*[C|]*%
61 \musSymbolMeter{\symbol{82}}%
62 \end{lateximage}%
63 }
64
65 \LWR@formatted{meterCutC}
66
67 \NewDocumentCommand{\LWR@HTML@meterThree}{}{%
68 \begin{lateximage}*[3|]*%
69     {\musNumFont{3}}%
70 \end{lateximage}%
71 }
72
73 \LWR@formatted{meterThree}
74
75 \NewDocumentCommand{\LWR@HTML@meterThreeTwo}{}{%
76 \begin{lateximage}*[3/2]*%
```

```
77        \raisebox{-0.26ex}{\musMeter{3}{2}}%
78 \end{lateximage}%
79 }
80
81 \LWR@formatted{meterThreeTwo}
82
83 \NewDocumentCommand{\LWR@HTML@meterCutCThree}{}{%
84 \begin{lateximage}*[C|3]*%
85        \meterCutCplus{\meterThree}%
86 \end{lateximage}%
87 }
88
89 \LWR@formatted{meterCutCThree}
90
91 \NewDocumentCommand{\LWR@HTML@meterCutCThreeTwo}{}{%
92 \begin{lateximage}*[C|3/2]*%
93        \meterCutCplus{\meterThreeTwo}%
94 \end{lateximage}%
95 }
96
97 \LWR@formatted{meterCutCThreeTwo}
98
99 \NewDocumentCommand{\LWR@HTML@meterCThree}{}{%
100 \begin{lateximage}*[C3]*%
101        \meterCplus{\meterThree}%
102 \end{lateximage}%
103 }
104
105 \LWR@formatted{meterCThree}
106
107
108 \NewDocumentCommand{\LWR@HTML@meterCThreeTwo}{}{%
109 \begin{lateximage}*[C3/2]*%
110 \meterCplus{\musStack{3 2}}%
111 \end{lateximage}%
112 }
113
114 \LWR@formatted{meterCThreeTwo}
115
116 \NewDocumentCommand{\LWR@HTML@meterCZ}{}{%
117 \begin{lateximage}*[CZ]*%
118        \meterCplus{\meterZsymbol}%
119 \end{lateximage}%
120 }
121
122 \LWR@formatted{meterCZ}
123
124 \NewDocumentCommand{\LWR@HTML@meterO}{}{\HTMLunicode{25EF}}
125
126 \LWR@formatted{meterO}
127
128
129 \NewDocumentCommand{\LWR@HTML@noFig}{ O{5} }{}%
130 \newcommand*{\LWR@HTML@musFlat}        {\HTMLunicode{266D}}
131 \newcommand*{\LWR@HTML@musDoubleFlat} {\HTMLunicode{1D12B}}
132 \newcommand*{\LWR@HTML@musSharp}       {\HTMLunicode{266F}}
133 \newcommand*{\LWR@HTML@musDoubleSharp}{\HTMLunicode{1D12A}}
134 \newcommand*{\LWR@HTML@musNatural}     {\HTMLunicode{266E}}
135
136 \LWR@formatted{noFig}
```

```
137 \LWR@formatted{musFlat}
138 \LWR@formatted{musDoubleFlat}
139 \LWR@formatted{musSharp}
140 \LWR@formatted{musDoubleSharp}
141 \LWR@formatted{musNatural}
142
143
144 \NewDocumentCommand{\LWR@HTML@musFig}{ O{r} m }{%
145 \begin{lateximage}*[%
146     {% ALT text for copy/paste
147         \RenewDocumentCommand{\noFig}{O{5}}{\LWR@HTML@noFig}%
148         \RenewDocumentCommand{\musSharp}{}{\LWR@HTML@musSharp}%
149         \RenewDocumentCommand{\musDoubleSharp}{}{\LWR@HTML@musDoubleSharp}%
150         \RenewDocumentCommand{\musFlat}{}{\LWR@HTML@musFlat}%
151         \RenewDocumentCommand{\musDoubleFlat}{}{\LWR@HTML@musDoubleFlat}%
152         \RenewDocumentCommand{\musNatural}{}{\LWR@HTML@musNatural}%
153         {#2}% braces here because \noFig uses []
154     }%
155 ]*?%
156     {\musFigSize\musStack[\musFigFont]{#2}[#1]}%
157 \end{lateximage}%
158 }
159
160 \LWR@formatted{musFig}
161
162
163 \NewDocumentCommand{\LWR@HTML@musWhole}        {}{\HTMLunicode{1D15D}}
164 \NewDocumentCommand{\LWR@HTML@musHalf}         {}{\HTMLunicode{1D15E}}
165 \NewDocumentCommand{\LWR@HTML@musQuarter}      {}{\HTMLunicode{1D15F}}
166 \NewDocumentCommand{\LWR@HTML@musEighth}       {}{\HTMLunicode{1D160}}
167 \NewDocumentCommand{\LWR@HTML@musSixteenth}    {}{\HTMLunicode{1D161}}
168 \NewDocumentCommand{\LWR@HTML@musThirtySecond} {}{\HTMLunicode{1D162}}
169 \NewDocumentCommand{\LWR@HTML@musSixtyFourth}  {}{\HTMLunicode{1D163}}
170
171 \LWR@formatted{musWhole}
172 \LWR@formatted{musHalf}
173 \LWR@formatted{musQuarter}
174 \LWR@formatted{musEighth}
175 \LWR@formatted{musSixteenth}
176 \LWR@formatted{musThirtySecond}
177 \LWR@formatted{musSixtyFourth}
178
179 \NewDocumentCommand{\LWR@HTML@musWholeDotted}{}
180     {\HTMLunicode{1D15D}\HTMLunicode{1D16D}}
181 \NewDocumentCommand{\LWR@HTML@musHalfDotted}{}
182     {\HTMLunicode{1D15E}\HTMLunicode{1D16D}}
183 \NewDocumentCommand{\LWR@HTML@musQuarterDotted}{}
184     {\HTMLunicode{1D15F}\HTMLunicode{1D16D}}
185 \NewDocumentCommand{\LWR@HTML@musEighthDotted}{}
186     {\HTMLunicode{1D160}\HTMLunicode{1D16D}}
187 \NewDocumentCommand{\LWR@HTML@musSixteenthDotted}{}
188     {\HTMLunicode{1D161}\HTMLunicode{1D16D}}
189 \NewDocumentCommand{\LWR@HTML@musThirtySecondDotted}{}
190     {\HTMLunicode{1D162}\HTMLunicode{1D16D}}
191 \NewDocumentCommand{\LWR@HTML@musSixtyFourthDotted}{}
192     {\HTMLunicode{1D163}\HTMLunicode{1D16D}}
193
194 \LWR@formatted{musWholeDotted}
195 \LWR@formatted{musHalfDotted}
196 \LWR@formatted{musQuarterDotted}
```

```
197 \LWR@formatted{musEighthDotted}
198 \LWR@formatted{musSixteenthDotted}
199 \LWR@formatted{musThirtySecondDotted}
200 \LWR@formatted{musSixtyFourthDotted}
```

File 327   **`lwarp-mwe.sty`**

§ 439   Package   # mwe

(*Emulates or patches code by* Martin Scharrer.)

mwe (*Pkg*)   mwe is used as-is, but a warning is issued to copy the images to the local directory.

for HTML output: `1 \LWR@ProvidesPackagePass{mwe}[2018/03/30]`

```
2 \AtEndDocument{%
3     \PackageWarningNoLine{lwarp}{%
4         For package mwe, copy any mwe images to be used for\MessageBreak
5         HTML, such as PNG or JPG, to the document's base\MessageBreak
6         directory.  Neither a subdirectory nor the mwe\MessageBreak
7         directory will work, due to the TeX file search\MessageBreak
8         algorithm%
9     }%
10 }%
```

File 328   **`lwarp-nameauth.sty`**

§ 440   Package   # nameauth

(*Emulates or patches code by* Charles P. Schaum.)

nameauth (*Pkg*)   nameauth is patched for use by lwarp.

for HTML output: `1 \LWR@ProvidesPackagePass{nameauth}[2023/02/03]`

\@nameauth@Hook (*Hook*)   lwarp formatting is inserted.
[nameauth]

```
2 \VerifyCommand[lwarp][nameauth]{\@nameauth@Hook}{E665BBD1C138AA37AF2AF5E3C3565584}
3
4 \renewcommand*\@nameauth@Hook[1]
5 {%
6   \ifdefined\@nameauth@InParser
7     \@nameauth@InHooktrue%
8     \protected@edef\test{#1}%
9     \expandafter\@nameauth@TestDot\expandafter{\test}%
10    \if@nameauth@MainFormat
11      \if@nameauth@FirstFormat
12        \bgroup\NamesFormat{%
13            \LWR@textcurrentcolor{\LWR@textcurrentfont{#1}}%    lwarp
14        }\egroup%
15      \else
16        \bgroup\MainNameHook{%
17            \LWR@textcurrentcolor{\LWR@textcurrentfont{#1}}%    lwarp
18        }\egroup%
```

```
19      \fi
20    \else
21      \if@nameauth@FirstFormat
22        \bgroup\FrontNamesFormat{%
23            \LWR@textcurrentcolor{\LWR@textcurrentfont{#1}}%       lwarp
24        }\egroup%
25      \else
26        \bgroup\FrontNameHook{%
27            \LWR@textcurrentcolor{\LWR@textcurrentfont{#1}}%       lwarp
28        }\egroup%
29      \fi
30    \fi
31  \fi
32 }
```

File 329   **`lwarp-nameref.sty`**

§ 441   Package   # nameref

nameref (*Pkg*)   nameref is nullified here, then emulated by lwarp.

for HTML output: Discard all options for lwarp-nameref:

```
1 \LWR@ProvidesPackageDrop{nameref}[2023-08-07]
```

File 330   **`lwarp-natbib.sty`**

§ 442   Package   # natbib

(*Emulates or patches code by* PATRICK W. DALY.)

natbib (*Pkg*)   natbib is patched for use by lwarp.

for HTML output: `1 \LWR@ProvidesPackagePass{natbib}[2010/09/13]`

Replace math < and > with \textless and \textgreater:

A macro to compare:

```
2 \newcommand{\LWRNB@NAT@open}{$<$}
```

To patch \NAT@open and \NAT@close

```
3 \newcommand{\LWRNB@patchnatbibopenclose}{
4 \ifdefstrequal{\NAT@open}{\LWRNB@NAT@open}
5 {
6     \renewcommand{\NAT@open}{\textless}
7     \renewcommand{\NAT@close}{\textgreater}
8 }{}
9 }
```

Do it now in case angle was selected as an option:

```
10 \LWRNB@patchnatbibopenclose
```

Also patch \setcitestyle to patch after settings are made:

```
11 \let\LWRNB@origsetcitestyle\setcitestyle
12
13 \renewcommand{\setcitestyle}[1]{%
14 \LWRNB@origsetcitestyle{#1}%
15 \LWRNB@patchnatbibopenclose%
16 }
```

Syncronize the autopage labels:

```
17 \xpretocmd{\NAT@reset@parser}
18    {\LWR@newautopagelabel{page}}%
19    {}
20    {\LWR@patcherror{natbib}{NAT@reset@parser}}
```

File 331   **lwarp-nccfancyhdr.sty**

§ 443   Package   # nccfancyhdr

(*Emulates or patches code by* ALEXANDER I. ROZHENKO.)

nccfancyhdr (*Pkg*)   nccfancyhdr is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{nccfancyhdr}[2004/12/07]

```
2 \newcommand*{\headrulewidth}{}
3 \newcommand*{\footrulewidth}{}
4 \newcommand{\headstrutheight}{}
5 \newcommand{\footstrutheight}{}
6 \newcommand*{\headrule}{}
7 \newcommand*{\footrule}{}
8
9 \newdimen\headwidth
10 \newcommand*{\extendedheaders}{}
11 \newcommand*{\normalheaders}{}
12
13 \newcommand*{\fancyhead}[2][]{}
14 \newcommand*{\fancyfoot}[2][]{}
15 \newcommand*{\fancyhf}[2][]{}
16 \newcommand*{\fancypagestyle}[2]{}
17 \newcommand*{\lhead}[2][]{}
18 \newcommand*{\chead}[2][]{}
19 \newcommand*{\rhead}[2][]{}
20 \newcommand*{\lfoot}[2][]{}
21 \newcommand*{\cfoot}[2][]{}
22 \newcommand*{\rfoot}[2][]{}
23
24 \newcommand{\nouppercase}[1]{#1}
25
26 \NewDocumentCommand{\fancycenter}{o o m m m}{}
27
28 \NewDocumentCommand{\newpagestyle}{m o m}{}
29
30 \newcommand*{\iffloatpage}[2]{#2}
31 \newcommand*{\iftopfloat}[2]{#2}
32 \newcommand*{\ifbotfloat}[2]{#2}
```

File 332   **lwarp-nccfoots.sty**

§ 444   Package   # nccfoots

(*Emulates or patches code by* ALEXANDER I. ROZHENKO.)

nccfoots (*Pkg*)   nccfoots is used as-is, and emulated for MATHJAX.

for HTML output: 1 \LWR@ProvidesPackagePass{nccfoots}[2005/02/03]

To nullify the footnotes where necessary:

2 \apptocmd{\LWR@nullifyfootnotes}{%
3   \renewcommand*{\Footnote}[1]{}%
4   \renewcommand*{\Footnotemark}[1]{}%
5 }{}{}

⚠   For MATHJAX. There is no way to test for an empty argument, so the mark is not
automatically duplicated.

6 \begin{warpMathJax}
7 \CustomizeMathJax{\newcommand{\Footnotemark}[1]{{}^{\mathrm{#1}}}}
8 \CustomizeMathJax{\newcommand{\Footnote}[2]{\Footnotemark{#1}}}
9 \end{warpMathJax}

File 333   **lwarp-nccmath.sty**

§ 445   Package   # nccmath

(*Emulates or patches code by* ALEXANDER I. ROZHENKO.)

nccmath (*Pkg*)   nccmath is patched for use by lwarp, and emulated for MATHJAX.

for HTML output: 1 \LWR@ProvidesPackagePass{nccmath}[2006/01/20]

2 \let\LWR@origeqnarray\eqnarray
3 \let\LWR@origendeqnarray\endeqnarray
4
5 \csletcs{LWR@origeqnarraystar}{eqnarray*}
6 \csletcs{LWR@origendeqnarraystar}{endeqnarray*}
7
8 \RenewEnviron{eqnarray}
9 {%
10
11   \LWR@eqnarrayfactor
12
13 }
14
15 \RenewEnviron{eqnarray*}
16 {%
17
18   \begingroup

```
19      \csletcs{LWR@origeqnarray}{LWR@origeqnarraystar}
20      \csletcs{LWR@origendeqnarray}{LWR@origendeqnarraystar}
21      \boolfalse{LWR@numbereqnarray}
22      \LWR@eqnarrayfactor
23      \endgroup
24
25 }
26
27 \def\eqs{%
28      \@ifstar\LWR@nccmath@eqsstar\LWR@nccmath@eqs%
29 }
30 \newcommand*{\LWR@nccmath@eqsstar}[2][]{\begin{eqnarray*}#2\end{eqnarray*}}
31 \newcommand*{\LWR@nccmath@eqs}[2][]{\begin{eqnarray}#2\end{eqnarray}}
32
33 \begin{warpMathJax}

34 \CustomizeMathJax{\renewcommand{\intertext}[2][]{\text{#2}\notag \\}}
35 \CustomizeMathJax{\newenvironment{fleqn}[1][]{}{}}
36 \CustomizeMathJax{\newenvironment{ceqn}{}{}}
37 \CustomizeMathJax{\newenvironment{darray}[2][c]{\begin{array}[#1]{#2}}{\end{array}}}
38 \CustomizeMathJax{\newcommand{\dmulticolumn}[3]{#3}}
```

As of v0.86, MATHJAX v3 does not offer *, so the unstarred version is used here.

```
39 \CustomizeMathJax{\newcommand{\LWRnrnostar}[1][0.5ex]{\\[#1]}}
40 \CustomizeMathJax{\newcommand{\nr}{\ifstar\LWRnrnostar\LWRnrnostar}}
41
42 \CustomizeMathJax{\newcommand{\mrel}[1]{\begin{aligned}#1\end{aligned}}}
43 \CustomizeMathJax{\newcommand{\underrel}[2]{\underset{#2}{#1}}}
44 \CustomizeMathJax{\newcommand{\medmath}[1]{#1}}
45 \CustomizeMathJax{\newcommand{\medop}[1]{#1}}
46 \CustomizeMathJax{\newcommand{\medint}[1]{#1}}
47 \CustomizeMathJax{\newcommand{\medintcorr}[1]{#1}}
48 \CustomizeMathJax{\newcommand{\mfrac}[2]{\frac{#1}{#2}}}
49 \CustomizeMathJax{\newcommand{\mbinom}[2]{\binom{#1}{#2}}}
50 \CustomizeMathJax{\newenvironment{mmatrix}{\begin{matrix}}{\end{matrix}}}

51 \CustomizeMathJax{\newcommand{\displaybreak}[1][]{}}
```

\eq, \eqs, \eqalign are created by LATEX, not MATHJAX.

```
52 \end{warpMathJax}
```

File 334   **lwarp-needspace.sty**

§ 446   Package   # needspace

(*Emulates or patches code by* PETER WILSON.)

needspace (*Pkg*)   needspace is ignored.

for HTML output: Discard all options for lwarp-needspace:

```
1 \LWR@ProvidesPackageDrop{needspace}[2010/09/12]
2
3 \DeclareDocumentCommand{\needspace}{m}{}
4 \DeclareDocumentCommand{\Needspace}{s m}{}
```

File 335 **`lwarp-newpxmath.sty`**

§ 447   Package # newpxmath

(*Emulates or patches code by* MICHAEL SHARPE.)

newpxmath (*Pkg*)   newpxmath is used as-is for SVG math, and is emulated for MATHJAX.

⚠   limitations   The MATHJAX emulation ignores all package options, except `slantedGreek` is honored. The dedicated macros for upright and italic Greek do work correctly.

SVG math should appear the same as the printed output.

for HTML output: The MATHJAX code from newtxmath is used:

```
1 \LWR@ProvidesPackagePass{newpxmath}[2020/01/09]
2
3 \LWR@infoprocessingmathjax{newpxmath}
4
5 \LWR@origRequirePackage{lwarp-common-mathjax-newpxtxmath}
6
7 \LWR@origRequirePackage{lwarp-common-mathjax-letters}
8
9 \begin{warpMathJax}
10
11 %   * \marg{2: prefix} \marg{3: postfix} \marg{4: i/u: italic/upright}
12 \LWR@mathjax@addgreek@u@up*{}{up}
13 \LWR@mathjax@addgreek@u@up*{up}{}
14 \LWR@mathjax@addgreek@l@up{up}{}
15 \LWR@mathjax@addgreek@l@up{}{up}
16 \LWR@mathjax@addgreek@u@it*{}{it}
17 \LWR@mathjax@addgreek@l@it{}{it}
```

Optional slanted Greek:

```
18 \ifpx@slantedG
19     \LWR@mathjax@addgreek@u@it*{}{}
20 \fi
21
22 \end{warpMathJax}
```

File 336   **`lwarp-newtxmath.sty`**

§ 448   Package # newtxmath

(*Emulates or patches code by* MICHAEL SHARPE.)

newtxmath (*Pkg*)   newtxmath is used as-is for SVG math, and is emulated for MATHJAX.

⚠   limitations   The MATHJAX emulation ignores all package options, except `slantedGreek` is honored, and except that bold italic Latin letters are not defined for MATHJAX if the option is not selected.

The dedicated macros for upright and italic Greek and bold italic Latin letters do work correctly.

svg math should appear the same as the printed output.

```
1 \LWR@ProvidesPackagePass{newtxmath}[2020/08/04]
2
3 \LWR@infoprocessingmathjax{newtxmath}
4
5 \LWR@origRequirePackage{lwarp-common-mathjax-newpxtxmath}
6
7 \LWR@origRequirePackage{lwarp-common-mathjax-letters}
8
9 \begin{warpMathJax}
10
11 %   * \marg{2: prefix} \marg{3: postfix} \marg{4: i/u: italic/upright}
12 \LWR@mathjax@addgreek@u@up*{}{up}
13 \LWR@mathjax@addgreek@u@up*{up}{}
14 \LWR@mathjax@addgreek@l@up{up}{}
15 \LWR@mathjax@addgreek@l@up{}{up}
16 \LWR@mathjax@addgreek@u@it*{}{it}
17 \LWR@mathjax@addgreek@l@it{}{it}
18
19 % only newtxmath, not newpxmath:
20 \LWR@mathjax@addgreek@u@it*{it}{}
21 \LWR@mathjax@addgreek@l@it{it}{}
22
23 % only newtxmath, not newpxmath:
24 \ifdef{\iftx@BI}{
25     \iftx@BI
26         \LWR@mathjax@addlatin@u@bfit{BI}
27         \LWR@mathjax@addlatin@l@bfit{BI}
28     \fi
29 }{}
```

Optional slanted Greek:

```
30 \iftx@slantedG
31     \LWR@mathjax@addgreek@u@it*{}{}
32 \fi
33
34 \end{warpMathJax}
```

File 337 **lwarp-newtxsf.sty**

§ 449   Package   # newtxsf

(*Emulates or patches code by* MICHAEL SHARPE.)

newtxsf (*Pkg*)   newtxsf is used as-is for svg math, and is emulated for MATHJAX.

⚠ limitations   The MATHJAX emulation ignores all package options, except slantedGreek is honored. The dedicated macros for upright and italic Greek and bold italic Latin letters do work correctly.

svg math should appear the same as the printed output.

```
1 \LWR@ProvidesPackagePass{newtxsf}[2020/05/02]
2
3 \LWR@infoprocessingmathjax{newtxsf}
4
5 \LWR@origRequirePackage{lwarp-common-mathjax-newpxtxmath}
6
7 \LWR@origRequirePackage{lwarp-common-mathjax-letters}
8
9 \begin{warpMathJax}
10
11 %   * \marg{2: prefix} \marg{3: postfix} \marg{4: i/u: italic/upright}
12 \LWR@mathjax@addgreek@u@up*{}{up}
13 \LWR@mathjax@addgreek@u@up*{up}{}
14 \LWR@mathjax@addgreek@l@up{up}{}
15 \LWR@mathjax@addgreek@l@up{}{up}
16 \LWR@mathjax@addgreek@u@it*{}{it}
17 \LWR@mathjax@addgreek@l@it{}{it}
18
19 % only newtxmath, not newpxmath:
20 \LWR@mathjax@addgreek@u@it*{it}{}
21 \LWR@mathjax@addgreek@l@it{it}{}
22 %
23 % only newtxmath, not newpxmath:
24 \ifdef{\iftx@BI}{
25     \iftx@BI
26         \LWR@mathjax@addlatin@u@bfit{BI}
27         \LWR@mathjax@addlatin@l@bfit{BI}
28     \fi
29 }{}
```

Optional slanted Greek:

```
30 \iftx@slantedG
31     \LWR@mathjax@addgreek@u@it*{}{}
32 \fi
33
34 \end{warpMathJax}
```

File 338   **`lwarp-nextpage.sty`**

§ 450   Package   # **nextpage**

(*Emulates or patches code by* PETER WILSON.)

nextpage (*Pkg*)   nextpage is ignored.

for HTML output: Discard all options for lwarp-nextpage.

```
1 \LWR@ProvidesPackageDrop{nextpage}[2009/09/03]
```

```
2 \DeclareDocumentCommand{\cleartoevenpage}{o}{}
3 \DeclareDocumentCommand{\movetoevenpage}{o}{}
4 \DeclareDocumentCommand{\cleartooddpage}{o}{}
5 \DeclareDocumentCommand{\movetooddpage}{o}{}
```

File 339   **`lwarp-nfssext-cfr.sty`**

§ 451   Package   # nfssext-cfr

(*Emulates or patches code by* CLEA F. REES.)

nfssext-cfr (*Pkg*)   nfssext-cfr is emulated in HTML, and used as-is in print output.

Results depend on the browser's font.

for HTML output: 1 \LWR@ProvidesPackagePass{nfssext-cfr}[2017/03/28]

Macros which are present in the lwarp core are commented out here.

```
2 \newrobustcmd{\LWR@HTML@lnstyle}{}
3 \newrobustcmd{\LWR@HTML@osstyle}{\LWR@HTML@scshape}
4 \newrobustcmd{\LWR@HTML@instyle}{}
5 \newrobustcmd{\LWR@HTML@sustyle}{}
6 \newrobustcmd{\LWR@HTML@swstyle}{}
7 \newrobustcmd{\LWR@HTML@pstyle}{}
8 \newrobustcmd{\LWR@HTML@tistyle}{}
9 \newrobustcmd{\LWR@HTML@ostyle}{\LWR@HTML@scshape}
10 \newrobustcmd{\LWR@HTML@postyle}{\LWR@HTML@scshape}
11 \newrobustcmd{\LWR@HTML@ltstyle}{}
12 \newrobustcmd{\LWR@HTML@ofstyle}{}
13 \newrobustcmd{\LWR@HTML@altstyle}{}
14 \newrobustcmd{\LWR@HTML@regstyle}{}
15 \newrobustcmd{\LWR@HTML@embossstyle}{}
16 \newrobustcmd{\LWR@HTML@ornamentalstyle}{}
17 \newrobustcmd{\LWR@HTML@qtstyle}{}
18 \newrobustcmd{\LWR@HTML@shstyle}{}
19 \newrobustcmd{\LWR@HTML@swashstyle}{}
20 \newrobustcmd{\LWR@HTML@tmstyle}{\renewcommand*{\LWR@f@family}{tt}}
21 \newrobustcmd{\LWR@HTML@tvstyle}{\renewcommand*{\LWR@f@family}{tt}}
22 \newrobustcmd{\LWR@HTML@tstyle}{}
23 \newrobustcmd{\LWR@HTML@lstyle}{}
24 \newrobustcmd{\LWR@HTML@tlstyle}{}
25 \newrobustcmd{\LWR@HTML@plstyle}{}
26 \newrobustcmd{\LWR@HTML@tostyle}{\LWR@HTML@scshape}
27 % \newrobustcmd{\LWR@HTML@sishape}{}
28 \newrobustcmd{\LWR@HTML@olshape}{}
29 \newrobustcmd{\LWR@HTML@scolshape}{}
30 \newrobustcmd{\LWR@HTML@ushape}{}
31 \newrobustcmd{\LWR@HTML@scushape}{}
32 \newrobustcmd{\LWR@HTML@uishape}{\LWR@HTML@itshape}
33 \newrobustcmd{\LWR@HTML@rishape}{}
34 \newrobustcmd{\LWR@HTML@regwidth}{}
35 \newrobustcmd{\LWR@HTML@nwwidth}{}
36 \newrobustcmd{\LWR@HTML@cdwidth}{}
37 \newrobustcmd{\LWR@HTML@ecwidth}{}
38 \newrobustcmd{\LWR@HTML@ucwidth}{}
39 \newrobustcmd{\LWR@HTML@etwidth}{}
40 \newrobustcmd{\LWR@HTML@epwidth}{}
41 \newrobustcmd{\LWR@HTML@exwidth}{}
42 \newrobustcmd{\LWR@HTML@uxwidth}{}
43 \newrobustcmd{\LWR@HTML@mbweight}{\renewcommand*{\LWR@f@series}{md}}
```

```
44 \newrobustcmd{\LWR@HTML@dbweight}{\renewcommand*{\LWR@f@series}{db}}
45 \newrobustcmd{\LWR@HTML@sbweight}{\renewcommand*{\LWR@f@series}{sb}}
46 % \newrobustcmd{\LWR@HTML@ebweight}{\renewcommand*{\LWR@f@series}{eb}}
47 \newrobustcmd{\LWR@HTML@ubweight}{\renewcommand*{\LWR@f@series}{ub}}
48 % \newrobustcmd{\LWR@HTML@lgweight}{\renewcommand*{\LWR@f@series}{lg}}
49 \newrobustcmd{\LWR@HTML@elweight}{\renewcommand*{\LWR@f@series}{el}}
50 \newrobustcmd{\LWR@HTML@ulweight}{\renewcommand*{\LWR@f@series}{ul}}
51 % \newrobustcmd{\LWR@HTML@itshape}{}
52 % \newrobustcmd{\LWR@HTML@scshape}{}
53 % \newrobustcmd{\LWR@HTML@upshape}{}
54 \newrobustcmd{\LWR@HTML@dfshape}{}
55
56 \ifdef{\LWR@HTML@swshape}{}{% duplicated by fontaxes
57     \newrobustcmd{\LWR@HTML@swshape}{}
58 }
59
60 \newrobustcmd{\LWR@HTML@ornament}[1]{}
61
62 \LWR@formatted{lnstyle}
63 \LWR@formatted{osstyle}
64 \LWR@formatted{instyle}
65 \LWR@formatted{sustyle}
66 \LWR@formatted{swstyle}
67 \LWR@formatted{pstyle}
68 \LWR@formatted{tistyle}
69 \LWR@formatted{ostyle}
70 \LWR@formatted{postyle}
71 \LWR@formatted{ltstyle}
72 \LWR@formatted{ofstyle}
73 \LWR@formatted{altstyle}
74 \LWR@formatted{regstyle}
75 \LWR@formatted{embossstyle}
76 \LWR@formatted{ornamentalstyle}
77 \LWR@formatted{qtstyle}
78 \LWR@formatted{shstyle}
79 \LWR@formatted{swashstyle}
80 \LWR@formatted{tmstyle}
81 \LWR@formatted{tvstyle}
82 \LWR@formatted{tstyle}
83 \LWR@formatted{lstyle}
84 \LWR@formatted{tlstyle}
85 \LWR@formatted{plstyle}
86 \LWR@formatted{tostyle}
87 % \LWR@formatted{sishape}
88 \LWR@formatted{olshape}
89 \LWR@formatted{scolshape}
90 \LWR@formatted{ushape}
91 \LWR@formatted{scushape}
92 \LWR@formatted{uishape}
93 \LWR@formatted{rishape}
94 \LWR@formatted{regwidth}
95 \LWR@formatted{nwwidth}
96 \LWR@formatted{cdwidth}
97 \LWR@formatted{ecwidth}
98 \LWR@formatted{ucwidth}
99 \LWR@formatted{etwidth}
100 \LWR@formatted{epwidth}
101 \LWR@formatted{exwidth}
102 \LWR@formatted{uxwidth}
103 \LWR@formatted{mbweight}
```

```
104 \LWR@formatted{dbweight}
105 \LWR@formatted{sbweight}
106 % \LWR@formatted{ebweight}
107 \LWR@formatted{ubweight}
108 % \LWR@formatted{lgweight}
109 \LWR@formatted{elweight}
110 \LWR@formatted{ulweight}
111 \LWR@formatted{itshape}% adapt to the new print version
112 \LWR@formatted{scshape}% adapt to the new print version
113 \LWR@formatted{upshape}% adapt to the new print version
114 \LWR@formatted{dfshape}
115
116 \ifdef{\LWR@HTML@swshape}{}{% duplicated by fontaxes
117     \LWR@formatted{swshape}
118 }
119
120 \LWR@formatted{ornament}


121 \FilenameNullify{%
122     \LetLtxMacro\lnstyle\@empty%
123     \LetLtxMacro\osstyle\@empty%
124     \LetLtxMacro\instyle\@empty%
125     \LetLtxMacro\sustyle\@empty%
126     \LetLtxMacro\swstyle\@empty%
127     \LetLtxMacro\pstyle\@empty%
128     \LetLtxMacro\tistyle\@empty%
129     \LetLtxMacro\ostyle\@empty%
130     \LetLtxMacro\postyle\@empty%
131     \LetLtxMacro\ltstyle\@empty%
132     \LetLtxMacro\ofstyle\@empty%
133     \LetLtxMacro\altstyle\@empty%
134     \LetLtxMacro\regstyle\@empty%
135     \LetLtxMacro\embossstyle\@empty%
136     \LetLtxMacro\ornamentalstyle\@empty%
137     \LetLtxMacro\qtstyle\@empty%
138     \LetLtxMacro\shstyle\@empty%
139     \LetLtxMacro\swashstyle\@empty%
140     \LetLtxMacro\tmstyle\@empty%
141     \LetLtxMacro\tvstyle\@empty%
142     \LetLtxMacro\tstyle\@empty%
143     \LetLtxMacro\lstyle\@empty%
144     \LetLtxMacro\tlstyle\@empty%
145     \LetLtxMacro\plstyle\@empty%
146     \LetLtxMacro\tostyle\@empty%
147 %   \LetLtxMacro\sishape\@empty%
148     \LetLtxMacro\olshape\@empty%
149     \LetLtxMacro\scolshape\@empty%
150     \LetLtxMacro\ushape\@empty%
151     \LetLtxMacro\scushape\@empty%
152     \LetLtxMacro\uishape\@empty%
153     \LetLtxMacro\rishape\@empty%
154     \LetLtxMacro\regwidth\@empty%
155     \LetLtxMacro\nwwidth\@empty%
156     \LetLtxMacro\cdwidth\@empty%
157     \LetLtxMacro\ecwidth\@empty%
158     \LetLtxMacro\ucwidth\@empty%
159     \LetLtxMacro\etwidth\@empty%
160     \LetLtxMacro\epwidth\@empty%
161     \LetLtxMacro\exwidth\@empty%
162     \LetLtxMacro\uxwidth\@empty%
```

```
163    \LetLtxMacro\mbweight\@empty%
164    \LetLtxMacro\dbweight\@empty%
165    \LetLtxMacro\sbweight\@empty%
166 %  \LetLtxMacro\ebweight\@empty%
167    \LetLtxMacro\ubweight\@empty%
168 %  \LetLtxMacro\lgweight\@empty%
169    \LetLtxMacro\elweight\@empty%
170    \LetLtxMacro\ulweight\@empty%
171 %  \LetLtxMacro\itshape\@empty%
172 %  \LetLtxMacro\scshape\@empty%
173 %  \LetLtxMacro\upshape\@empty%
174    \LetLtxMacro\dfshape\@empty%
175    \LetLtxMacro\swshape\@empty%
176    \LetLtxMacro\ornament\@gobble%
177 }
178
179 \newrobustcmd{\LWR@HTML@textln}[1]{\InlineClass{textln}{#1}}
180 \newrobustcmd{\LWR@HTML@textos}[1]{\textsc{#1}}
181 \newrobustcmd{\LWR@HTML@textin}[1]{#1}
182 \providerobustcmd{\textin}[1]{}
183 \newrobustcmd{\LWR@HTML@textsu}[1]{#1}
184 % \newrobustcmd{\LWR@HTML@textsi}[1]{#1}
185 \newrobustcmd{\LWR@HTML@textdf}[1]{#1}


186 \ifdef{\LWR@HTML@textsw}{}{% duplicated by fontaxes
187    \newrobustcmd{\LWR@HTML@textsw}[1]{#1}
188    \LWR@formatted{textsw}
189 }
190
191 \newrobustcmd{\LWR@HTML@textti}[1]{#1}
192 \newrobustcmd{\LWR@HTML@textlt}[1]{#1}
193 \newrobustcmd{\LWR@HTML@textof}[1]{#1}
194 \newrobustcmd{\LWR@HTML@textalt}[1]{#1}
195 \newrobustcmd{\LWR@HTML@textreg}[1]{#1}
196 \newrobustcmd{\LWR@HTML@emboss}[1]{#1}
197 \newrobustcmd{\LWR@HTML@textorn}[1]{#1}
198 \newrobustcmd{\LWR@HTML@textqt}[1]{#1}
199 \newrobustcmd{\LWR@HTML@textsh}[1]{#1}
200 \newrobustcmd{\LWR@HTML@texttm}[1]{\texttt{#1}}
201 \newrobustcmd{\LWR@HTML@texttv}[1]{\texttt{#1}}
202 \newrobustcmd{\LWR@HTML@textl}[1]{\InlineClass{textln}{#1}}
203 \newrobustcmd{\LWR@HTML@texto}[1]{\textsc{#1}}
204 \newrobustcmd{\LWR@HTML@textp}[1]{\InlineClass{textp}{#1}}
205 \newrobustcmd{\LWR@HTML@textt}[1]{\InlineClass{textt}{#1}}
206 \newrobustcmd{\LWR@HTML@textpl}[1]{#1}
207 \newrobustcmd{\LWR@HTML@textpo}[1]{\textsc{#1}}
208 \newrobustcmd{\LWR@HTML@texttl}[1]{\InlineClass{textln}{#1}}
209 \newrobustcmd{\LWR@HTML@textto}[1]{\textsc{#1}}
210 \newrobustcmd{\LWR@HTML@textol}[1]{#1}
211 \newrobustcmd{\LWR@HTML@textswash}[1]{#1}
212 \newrobustcmd{\LWR@HTML@textu}[1]{#1}
213 \newrobustcmd{\LWR@HTML@textscu}[1]{#1}
214 \newrobustcmd{\LWR@HTML@textui}[1]{\LWR@HTML@textit{#1}}
215 \newrobustcmd{\LWR@HTML@textri}[1]{#1}
216 \newrobustcmd{\LWR@HTML@textnw}[1]{#1}
217 \newrobustcmd{\LWR@HTML@textcd}[1]{#1}
218 \newrobustcmd{\LWR@HTML@textec}[1]{#1}
219 \newrobustcmd{\LWR@HTML@textuc}[1]{#1}
220 \newrobustcmd{\LWR@HTML@textet}[1]{#1}
221 \newrobustcmd{\LWR@HTML@textep}[1]{#1}
```

```
222 \newrobustcmd{\LWR@HTML@textex}[1]{#1}
223 \newrobustcmd{\LWR@HTML@textux}[1]{#1}
224 \newrobustcmd{\LWR@HTML@textrw}[1]{#1}
225 \newrobustcmd{\LWR@HTML@textmb}[1]{{\LWR@HTML@mbweight\InlineClass{textmb}{#1}}}
226 \newrobustcmd{\LWR@HTML@textdb}[1]{{\LWR@HTML@dbweight\InlineClass{textdb}{#1}}}
227 \newrobustcmd{\LWR@HTML@textsb}[1]{{\LWR@HTML@sbweight\InlineClass{textsb}{#1}}}
228 % \newrobustcmd{\LWR@HTML@texteb}[1]]{#1}
229 \newrobustcmd{\LWR@HTML@textub}[1]{{\LWR@HTML@ubweight\InlineClass{textub}{#1}}}
230 % \newrobustcmd{\LWR@HTML@textlg}[1]]{#1}
231 \newrobustcmd{\LWR@HTML@textel}[1]{{\LWR@HTML@elweight\InlineClass{textel}{#1}}}
232 \newrobustcmd{\LWR@HTML@textul}[1]{{\LWR@HTML@ulweight\InlineClass{textul}{#1}}}
233
234 \LWR@formatted{textln}
235 \LWR@formatted{textos}
236 \LWR@formatted{textin}
237 \LWR@formatted{textsu}
238 % \LWR@formatted{textsi}
239 \LWR@formatted{textdf}
240 \LWR@formatted{textti}
241 \LWR@formatted{textlt}
242 \LWR@formatted{textof}
243 \LWR@formatted{textalt}
244 \LWR@formatted{textreg}
245 \LWR@formatted{emboss}
246 \LWR@formatted{textorn}
247 \LWR@formatted{textqt}
248 \LWR@formatted{textsh}
249 \LWR@formatted{texttm}
250 \LWR@formatted{texttv}
251 \LWR@formatted{textl}
252 \LWR@formatted{texto}
253 \LWR@formatted{textp}
254 \LWR@formatted{textt}
255 \LWR@formatted{textpl}
256 \LWR@formatted{textpo}
257 \LWR@formatted{texttl}
258 \LWR@formatted{textto}
259 \LWR@formatted{textol}
260 \LWR@formatted{textswash}
261 \LWR@formatted{textu}
262 \LWR@formatted{textscu}
263 \LWR@formatted{textui}
264 \LWR@formatted{textri}
265 \LWR@formatted{textnw}
266 \LWR@formatted{textcd}
267 \LWR@formatted{textec}
268 \LWR@formatted{textuc}
269 \LWR@formatted{textet}
270 \LWR@formatted{textep}
271 \LWR@formatted{textex}
272 \LWR@formatted{textux}
273 \LWR@formatted{textrw}
274 \LWR@formatted{textmb}
275 \LWR@formatted{textdb}
276 \LWR@formatted{textsb}
277 % \LWR@formatted{texteb}
278 \LWR@formatted{textub}
279 % \LWR@formatted{textlg}
280 \LWR@formatted{textel}
281 \LWR@formatted{textul}
```

282

```
283 \FilenameNullify{%
284     \LetLtxMacro\textln\@firstofone%
285     \LetLtxMacro\textos\@firstofone%
286     \LetLtxMacro\textin\@firstofone%
287     \LetLtxMacro\textsu\@firstofone%
288 %   \LetLtxMacro\textsi\@firstofone%
289     \LetLtxMacro\textdf\@firstofone%
290     \LetLtxMacro\textsw\@firstofone%
291     \LetLtxMacro\textti\@firstofone%
292     \LetLtxMacro\textlt\@firstofone%
293     \LetLtxMacro\textof\@firstofone%
294     \LetLtxMacro\textalt\@firstofone%
295     \LetLtxMacro\textreg\@firstofone%
296     \LetLtxMacro\emboss\@firstofone%
297     \LetLtxMacro\textorn\@firstofone%
298     \LetLtxMacro\textqt\@firstofone%
299     \LetLtxMacro\textsh\@firstofone%
300     \LetLtxMacro\texttm\@firstofone%
301     \LetLtxMacro\texttv\@firstofone%
302     \LetLtxMacro\textl\@firstofone%
303     \LetLtxMacro\texto\@firstofone%
304     \LetLtxMacro\textp\@firstofone%
305     \LetLtxMacro\textt\@firstofone%
306     \LetLtxMacro\textpl\@firstofone%
307     \LetLtxMacro\textpo\@firstofone%
308     \LetLtxMacro\textttl\@firstofone%
309     \LetLtxMacro\textto\@firstofone%
310     \LetLtxMacro\textol\@firstofone%
311     \LetLtxMacro\textswash\@firstofone%
312     \LetLtxMacro\textu\@firstofone%
313     \LetLtxMacro\textscu\@firstofone%
314     \LetLtxMacro\textui\@firstofone%
315     \LetLtxMacro\textri\@firstofone%
316     \LetLtxMacro\textnw\@firstofone%
317     \LetLtxMacro\textcd\@firstofone%
318     \LetLtxMacro\textec\@firstofone%
319     \LetLtxMacro\textuc\@firstofone%
320     \LetLtxMacro\textet\@firstofone%
321     \LetLtxMacro\textep\@firstofone%
322     \LetLtxMacro\textex\@firstofone%
323     \LetLtxMacro\textux\@firstofone%
324     \LetLtxMacro\textrw\@firstofone%
325     \LetLtxMacro\textmb\@firstofone%
326     \LetLtxMacro\textdb\@firstofone%
327     \LetLtxMacro\textsb\@firstofone%
328 %   \LetLtxMacro\texteb\@firstofone%
329     \LetLtxMacro\textub\@firstofone%
330 %   \LetLtxMacro\textlg\@firstofone%
331     \LetLtxMacro\textel\@firstofone%
332     \LetLtxMacro\textul\@firstofone%
333 }
334
335 \providecommand*{\zeroslash}{0}
336 \newrobustcmd*{\LWR@HTML@zeroslash}{0}
337 \LWR@formatted{zeroslash}
```

File 340   **lwarp-nicefrac.sty**

§ 452   Package   **nicefrac**

(*Emulates or patches code by* AXEL REICHERT.)

nicefrac (*Pkg*)   nicefrac is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{nicefrac}[1998/08/04]

2 \DeclareRobustCommand*{\LWR@HTML@@UnitsNiceFrac}[3][]{%
3   {% localize font selection
4   #1{%
5   \LWR@textcurrentfont{%
6   \InlineClass{numerator}{#2}%
7   /%
8   \InlineClass{denominator}{#3}%
9   }%
10   }%
11   }%
12 }
13
14 \LWR@formatted{@UnitsNiceFrac}
15
16 \DeclareRobustCommand*{\LWR@HTML@@UnitsUglyFrac}[3][]{%
17   {% localize font selection
18   #1{\LWR@textcurrentfont{#2/#3}}%
19   }%
20 }
21
22 \LWR@formatted{@UnitsUglyFrac}

For MATHJAX:

23 \begin{warpMathJax}
24 \CustomizeMathJax{\newcommand{\nicefrac}[3][]{\mathinner{{}^{#2}\!/\!_{#3}}}}
25 \end{warpMathJax}

File 341   **lwarp-niceframe.sty**

§ 453   Package   **niceframe**

niceframe (*Pkg*)   niceframe is emulated.

for HTML output: 1 \LWR@ProvidesPackageDrop{niceframe}% the original date is in yyyy/dd/mm format

2 \newcommand{\LWR@niceframe}[3]{%
3   \begin{LWR@setvirtualpage}*%
4   \setlength{\LWR@templengthone}{#1}%
5   \begin{BlockClass}[max-width:\LWR@printlength{\LWR@templengthone}]{#3}%
6   #2
7   \end{BlockClass}%

```
 8      \end{LWR@setvirtualpage}%
 9 }
10
11 \newcommand{\niceframe}[2][\textwidth]{\LWR@niceframe{#1}{#2}{niceframe}}
12 \newcommand{\curlyframe}[2][\textwidth]{\LWR@niceframe{#1}{#2}{curlyframe}}
13 \newcommand{\artdecoframe}[2][\textwidth]{\LWR@niceframe{#1}{#2}{artdecoframe}}
14
15 \newcommand{\generalframe}[9]{\LWR@niceframe{\textwidth}{#9}{generalframe}}
```

File 342   **`lwarp-nicematrix.sty`**

§ 454   Package   # nicematrix

(*Emulates or patches code by* F. PANTIGNY.)

nicematrix (*Pkg*)   nicematrix is used as-is for SVG math, and is emulated for MATHJAX.

⚠ MATHJAX   Keys/values are ignored in MATHJAX. \Cdots, etc. do not span multiple cells. AutoNiceMatrix, etc. are not supported for MATHJAX. SVG math output preserves all nicematrix features. To force SVG output for one or more consecutive math expressions, for inline math use \inlinemathother and \inlinemathnormal, or for display math use \displaymathother and \displaymathnormal.

for HTML output: Skip the test for array, which does not work with lwarp:

```
1 \PassOptionsToPackage{no-test-for-array}{nicematrix}
```

```
2 \LWR@ProvidesPackagePass{nicematrix}[2022/10/06]
```

NiceTabular must be converted to SVG to support the various nicematrix options:

```
 3 \begin{warpHTML}
 4 \BeforeBeginEnvironment{NiceTabular}{%
 5     \begin{lateximage}[-nicematrix-~\PackageDiagramAltText]?%
 6 }
 7 \AfterEndEnvironment{NiceTabular}{\end{lateximage}}
 8 \BeforeBeginEnvironment{NiceTabular*}{%
 9     \begin{lateximage}[-nicematrix-~\PackageDiagramAltText]?%
10 }
11 \AfterEndEnvironment{NiceTabular*}{\end{lateximage}}
12 \end{warpHTML}
```

Special handling for the optional arguments, and the lack of a delimiter:

```
13 \begin{warpMathJax}
14 \CustomizeMathJax{\newcommand{\LWRnicearrayarray}[1]{\begin{array}{#1}}}
15 \CustomizeMathJax{\def\LWRnicearrayarrayopt#1[#2] {\begin{array}{#1}}}
16
17 \CustomizeMathJax{%
18     \newenvironment{NiceArray}[2][]%
19         {\ifnextchar[{\LWRnicearrayarrayopt{#2}}{\LWRnicearrayarray{#2}}}%
20         {\end{array}}%
21 }
22
23 \CustomizeMathJax{%
24     \newcommand{\LWRnicearraywithdelimtwo}[2][]{%
```

```
25        \ifnextchar[{\LWRnicearrayarrayopt{#2}}{\LWRnicearrayarray{#2}}%
26    }%
27 }
```

General case with left / right delimiters:

```
28 \CustomizeMathJax{%
29    \newenvironment{NiceArrayWithDelims}[2]%
30        {%
31            \def\LWRnicearrayrightdelim{\right#2}%
32            \left#1%
33            \LWRnicearraywithdelimtwo%
34        }%
35        {\end{array}\LWRnicearrayrightdelim}%
36 }
```

Instances of specific delimiters:

```
37 \CustomizeMathJax{%
38    \newenvironment{pNiceArray}
39        {\begin{NiceArrayWithDelims}{(}{)}}
40        {\end{NiceArrayWithDelims}}
41 }
42
43 \CustomizeMathJax{%
44    \newenvironment{bNiceArray}
45        {\begin{NiceArrayWithDelims}{[}{]}}
46        {\end{NiceArrayWithDelims}}
47 }
48
49 \CustomizeMathJax{%
50    \newenvironment{BNiceArray}
51        {\begin{NiceArrayWithDelims}{\{}{\}}}
52        {\end{NiceArrayWithDelims}}
53 }
54
55 \CustomizeMathJax{%
56    \newenvironment{vNiceArray}
57        {\begin{NiceArrayWithDelims}{\vert}{\vert}}
58        {\end{NiceArrayWithDelims}}
59 }
60
61 \CustomizeMathJax{%
62    \newenvironment{VNiceArray}
63        {\begin{NiceArrayWithDelims}{\Vert}{\Vert}}
64        {\end{NiceArrayWithDelims}}
65 }
```

Ignore optional arg and use standard environments:

```
66 \CustomizeMathJax{\newenvironment{NiceMatrix}[1][]{\begin{matrix}}{\end{matrix}}}
67 \CustomizeMathJax{\newenvironment{pNiceMatrix}[1][]{\begin{pmatrix}}{\end{pmatrix}}}
68 \CustomizeMathJax{\newenvironment{bNiceMatrix}[1][]{\begin{bmatrix}}{\end{bmatrix}}}
69 \CustomizeMathJax{\newenvironment{BNiceMatrix}[1][]{\begin{Bmatrix}}{\end{Bmatrix}}}
70 \CustomizeMathJax{\newenvironment{vNiceMatrix}[1][]{\begin{vmatrix}}{\end{vmatrix}}}
71 \CustomizeMathJax{\newenvironment{VNiceMatrix}[1][]{\begin{Vmatrix}}{\end{Vmatrix}}}
```

Ignore optional argument and size. Print contents.

```
72 \CustomizeMathJax{\newcommand{\LWRnicematrixBlock}[1]{#1}}
73 \CustomizeMathJax{\def\LWRnicematrixBlockopt<#1>#2{#2}}
74
75 \CustomizeMathJax{%
76    \newcommand{\Block}[2][]{\ifnextchar<\LWRnicematrixBlockopt\LWRnicematrixBlock}%
77 }
```

Form an approximation:

```
78 \CustomizeMathJax{%
79    \newcommand{\diagbox}[2]{%
80        \begin{array}{l}\hfill\quad#2\\\hline#1\quad\hfill\end{array}%
81    }%
82 }
```

More approximations:

```
83 \CustomizeMathJax{\let\hdottedline\hdashline}
```

```
84 \CustomizeMathJax{\newcommand{\Hline}[1][]{\hline}}
```

```
85 \CustomizeMathJax{\newcommand{\CodeBefore}{}}
86 \CustomizeMathJax{\newcommand{\Body}{}}
87 \CustomizeMathJax{\newcommand{\CodeAfter}{}}
88 \CustomizeMathJax{\newcommand{\line}[3][]{}}
89 \CustomizeMathJax{\newcommand{\RowStyle}[2][]{}}
90 \CustomizeMathJax{\newcommand{\LWRSubMatrix}[1][]{}}
91 \CustomizeMathJax{\newcommand{\SubMatrix}[4]{\LWRSubMatrix}}
92 \CustomizeMathJax{\newcommand{\OverBrace}[4][]{}}
93 \CustomizeMathJax{\newcommand{\UnderBrace}[4][]{}}
94 \CustomizeMathJax{\newcommand{\ShowCellNames}{}}
```

```
95 \CustomizeMathJax{\newcommand{\cellcolor}[3][]{}}
96 \CustomizeMathJax{\newcommand{\rowcolor}[3][]{}}
97 \CustomizeMathJax{\newcommand{\LWRrowcolors}[1][]{}}
98 \CustomizeMathJax{\newcommand{\rowcolors}[4][]{\LWRrowcolors}}
99 \CustomizeMathJax{\newcommand{\rowlistcolors}[3][]{\LWRrowcolors}}
100 \CustomizeMathJax{\newcommand{\columncolor}[3][]{}}
101 \CustomizeMathJax{\newcommand{\rectanglecolor}[4][]{}}
102 \CustomizeMathJax{\newcommand{\arraycolor}[2][]{}}
103 \CustomizeMathJax{\newcommand{\chessboardcolors}[3][]{}}
```

```
104 \CustomizeMathJax{\newcommand{\ldots}[1][]{\dots}}
105 \CustomizeMathJax{\newcommand{\Cdots}[1][]{\cdots}}
106 \CustomizeMathJax{\newcommand{\Vdots}[1][]{\vdots}}
107 \CustomizeMathJax{\newcommand{\Ddots}[1][]{\ddots}}
108 \CustomizeMathJax{\newcommand{\Iddots}[1][]{\mathinner{\unicode{x22F0}}}}
109
110 \CustomizeMathJax{\newcommand{\Hdotsfor}[1]{\ldots}}
111 \CustomizeMathJax{\newcommand{\Vdotsfor}[1]{\vdots}}
```

There is no way to emulate AutoNiceMatrix in MathJax.

```
112 \CustomizeMathJax{\newcommand{\AutoNiceMatrix}[2]{\text{(AutoNiceMatrix #1)}}}
113 \CustomizeMathJax{\let\pAutoNiceMatrix\AutoNiceMatrix}
114 \CustomizeMathJax{\let\bAutoNiceMatrix\AutoNiceMatrix}
115 \CustomizeMathJax{\let\BAutoNiceMatrix\AutoNiceMatrix}
```

```
116 \CustomizeMathJax{\let\vAutoNiceMatrix\AutoNiceMatrix}
117 \CustomizeMathJax{\let\VAutoNiceMatrix\AutoNiceMatrix}
118 \end{warpMathJax}
```

File 343   **`lwarp-noitcrul.sty`**

§ 455   Package   # noitcrul

(*Emulates or patches code by* Paul Ebermann.)

noitcrul (*Pkg*)   noitcrul is used as-is for svg and emulated for MathJax.

for HTML output: 1 `\LWR@ProvidesPackagePass{noitcrul}[2006/04/11]`

```
2 \begin{warpMathJax}
3 \CustomizeMathJax{\newcommand{\noitUnderline}[1]{\underline{#1}\!}}
4 \end{warpMathJax}
```

File 344   **`lwarp-nolbreaks.sty`**

§ 456   Package   # nolbreaks

(*Emulates or patches code by* Donald Arseneau.)

nolbreaks (*Pkg*)   nolbreaks is emulated.

for HTML output: 1 `\LWR@ProvidesPackageDrop{nolbreaks}[2012/05/31]`

2 `\NewDocumentCommand{\nolbreaks}{s m}{\InlineClass{nolbreaks}{#2}}`

File 345   **`lwarp-nomencl.sty`**

§ 457   Package   # nomencl

(*Emulates or patches code by* Boris Veytsman, Bernd Schandl, Lee Netherton, CV Radhakrishnan.)

nomencl (*Pkg*)   nomencl is patched for use by lwarp.

To process the html nomenclature:

    makeindex   <project>_html.nlo   -s   nomencl.ist   -o
    <project>_html.nls

for HTML output: 1 `\LWR@ProvidesPackagePass{nomencl}[2021/11/10]`

\BaseJobname is added to the label in case xr or xr-hyper are used.

```
2 \xpatchcmd{\@@@nomenclature}
3   {\thepage}
4   {\theLWR@previousautopagelabel}
5   {}
6   {\LWR@patcherror{nomencl}{@@@nomenclature}}
```

```
7
8 \renewcommand*{\pagedeclaration}[1]{, \nameref{\BaseJobname-autopage-#1}}%
```

File 346   `lwarp-nonfloat.sty`

§ 458   Package   # nonfloat

(*Emulates or patches code by* KAI RASCHER.)

nonfloat (*Pkg*)   nonfloat is emulated.

for HTML output: `1 \LWR@ProvidesPackageDrop{nonfloat}[1999/07/05]`

```
2 \LetLtxMacro\topcaption\caption
3 \newcommand{\figcaption}{\def\@captype{figure}\caption}
4 \newcommand{\tabcaption}{\def\@captype{table}\topcaption}
5 \newenvironment{narrow}[2]{}{}
```

File 347   `lwarp-nonumonpart.sty`

§ 459   Package   # nonumonpart

nonumonpart (*Pkg*)   nonumonpart is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{nonumonpart}[2011/04/15]`

File 348   `lwarp-nopageno.sty`

§ 460   Package   # nopageno

nopageno (*Pkg*)   nopageno is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{nopageno}[1989/01/01]`

File 349   `lwarp-notes.sty`

§ 461   Package   # notes

notes (*Pkg*)   notes is emulated.

for HTML output: `1 \LWR@ProvidesPackageDrop{notes}[2002/10/29]`

```
2 \newcommand*{\LWR@notes@onenote}[2]{%
3 \newenvironment{#1}
4     {
5         \BlockClass{notes#1}
6         \begin{BlockClass}{notesicon}\textcircled{~#2~}\end{BlockClass}
7         \BlockClass{notescontents}
```

```
 8    }
 9    {\endBlockClass\endBlockClass}
10 }
11
12 \LWR@notes@onenote{importantnote}{!}
13
14 \LWR@notes@onenote{warningnote}{--}
15
16 \LWR@notes@onenote{informationnote}{i}
```

File 350   **`lwarp-notespages.sty`**

§ 462   Package   # notespages

notespages (*Pkg*)   notespages is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{notespages}[2016/08/21]`

```
 2 \newcommand*{\npnotesname}{}
 3 \newcommand*{\npnotestext}{}
 4 \newcommand*{\remainingtextheight}{}
 5 \newdimen\remainingtextheight
 6 \newcommand*{\notestitletext}{}
 7 \newcommand*{\notesareatext}{}
 8 \newcommand*{\npnpinfo}[1]{}
 9 \newcommand*{\tracingnpmarks}{}
10 \newcommand*{\notespage}[1][]{}
11 \newcommand*{\notespages}[1][]{}
12 \newcommand*{\notesfill}[1][]{}
13 \newcommand*{\setnotespages}[1]{}
14 \newcommand*{\definenotesoption}[2]{}
15 \newcommand{\definenotesstyle}[2]{}
16 \newcommand{\definetitlestyle}[2]{}
17 \newcommand{\nppatchchapter}[1]{}
18 \newcommand{\npunpatchchapter}{}
```

File 351   **`lwarp-nowidow.sty`**

§ 463   Package   # nowidow

(*Emulates or patches code by* Raphaël Pinson.)

nowidow (*Pkg*)   nowidow is ignored.

for HTML output: DiLWR@roNislesPackageelwapfonwoidoidL2011/09/20]

\nowidow   [⟨*lines*⟩]

\setnowidow   [⟨*lines*⟩]

```
2 \newcommand*{\nowidow}[1][]{}
3 \newcommand*{\setnowidow}[1][]{}
```

\noclub   [⟨*lines*⟩]

\setnoclub   [⟨*lines*⟩]

```
4 \newcommand*{\noclub}[1][]{}
5 \newcommand*{\setnoclub}[1][]{}
```

File 352   **lwarp-ntheorem.sty**

§ 464   Package   # ntheorem

(*Emulates or patches code by* WOLFGANG MAY, ANDREAS SCHEDLER.)

ntheorem (*Pkg*)   ntheorem is patched for use by lwarp.

Table 20: Ntheorem package — CSS styling of theorems and proofs

Theorem: <div> of class theorembody<theoremstyle>

Theorem Header: <span> of class theoremheader<style>

where <theoremstyle> is plain, break, etc.

§ 464.1 **Limitations**

⚠ Font control   This conversion is not total. Font control is via CSS, and the custom LaTeX font settings are ignored.

⚠ Equation numbering   ntheorem has a bug with equation numbering in $\mathcal{A}_{\mathcal{M}}\mathcal{S}$ environments when the option thref is used. lwarp does not share this bug, so equations with \split, etc, are numbered correctly with lwarp's HTML output, but not with the print output. It is recommended to use cleveref instead of ntheorem's thref option.

§ 464.2 **Options**

Options amsthm or standard choose which set of theorems and proofs to initialize.

⚠ Disabled options   The options thmmarks and amsmath are disabled, since they heavily modify the underlying math code. Theorem marks are emulated. The AMS-math modifications are not done.

Option thref is disabled because cleveref functions are used instead. \thref is emulated.

Option hyperref is disabled because lwarp emulated hyperref.

for HTML output: Some disabled options:

```
1 \DeclareOption{thref}{
2     \AtEndDocument{
3         \PackageWarningNoLine{lwarp}{%
4             Lwarp uses cleveref, which takes over ntheorem's\MessageBreak
5             referencing, including
6                 \protect\label \space and \protect\thref.\MessageBreak
7             Cleveref does not accept ntheorem's optional\MessageBreak
8             argument for \protect\label, so it will appear\MessageBreak
9             in the text.  It is recommended to remove the\MessageBreak
10            thref option, \protect\usepackage{cleveref} instead,\MessageBreak
11            and remove any trailing optional arguments for \protect\label%
```

```
12        }%
13     }
14 }
15
16
17 \newbool{LWR@ntheoremmarks}
18 \boolfalse{LWR@ntheoremmarks}
19
20 \DeclareOption{thmmarks}{
21 \booltrue{LWR@ntheoremmarks}
22 \newif\ifsetendmark\setendmarktrue
23 }
24
25
26 \newbool{LWR@ntheoremamsthm}
27 \boolfalse{LWR@ntheoremamsthm}
28
29 \DeclareOption{amsthm}{\booltrue{LWR@ntheoremamsthm}}
30
31
32 \DeclareOption{amsmath}{}
33 \DeclareOption{hyperref}{}
34
35 \LWR@ProvidesPackagePass{ntheorem}[2011/08/15]
```

§ 464.3 **Remembering the theorem style**

Storage for the style being used for new theorems.

```
36 \newcommand{\LWR@newtheoremstyle}{plain}
```

```
37 \AtBeginDocument{
38 \IfPackageLoadedTF{cleveref}{
39 \VerifyCommand[lwarp][ntheorem-cleveref]{\@thm}
40    {24EBE14DBADF0B9992C3EDE1D70BA897}
41 \gdef\@thm#1#2#3{%
42   \if@thmmarks
43     \stepcounter{end\InTheoType ctr}%
44   \fi
45   \renewcommand{\InTheoType}{#1}%
46   \if@thmmarks
47     \stepcounter{curr#1ctr}%
48     \setcounter{end#1ctr}{0}%
49   \fi
50   \refstepcounter[#1]{#2}%  <<< cleveref modification
51   \theorem@prework
52     \LWR@forcenewpage% lwarp


53     \LWR@printpendingfootnotes%                    lwarp


54     \BlockClass{theorembody#1}%\LWR@thisthmstyle% lwarp
55   \trivlist % latex's \trivlist, calling latex's \@trivlist unchanged
56   \ifuse@newframeskips  % cf. latex.ltx for topsepadd: \@trivlist
57     \ifthm@inframe
58       \thm@topsep\theoreminframepreskipamount
59       \thm@topsepadd\theoreminframepostskipamount
60     \else
61       \thm@topsep\theorempreskipamount
62       \thm@topsepadd\theorempostskipamount
```

```
63        \fi
64      \else% oldframeskips
65        \thm@topsep\theorempreskipamount
66        \thm@topsepadd \theorempostskipamount
67        \ifvmode\advance\thm@topsepadd\partopsep\fi
68      \fi
69      \@topsep\thm@topsep
70      \@topsepadd\thm@topsepadd
71      \advance\linewidth -\theorem@indent
72      \advance\linewidth -\theorem@rightindent
73      \advance\@totalleftmargin \theorem@indent
74      \parshape \@ne \@totalleftmargin \linewidth
75      \@ifnextchar[{\@ythm{#1}{#2}{#3}}{\@xthm{#1}{#2}{#3}}
76 }
77 }{% not @ifpackageloaded{cleveref}
78 % \VerifyCommand[lwarp][ntheorem]{\@thm}{0}
79 \gdef\@thm#1#2#3{%
80      \if@thmmarks
81        \stepcounter{end\InTheoType ctr}%
82      \fi
83      \renewcommand{\InTheoType}{#1}%
84      \if@thmmarks
85        \stepcounter{curr#1ctr}%
86        \setcounter{end#1ctr}{0}%
87      \fi
88      \refstepcounter{#2}%
89      \theorem@prework
90       \LWR@forcenewpage% lwarp


91       \LWR@printpendingfootnotes%                               lwarp


92       \BlockClass{theorembody#1}%\LWR@thisthmstyle% lwarp
93      \trivlist % latex's \trivlist, calling latex's \@trivlist unchanged
94      \ifuse@newframeskips  % cf. latex.ltx for topsepadd: \@trivlist
95        \ifthm@inframe
96          \thm@topsep\theoreminframepreskipamount
97          \thm@topsepadd\theoreminframepostskipamount
98        \else
99          \thm@topsep\theorempreskipamount
100         \thm@topsepadd\theorempostskipamount
101       \fi
102     \else% oldframeskips
103       \thm@topsep\theorempreskipamount
104       \thm@topsepadd \theorempostskipamount
105       \ifvmode\advance\thm@topsepadd\partopsep\fi
106     \fi
107     \@topsep\thm@topsep
108     \@topsepadd\thm@topsepadd
109     \advance\linewidth -\theorem@indent
110     \advance\linewidth -\theorem@rightindent
111     \advance\@totalleftmargin \theorem@indent
112     \parshape \@ne \@totalleftmargin \linewidth
113     \@ifnextchar[{\@ythm{#1}{#2}{#3}}{\@xthm{#1}{#2}{#3}}
114 }
115 }
116 }% AtBeginDocument
```

Patched to remember the style being used for new theorems:

```
117 \VerifyCommand[lwarp][ntheorem]{\theoremstyle}
118        {A735E431A49EB3A7B4BABD9AEAEE7E10}
119 \gdef\theoremstyle#1{%
120    \@ifundefined{th@#1}{\@warning
121        {Unknown theoremstyle `#1'. Using `plain'}%
122        \theorem@style{plain}
123          \renewcommand{\LWR@newtheoremstyle}{plain}% lwarp
124          }%
125    {
126      \theorem@style{#1}
127      \renewcommand{\LWR@newtheoremstyle}{#1}% lwarp
128      }
129 }
```

Patched to remember the style for this theorem type, and set it later when the environment is started.

```
130 \VerifyCommand[lwarp][ntheorem]{\@xnthm}{699CB37D7349C4F062B16B9B890FFE90}
131
132 \gdef\@xnthm#1#2[#3]{%
133    \ifthm@tempif
134      \csedef{LWR@thmstyle#1}{\LWR@newtheoremstyle}% lwarp
135      \expandafter\@ifundefined{c@#1}%
136        {\@definecounter{#1}}{}%
137      \@newctr{#1}[#3]%
138      \expandafter\xdef\csname the#1\endcsname{%
139        \expandafter\noexpand\csname the#3\endcsname \@thmcountersep
140          {\noexpand\csname\the\theoremnumbering\endcsname{#1}}}%
141      \expandafter\gdef\csname mkheader@#1\endcsname
142        {\csname setparms@#1\endcsname
143        \@thm{#1}{#1}{#2}
144        }%
145      \global\@namedef{end#1}{\@endtheorem}
146    \AtBeginEnvironment{#1}{\edef\LWR@thisthmstyle{\@nameuse{LWR@thmstyle#1}}}% lwarp
147  \fi
148 }
149
150 \VerifyCommand[lwarp][ntheorem]{\@ynthm}{E0E49F4C2FF76BA3024F2413E2E3DA0D}
151
152 \gdef\@ynthm#1#2{%
153    \ifthm@tempif
154      \csedef{LWR@thmstyle#1}{\LWR@newtheoremstyle}% lwarp
155      \expandafter\@ifundefined{c@#1}%
156        {\@definecounter{#1}}{}%
157      \expandafter\xdef\csname the#1\endcsname
158        {\noexpand\csname\the\theoremnumbering\endcsname{#1}}%
159      \expandafter\gdef\csname mkheader@#1\endcsname
160        {\csname setparms@#1\endcsname
161        \@thm{#1}{#1}{#2}
162        }%
163      \global\@namedef{end#1}{\@endtheorem}
164    \AtBeginEnvironment{#1}{\edef\LWR@thisthmstyle{\@nameuse{LWR@thmstyle#1}}}% lwarp
165  \fi
166 }
167
168 \VerifyCommand[lwarp][ntheorem]{\@othm}{A6D2FCC33AB3F7C7F998399F013FB6A8}
169
170 \gdef\@othm#1[#2]#3{%
171    \@ifundefined{c@#2}{\@nocounterr{#2}}%
172      {\ifthm@tempif
```

```
173       \csedef{LWR@thmstyle#1}{\LWR@newtheoremstyle}% lwarp
174       \global\@namedef{the#1}{\@nameuse{the#2}}%
175       \expandafter\protected@xdef\csname num@addtheoremline#1\endcsname{%
176              \noexpand\@num@addtheoremline{#1}{#3}}%
177       \expandafter\protected@xdef\csname nonum@addtheoremline#1\endcsname{%
178              \noexpand\@nonum@addtheoremline{#1}{#3}}%
179     \theoremkeyword{#3}%
180     \expandafter\protected@xdef\csname #1Keyword\endcsname
181            {\the\theoremkeyword}%
182     \expandafter\gdef\csname mkheader@#1\endcsname
183       {\csname setparms@#1\endcsname
184               \@thm{#1}{#2}{#3}
185         }%
186     \global\@namedef{end#1}{\@endtheorem}
187     \AtBeginEnvironment{#1}{\edef\LWR@thisthmstyle{\@nameuse{LWR@thmstyle#1}}}% lwarp
188   \fi}
189 }
```

§ 464.4 HTML cross-referencing

Mimics a float by incrementing the float counter and generating an HTML anchor.
These are used for list-of-theorem cross-references.

```
190 \newcommand{\LWR@inctheorem}{%
191 \addtocounter{LWR@thisautoid}{1}%
192 \LWR@stoppars%
193 \LWR@htmltag{%
194   a id=\textquotedbl\LWR@print@mbox{autoid-\arabic{LWR@thisautoid}}\textquotedbl%
195 }%
196 \LWR@htmltag{/a}\LWR@orignewline%
197 \LWR@startpars%
198 }
```

§ 464.5 \newtheoremstyle

The following are patched for CSS.

These were in individual files thp.sty for plain, thmb.sty for margin break, etc.
They are gathered together here.

Each theorem is encased in a BlockClass environment of class theorembody<style>.

Each header is encased in an \InlineClass of class theoremheader<style>.

```
199 \VerifyCommand[lwarp][ntheorem]{\newtheoremstyle}{8173F61CEBA45226CD3015E5E258C93D}
200
201 \gdef\newtheoremstyle#1#2#3{%
202   \expandafter\@ifundefined{th@#1}%
203   {\expandafter\gdef\csname th@#1\endcsname{%
204     \def\@begintheorem####1####2{%
205     \LWR@inctheorem% lwarp
206     #2}%
207     \def\@opargbegintheorem####1####2####3{%
208     \LWR@inctheorem% lwarp
209     #3}%
210 }%
211 }%
212 {\PackageError{\basename}{Theorem style #1 already defined}\@eha}
213 }
```

§ 464.6 Standard styles

```
214 \renewtheoremstyle{plain}%
215   {\item[
216     \InlineClass{theoremheaderplain}{##1\ ##2\theorem@separator}]}%
217   {\item[
218     \InlineClass{theoremheaderplain}{##1\ ##2\ (##3)\theorem@separator}]}
219
220 \renewtheoremstyle{break}%
221   {\item[
222     \InlineClass{theoremheaderbreak}{##1\ ##2\theorem@separator}\newline
223     ]}%
224   {\item[
225     \InlineClass{theoremheaderbreak}%
226       {##1\ ##2\ (##3)\theorem@separator}\newline
227     ]}
228
229 \renewtheoremstyle{change}%
230   {\item[
231     \InlineClass{theoremheaderchange}{##2\ ##1\theorem@separator}]}%
232   {\item[
233     \InlineClass{theoremheaderchange}{##2\ ##1\ (##3)\theorem@separator}]}
234
235 \renewtheoremstyle{changebreak}%
236   {\item[
237       \InlineClass{theoremheaderchangebreak}%
238           {##2\ ##1\theorem@separator}\newline
239     ]}%
240   {\item[
241       \InlineClass{theoremheaderchangebreak}%
242           {##2\ ##1\ (##3)\theorem@separator}\newline
243     ]}
244
245 \renewtheoremstyle{margin}%
246   {\item[
247       \InlineClass{theoremheadermargin}{##2 \qquad ##1\theorem@separator}
248     ]}%
249   {\item[
250     \InlineClass{theoremheadermargin}{##2 \qquad ##1\ (##3)\theorem@separator}
251     ]}
252
253 \renewtheoremstyle{marginbreak}%
254   {\item[
255     \InlineClass{theoremheadermarginbreak}%
256       {##2 \qquad ##1\theorem@separator}\newline
257     ]}%
258   {\item[
259     \InlineClass{theoremheadermarginbreak}%
260       {##2 \qquad ##1\ (##3)\theorem@separator}\newline
261     ]}
262
263 \renewtheoremstyle{nonumberplain}%
264   {\item[
265     \InlineClass{theoremheaderplain}{##1\theorem@separator}]}%
266   {\item[
267     \InlineClass{theoremheaderplain}{##1\ (##3)\theorem@separator}]}
268
269 \renewtheoremstyle{nonumberbreak}%
270   {\item[
271     \InlineClass{theoremheaderbreak}{##1\theorem@separator}\newline
```

```
272     ]}%
273 {\item[
274     \InlineClass{theoremheaderbreak}{##1\ (##3)\theorem@separator}\newline
275     ]}

276
277 \renewtheoremstyle{empty}%
278  {\item[]}%
279  {\item[
280     \InlineClass{theoremheaderplain}{##3}]}

281
282 \renewtheoremstyle{emptybreak}%
283  {\item[]}%
284  {\item[
285     \InlineClass{theoremheaderplain}{##3}] \ \newline}
```

§ 464.7 **Additional objects**

The following manually adjust the css for the standard configuration objects which are not a purely plain style:

```
286 \ifbool{LWR@ntheoremamsthm}{}{%
```

Upright text via CSS:

```
287     \newtheoremstyle{plainupright}%
288     {\item[
289         \InlineClass{theoremheaderplain}{##1\ ##2\theorem@separator}]}%
290     {\item[
291         \InlineClass{theoremheaderplain}{##1\ ##2\ (##3)\theorem@separator}]}
```

Upright text and small caps header via CSS:

```
292     \newtheoremstyle{nonumberplainuprightsc}%
293     {\item[
294         \InlineClass{theoremheadersc}{##1\theorem@separator}]}%
295     {\item[
296         \InlineClass{theoremheadersc}{##1\ (##3)\theorem@separator}]}
297 }% not amsthm
```

§ 464.8 **Renewed standard configuration**

The following standard configuration is renewed using the new css:

```
298 \ifbool{LWR@ntheoremamsthm}{}{%

299 \ifx\thm@usestd\@undefined
300 \else
301     \theoremnumbering{arabic}
302     \theoremstyle{plain}
303     \RequirePackage{latexsym}
304     \theoremsymbol{\Box}
305     \theorembodyfont{\itshape}
306     \theoremheaderfont{\normalfont\bfseries}
307     \theoremseparator{}
308     \renewtheorem{Theorem}{Theorem}
309     \renewtheorem{theorem}{Theorem}
310     \renewtheorem{Satz}{Satz}
311     \renewtheorem{satz}{Satz}
```

```
312     \renewtheorem{Proposition}{Proposition}
313     \renewtheorem{proposition}{Proposition}
314     \renewtheorem{Lemma}{Lemma}
315     \renewtheorem{lemma}{Lemma}
316     \renewtheorem{Korollar}{Korollar}
317     \renewtheorem{korollar}{Korollar}
318     \renewtheorem{Corollary}{Corollary}
319     \renewtheorem{corollary}{Corollary}
320
321     \theoremstyle{plainupright}
322     \theorembodyfont{\upshape}
323     \theoremsymbol{\HTMLunicode{25A1}}% UTF-8 white box
324     \renewtheorem{Example}{Example}
325     \renewtheorem{example}{Example}
326     \renewtheorem{Beispiel}{Beispiel}
327     \renewtheorem{beispiel}{Beispiel}
328     \renewtheorem{Bemerkung}{Bemerkung}
329     \renewtheorem{bemerkung}{Bemerkung}
330     \renewtheorem{Anmerkung}{Anmerkung}
331     \renewtheorem{anmerkung}{Anmerkung}
332     \renewtheorem{Remark}{Remark}
333     \renewtheorem{remark}{Remark}
334     \renewtheorem{Definition}{Definition}
335     \renewtheorem{definition}{Definition}
336
337     \theoremstyle{nonumberplainuprightsc}
338     \theoremsymbol{\HTMLunicode{220E}}% UTF-8 end-of-proof
339     \renewtheorem{Proof}{Proof}
340     \renewtheorem{proof}{Proof}
341     \renewtheorem{Beweis}{Beweis}
342     \renewtheorem{beweis}{Beweis}
343     \qedsymbol{\HTMLunicode{220E}}% UTF-8 end-of-proof
344
345     \theoremsymbol{}
346 \fi
347 }% not amsthm
```

§ 464.9 **amsthm option**

Many of the following are not \VerifyCommand because they are not yet defined.

Only if the amsthm option was given:

```
348 \ifbool{LWR@ntheoremamsthm}{
349
350 \VerifyCommand[lwarp][ntheorem-amsthm]{\th@plain}
351        {1FB8AE22BD4090F90D9031CFFC57DAFB}
352 \gdef\th@plain{%
353   \def\theorem@headerfont{\normalfont\bfseries}\itshape%
354   \def\@begintheorem##1##2{%
355        \LWR@inctheorem% lwarp
356      \item[
357   \InlineClass{theoremheaderplain}{##1\ ##2.}
358        ]}%
359   \def\@opargbegintheorem##1##2##3{%
360        \LWR@inctheorem% lwarp
361      \item[
362   \InlineClass{theoremheaderplain}{##1\ ##2\ (##3).}
363        ]}}
364
```

```
365 \VerifyCommand[lwarp][ntheorem-amsthm]{\th@nonumberplain}
366        {3E82A39A222021777BF8221C490E86EC}
367 \gdef\th@nonumberplain{%
368   \def\theorem@headerfont{\normalfont\bfseries}\itshape%
369   \def\@begintheorem##1##2{%
370        \LWR@inctheorem% lwarp
371      \item[
372    \InlineClass{theoremheaderplain}{##1.}
373        ]}%
374   \def\@opargbegintheorem##1##2##3{%
375        \LWR@inctheorem% lwarp
376      \item[
377    \InlineClass{theoremheaderplain}{##1\ (##3).}
378        ]}}
379
380 \gdef\th@definition{%
381   \def\theorem@headerfont{\normalfont\bfseries}\normalfont%
382   \def\@begintheorem##1##2{%
383        \LWR@inctheorem% lwarp
384      \item[
385    \InlineClass{theoremheaderdefinition}{##1\ ##2.}
386        ]}%
387   \def\@opargbegintheorem##1##2##3{%
388        \LWR@inctheorem% lwarp
389      \item[
390    \InlineClass{theoremheaderdefinition}{##1\ ##2\ (##3).}
391        ]}}
392
393 \gdef\th@nonumberdefinition{%
394   \def\theorem@headerfont{\normalfont\bfseries}\normalfont%
395   \def\@begintheorem##1##2{%
396        \LWR@inctheorem% lwarp
397      \item[
398    \InlineClass{theoremheaderdefinition}{##1.}
399        ]}%
400   \def\@opargbegintheorem##1##2##3{%
401        \LWR@inctheorem% lwarp
402      \item[
403    \InlineClass{theoremheaderdefinition}{##1\ (##3).}
404        ]}}
405
406 \gdef\th@remark{%
407   \def\theorem@headerfont{\itshape}\normalfont%
408   \def\@begintheorem##1##2{%
409        \LWR@inctheorem% lwarp
410      \item[
411    \InlineClass{theoremheaderremark}{##1\ ##2.}
412        ]}%
413   \def\@opargbegintheorem##1##2##3{%
414        \LWR@inctheorem% lwarp
415      \item[
416    \InlineClass{theoremheaderremark}{##1\ ##2\ (##3).}
417        ]}}
418
419 \gdef\th@nonumberremark{%
420   \def\theorem@headerfont{\itshape}\normalfont%
421   \def\@begintheorem##1##2{%
422        \LWR@inctheorem% lwarp
423      \item[
424    \InlineClass{theoremheaderremark}{##1.}
```

```
425        ]}%
426  \def\@opargbegintheorem##1##2##3{%
427        \LWR@inctheorem% lwarp
428     \item[
429    \InlineClass{theoremheaderremark}{##1\ (##3).}
430        ]}}
431
432 \gdef\th@proof{%
433  \def\theorem@headerfont{\normalfont\bfseries}\itshape%
434  \def\@begintheorem##1##2{%
435        \LWR@inctheorem% lwarp
436     \item[
437    \InlineClass{theoremheaderproof}{##1.}
438        ]}%
439  \def\@opargbegintheorem##1##2##3{%
440        \LWR@inctheorem% lwarp
441     \item[
442    \InlineClass{theoremheaderproof}{##1\ (##3).}
443        ]}}
444
445
446
447 \newcounter{proof}%
448 \if@thmmarks
449     \newcounter{currproofctr}%
450     \newcounter{endproofctr}%
451 \fi
452
453 \gdef\proofSymbol{\openbox}
454
455 \newcommand{\proofname}{Proof}
456
457 \newenvironment{proof}[1][\proofname]{
458     \th@proof
459     \def\theorem@headerfont{\itshape}%
460     \normalfont
461     \theoremsymbol{\HTMLunicode{220E}}% UTF-8 end-of-proof
462     \@thm{proof}{proof}{#1}
463 }%
464 {\@endtheorem}
465
466 }{}% amsthm option
```

§ 464.10   **Ending a theorem**

Patched for css:

```
467 \let\LWR@origendtheorem\@endtheorem
468 \renewcommand{\@endtheorem}{%
469 \ifbool{LWR@ntheoremmarks}{%
470     \ifsetendmark%
471     \InlineClass{theoremendmark}{\csname\InTheoType Symbol\endcsname}%
472     \setendmarkfalse%
473     \fi%
474 }{}%
475 \LWR@origendtheorem% also does \@endtrivlist
476 \ifbool{LWR@ntheoremmarks}{\global\setendmarktrue}{}%


477     \LWR@printpendingfootnotes%                            lwarp
```

```
478 \endBlockClass%
479 }
```

§ 464.11 **\NoEndMark**

```
480 \gdef\NoEndMark{\global\setendmarkfalse}
```

§ 464.12 **List-of**

Redefined to reuse the float mechanism to add list-of-theorem links:

\thm@thmline {⟨*1: printed type*⟩} {⟨*2: #*⟩} {⟨*3: optional*⟩} {⟨*4: page*⟩}

```
481 \renewcommand{\thm@@thmline@noname}[4]{%
482 \hypertocfloat{1}{theorem}{thm}{#2 #3}{}%
483 }
484
485 \renewcommand{\thm@@thmline@name}[4]{%
486 \hypertocfloat{1}{theorem}{thm}{#1 #2 #3}{}%
487 }
```

This was redefined by ntheorem when loaded, so it is now redefined for lwarp:

```
488 \def\thm@@thmline{\thm@@thmline@name}
```

Patch for css:

```
489 \xpretocmd{\listtheorems}
490     {\LWR@htmlelementclass{nav}{lothm}}
491     {}
492     {\LWR@patcherror{ntheorem}{listtheorems}}
493
494 \xapptocmd{\listtheorems}
495     {\LWR@htmlelementclassend{nav}{lothm}}
496     {}
497     {\LWR@patcherror{ntheorem}{listtheorems}}
```

§ 464.13 **Symbols**

Proof QED symbol:

```
498 \newcommand{\qed}{\qquad\the\qedsymbol}
499
500 \AtBeginDocument{
501 \@ifundefined{LWR@orig@openbox}{
502 \LetLtxMacro\LWR@orig@openbox\openbox
503 \LetLtxMacro\LWR@orig@blacksquare\blacksquare
504 \LetLtxMacro\LWR@orig@Box\Box
505
506 \def\openbox{\text{\HTMLunicode{25A1}}}% UTF-8 white box
507 \def\blacksquare{\text{\HTMLunicode{220E}}}% UTF-8 end-of-proof
508 \def\Box{\text{\HTMLunicode{25A1}}}% UTF-8 white box
509
510 \appto\LWR@restoreorigformatting{%
511 \LetLtxMacro\openbox\LWR@orig@openbox%
512 \LetLtxMacro\blacksquare\LWR@orig@blacksquare%
513 \LetLtxMacro\Box\LWR@orig@Box%
514 }% appto
515 }{}% @ifundefined
516 }% AtBeginDocument
```

§ 464.14 Cross-referencing

\thref {⟨*label*⟩}

517 \newcommand*{\thref}[1]{\cref{#1}}%

File 353 **lwarp-octave.sty**

§ 465 Package **octave**

(*Emulates or patches code by* Andrew A. Cashner.)

octave (*Pkg*) octave is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{octave}[2017/10/31]

Remove the leading 1pt kern:

```
2 \VerifyCommand[lwarp][octave]{\@PrintTicks}{26490A1A3593981987395ED149B4D54D}
3
4 \RenewDocumentCommand{\@PrintTicks}{ m }{%
5 \kern-1pt%  lwarp
6 \@TickNum = #1%
7 \loop
8 \@Tick{}%
9 \advance\@TickNum by -1
10 \ifnum\@TickNum > 0
11 \repeat
12 }
```

Use unicode for the prime character:

```
13 \RenewDocumentCommand{\@Tick}{}{\HTMLunicode{2032}}
```

Catch the inline font:

```
14 \VerifyCommand[lwarp][octave]{\pitch}{3803E3D6B44EDFF8880F06BBE60571D9}
15
16 \RenewDocumentCommand{\pitch}{ m o m }{%
17 \if@OctaveNumber%
18 {%
19     \pitchfont{%
20         \LWR@textcurrentfont{%  lwarp
21             \MakeUppercase{#1}%
22             \IfValueTF{#2}{#2}{}\textsubscript{#3}%
23         }%
24     }%
25 }%
26 \else%
27 {%
28     \pitchfont{%
29         \LWR@textcurrentfont{%  lwarp
30             \@GetOctaveTick{#1}[#2]{#3}%
31         }%
32     }%
33 }%
34 \fi%
```

```
35 }
```

The original was hard to adapt to lwarp's handling of &.

```
36 \StartDefiningTabulars
37 \renewcommand{\octavetable}{%
38 \begin{tabular}{ll}
39 \octaveprimes \pitch{C}{0} & \octavenumbers \pitch{C}{0} \\
40 \octaveprimes \pitch{C}{1} & \octavenumbers \pitch{C}{1} \\
41 \octaveprimes \pitch{C}{2} & \octavenumbers \pitch{C}{2} \\
42 \octaveprimes \pitch{C}{3} & \octavenumbers \pitch{C}{3} \\
43 \octaveprimes \pitch{C}{4} & \octavenumbers \pitch{C}{4} \\
44 \octaveprimes \pitch{C}{5} & \octavenumbers \pitch{C}{5} \\
45 \octaveprimes \pitch{C}{6} & \octavenumbers \pitch{C}{6} \\
46 \octaveprimes \pitch{C}{7} & \octavenumbers \pitch{C}{7} \\
47 \end{tabular}
48 }
49 \StopDefiningTabulars
```

File 354 **`lwarp-orcidlink.sty`**

§ 466 Package **orcidlink**

(*Emulates or patches code by* LEO C. STEIN.)

orcidlink (*Pkg*) orcidlink is patched for use by lwarp.

for HTML output:
```
1 \RequirePackage{lwarp-scalerel}
2
3 \LWR@ProvidesPackagePass{orcidlink}[2024/06/26]

4 \xpretocmd\orcidlogo
5     {\begin{lateximage}[orcidlogo]}
6     {}
7     {\LWR@patcherror{orcidlink}{orcidlogo pre}}
8
9 \xapptocmd\orcidlogo
10    {\end{lateximage}}
11    {}
12    {\LWR@patcherror{orcidlink}{orcidlogo post}}
13
14 \VerifyCommand[lwarp][orcidlink]{\orcidlinkX}
15    {3CDD2DC196F113AE4758968E937D7D08}
16
17 \renewrobustcmd\orcidlinkX[3]{%
18    \href%
19        {https://orcid.org/#2}%
20        {%
21            \ifstrempty{#1}{}{#1\,}%
22            \begin{lateximage}*[orcid #2]?%
23            \orcidlogo%
24            \end{lateximage}%    lwarp
25            \ifstrempty{#3}{}{\,#3}%
26        }%
27 }
28
29 \begin{warpMathJax}
```

```
30 \CustomizeMathJax{\newcommand{\orcidlink}[1]{}}
31 \end{warpMathJax}
```

File 355   **`lwarp-overpic.sty`**

§ 467   Package   # overpic

(*Emulates or patches code by* ROLF NIEPRASCHK.)

overpic (*Pkg*)   overpic is patched for use by lwarp.

⚠ scaling   The macros \overpicfontsize and \overpicfontskip are used during HTML gen-
eration. These are sent to \fontsize to adjust the font size for scaling differences
between the print and HTML versions of the document. Renew these macros before
using the overpic and Overpic environments.

See section 90.2 for the print-mode version of \overpicfontsize and \overpicfontskip.

for HTML output: 1 \LWR@ProvidesPackagePass{overpic}[2017/10/06]

```
2 \newcommand*{\overpicfontsize}{12}
3 \newcommand*{\overpicfontskip}{14}
4
5 \BeforeBeginEnvironment{overpic}{%
6     \begin{lateximage}%
7     \fontsize{\overpicfontsize}{\overpicfontskip}%
8     \selectfont%
9 }
10
11 \AfterEndEnvironment{overpic}{\end{lateximage}}
12
13 \BeforeBeginEnvironment{Overpic}{%
14     \begin{lateximage}%
15     \fontsize{\overpicfontsize}{\overpicfontskip}%
16     \selectfont%
17 }
18
19 \AfterEndEnvironment{Overpic}{\end{lateximage}}
```

File 356   **`lwarp-pagegrid.sty`**

§ 468   Package   # pagegrid

pagegrid (*Pkg*)   pagegrid is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{pagegrid}[2016/05/16]

2 \newcommand*{\pagegridsetup}[1]{}

File 357   **lwarp-pagenote.sty**

§ 469   Package   # pagenote

pagenote (*Pkg*)   **pagenote** works as-is, but the page option is disabled.

⚠ labels   Note that labels in page notes do not appear as expected, even in the print version.

for HTML output: 1 \DeclareOption{page}{}
2 \LWR@ProvidesPackagePass{pagenote}[2009/09/03]

For MATHJAX:

3 \begin{warpMathJax}
4 \appto\LWR@syncnotenumbers{\LWR@synconenotenumber{LWRpagenote}{\thepagenote}}
5 \CustomizeMathJax{\def\LWRpagenote{1}}
6 \CustomizeMathJax{\newcommand{\pagenote}[2][\LWRpagenote]{{}^{\mathrm{#1}}}}
7 \end{warpMathJax}

There is no \pagenotemark, so the following are not required:

\providecommand{\pagenotename}{pagenote}
\appto\LWR@syncnotenames{\LWR@synconenotename{LWRpagenote}{\pagenotename}}

File 358   **lwarp-pagesel.sty**

§ 470   Package   # pagesel

pagesel (*Pkg*)   **pagesel** is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{pagesel}[2016/05/16]

File 359   **lwarp-paralist.sty**

§ 471   Package   # paralist

(*Emulates or patches code by* BERND SCHANDL.)

paralist (*Pkg*)   **paralist** is supported with minor changes.

for HTML output: 1 \LWR@ProvidesPackagePass{paralist}[2017/01/22]

The compact environments are identical to the regular ones:

2 \LetLtxMacro\compactitem\itemize
3 \LetLtxMacro\compactenum\enumerate
4 \LetLtxMacro\compactdesc\description
5 \LetLtxMacro\endcompactitem\enditemize
6 \LetLtxMacro\endcompactenum\endenumerate
7 \LetLtxMacro\endcompactdesc\enddescription

For the inline environments, revert \item to its original print-mode version:

```
8 \AtBeginEnvironment{inparaitem}{\LetLtxMacro\item\LWR@origitem}
9 \AtBeginEnvironment{inparaenum}{\LetLtxMacro\item\LWR@origitem}
10 \AtBeginEnvironment{inparadesc}{\LetLtxMacro\item\LWR@origitem}
```

Manual formatting of the description labels:

```
11 \def\paradescriptionlabel#1{{\normalfont\textbf{#1}}}
```

File 360 **`lwarp-parallel.sty`**

§ 472   Package   **parallel**

(*Emulates or patches code by* Matthias Eckermann.)

parallel (*Pkg*)   **parallel** is emulated.

Package options are ignored. Footnotes are treated as normal lwarp footnotes.

Environment option c gives side-by-side <div>s of class minipage, each of whose width is a percent depending on the given left and right widths, proportional to \linewidth.

Inside each environment, \linewidth and \textwidth are set for the print-output sizes.

for HTML output: Discard all options for lwarp-parallel:

```
1 \LWR@ProvidesPackageDrop{parallel}[2003/04/13]
```

```
2 \newcounter{LWR@parallel@Lwidth}
3 \newcounter{LWR@parallel@Rwidth}
4 \newcommand*{\LWR@parallel@border}
5
6 \newenvironment*{Parallel}[3][]%
7     {%
8         \LWR@printpendingfootnotes%
9         \setlength{\linewidth}{\LWR@userstextwidth}%
10        \setlength{\textwidth}{\LWR@userstextwidth}%
11        \renewcommand*{\LWR@parallel@border}{}%
12        \ifstrequal{#1}{v}%
13            {%
14            \renewcommand*{\LWR@parallel@border}{ ; border-left: 2px solid black}%
15            }%
16            {}%
17        \ifblank{#2}{%
18            \ifblank{#3}{% {}{}
19                \setcounter{LWR@parallel@Lwidth}{45}%
20                \setcounter{LWR@parallel@Rwidth}{45}%
21            }% {}{}
22            {% {}{x}
23                \setlength{\LWR@templengthone}{\linewidth-#3}%
24                \setcounter{LWR@parallel@Lwidth}{%
25                    90*\ratio{\LWR@templengthone}{\linewidth}%
26                }%
27                \setcounter{LWR@parallel@Rwidth}{%
```

```
28                    90*\ratio{#3}{\linewidth}%
29              }%
30          }% {}{x}
31      }% #2 blank
32      {% #2 non-blank
33          \ifblank{#3}{% {x}{}
34              \setcounter{LWR@parallel@Lwidth}{%
35                  90*\ratio{#2}{\linewidth}%
36              }%
37              \setlength{\LWR@templengthone}{\linewidth-#2}%
38              \setcounter{LWR@parallel@Rwidth}{%
39                  90*\ratio{\LWR@templengthone}{\linewidth}%
40              }%
41          }% {x}{}
42          {% {x}{x}
43              \setcounter{LWR@parallel@Lwidth}{%
44                  90*\ratio{#2}{\linewidth}%
45              }%
46              \setcounter{LWR@parallel@Rwidth}{%
47                  90*\ratio{#3}{\linewidth}%
48              }%
49          }% {x}{x}
50      }% #2 non-blank
51  }
52  {%
53      \ParallelAtEnd%
54      \renewcommand*{\ParallelAtEnd}{}%
55      \LWR@printpendingfootnotes%
56  }
57
58  \newcommand*{\ParallelLText}[1]{%
59      \begin{BlockClass}[%
60          width:\arabic{LWR@parallel@Lwidth}\% ; % space
61          padding: .5ex 1\% ; % space
62      ]{minipage}%
63      #1%
64      \end{BlockClass}%
65  }
66
67  \newcommand*{\ParallelRText}[1]{%
68      \begin{BlockClass}[%
69          width:\arabic{LWR@parallel@Rwidth}\% ; % space
70          padding: .5ex 1\% ; % space
71          \LWR@parallel@border%
72      ]{minipage}%
73      #1%
74      \end{BlockClass}%
75  }
76
77  \newcommand*{\ParallelPar}{\LWR@printpendingfootnotes}
78
79  \newcommand*{\ParallelAtEnd}{}
```

File 361 **`lwarp-parcolumns.sty`**

§ 473   Package # parcolumns

(*Emulates or patches code by* Jonathan Sauer.)

parcolumns (*Pkg*)   **parcolumns** is emulated.

rulebetween is honored. The other keys are ignored, including colwidths.

Each column is placed inside a <div> of class minipage, each of whose width is fixed at 85% divided by the number of columns. In most cases, this results in side-by-side minipages adapting to the browser width. Inside each minipage, \linewidth, \textwidth, and \textheight are set for a virtual 6×9 inch page, with \linewidth divided by the number of columns.

for HTML output: Discard all options for lwarp-parcolumns:

```
1 \RequirePackage{keyval}%
2
3 \LWR@ProvidesPackageDrop{parcolumns}[2004/11/25]

4 \newcounter{LWR@parcolumns@numcols}
5 \newcounter{LWR@parcolumns@thiscol}
6 \newcounter{LWR@parcolumns@width}
7 \newbool{LWR@parcolumns@started}
8 \newbool{LWR@parcolumns@rule}
9
10 \define@key{LWRparcols}{colwidths}{}
11 \define@key{LWRparcols}{distance}{}
12 \define@key{LWRparcols}{rulebetween}[true]{%
13     \setbool{LWR@parcolumns@rule}{#1}%
14 }
15 \define@key{LWRparcols}{nofirstindent}{}
16 \define@key{LWRparcols}{sloppy}{}
17 \define@key{LWRparcols}{sloppyspaces}{}
18
19 \newenvironment*{parcolumns}[2][]
20     {%
21         \begin{LWR@setvirtualpage}*[#2]%
22         \setcounter{LWR@parcolumns@numcols}{#2}%
23         \setcounter{LWR@parcolumns@thiscol}{1}%
24         \boolfalse{LWR@parcolumns@started}%
25         \boolfalse{LWR@parcolumns@rule}%
26         \setcounter{LWR@parcolumns@width}{%
27             85/#2
28         }%
29         \setkeys{LWRparcols}{#1}%
30     }
31     {%


32         \colplacechunks%
33         \end{LWR@setvirtualpage}%
34     }
35
36 \newcommand{\LWR@parcolumns@onecol}[1]{%
37     \ifbool{LWR@parcolumns@started}%
38         {}%
39         {%
40             \LWR@htmldivclass{parcolumns}%
41             \booltrue{LWR@parcolumns@started}%
42         }%
43     \ifboolexpr{%
44         bool {LWR@parcolumns@rule} and
45         test {%
```

```
46              \ifnumgreater
47                  {\value{LWR@parcolumns@thiscol}}
48                  {1}
49          }%
50      }%
51          {\renewcommand{\LWR@tempone}{ ; border-left: 2px solid black}}%
52          {\renewcommand{\LWR@tempone}{}}%
53      \begin{BlockClass}[%
54          width:\arabic{LWR@parcolumns@width}\% ; % space
55          padding: .5ex 1\% ; % space
56          \LWR@tempone%
57      ]{minipage}%
58      #1%
59      \end{BlockClass}%
60      \addtocounter{LWR@parcolumns@thiscol}{1}%
61 }
62
63 \newcommand{\colchunk}[2][\value{LWR@parcolumns@thiscol}]{%
64      \whileboolexpr{%
65          test {%
66              \ifnumcomp%
67                  {\value{LWR@parcolumns@thiscol}}
68                  {<}
69                  {#1}%
70          }%
71      }{%
72          \LWR@parcolumns@onecol{}%
73      }%
74      \LWR@parcolumns@onecol{#2}%
75 }
76
77 \newcommand*{\colplacechunks}{%
78      \ifbool{LWR@parcolumns@started}%
79          {%
80              \LWR@htmldivclassend{div}%
81              \boolfalse{LWR@parcolumns@started}%
82          }%
83          {}%
84      \setcounter{LWR@parcolumns@thiscol}{1}%
85 }
```

File 362   **`lwarp-parnotes.sty`**

§ 474   Package   # parnotes

(*Emulates or patches code by* Chelsea Hughes.)

parnotes (*Pkg*)   parnotes is supported with some patches.

for HTML output: 1 \LWR@ProvidesPackagePass{parnotes}[2019/07/23]

2 \VerifyCommand[lwarp][parnotes]{\PN@parnote@real}{91361D751D6393BA644478FDE4A764DA}
3
4 \long\def\PN@parnote@real#1#2{%
5   \parnotemark{#1}%
6   % Unless this is the first parnote in \PN@text, add a separator first
7   \unless\ifx\PN@text\@empty\g@addto@macro\PN@text{\parnoteintercmd}\fi

```
8      % Redefine \@currentlabel to the parnote label, so \label works
9      \g@addto@macro\PN@text{%
10 %       \phantomsection%
11       \def\@currentlabel{#1}%
12       \def\cref@currentlabel{%          lwarp
13           [parnotemark][\arabic{parnotemark}][]\theparnotemark%
14       }%
15     }%
16     \g@addto@macro\PN@text{%
17       \LWR@textcurrentfont{%          lwarp
18           \parnotemark{#1}\nolinebreak\thinspace#2%
19       }%
20     }%
21 }
22
23 \VerifyCommand[lwarp][parnotes]{\PN@parnotes@real}{AF1257823BFCBC31ADDA4AAE1F3F3710}
24
25 \def\PN@parnotes@real{%
26 \ifPN@inparnotes
27 \else
28     \LWR@stoppars%
```

Avoid nested paragraphs:

```
29     \addtocounter{LWR@spandepth}{1}%

30     % We call \par later, so this avoids recursion with \PN@parnotes@auto
31     \PN@inparnotestrue
32 %      \unless\ifvmode\par\fi
33     % Avoid page breaks between a paragraph and its parnotes
34 %      \nopagebreak\addvspace{\parnotevskip}%
35     \begin{BlockClass}(note){footnotes}%    lwarp
36     \leavevmode\LWR@orignewline%
```

Typeset the parnote inside its own group to avoid global changes:

```
37     {%
38         \parnotefmt{\PN@text}%
39     }%
40     \leavevmode\LWR@orignewline%
41     \end{BlockClass}%                        lwarp

42     \leavevmode\LWR@orignewline%
43     \global\def\PN@text{}%
44     %
45     % These can be enabled or disabled by package options
46     %
47     \PN@disable@indent
48     \PN@reset@optional
49     \PN@inparnotesfalse
```

Reenable normal paragraph handling:

```
50     \addtocounter{LWR@spandepth}{-1}%
51 \fi
52 }
```

```
53 \newbool{LWR@parnotes@doingauto}
54 \boolfalse{LWR@parnotes@doingauto}
```

```
55 \VerifyCommand[lwarp][parnotes]{\PN@parnotes@auto}{08CC1722ABA55FA01D64F2B29C919D70}
56
57 \def\PN@parnotes@auto{%
58     \ifbool{LWR@parnotes@doingauto}{
59         \ifx\@currenvir\@PN@autopn
60             \unless\ifPN@inparnotes
61                 \unless\ifx\PN@text\@empty
62                     \expandafter\PN@parnotes@real
63                 \fi
64             \fi
65         \fi
66     }{}%
67 }
```

Replace original logic due to the use of new LATEXparagraph hook handling:

```
68 \renewenvironment{autopn}%
69     {\booltrue{LWR@parnotes@doingauto}}
70     {\PN@parnotes@auto}%
```

If **cleveref** is in use, name the new notes:

```
71 \AtBeginDocument{
72     \ifdef{\crefname}{
73         \crefname{parnotemark}{paragraph note}{paragraph notes}
74         \Crefname{parnotemark}{Paragraph note}{Paragraph notes}
75     }{}
76 }
```

To nullify the footnotes where necessary:

```
77 \apptocmd{\LWR@nullifyfootnotes}{%
78     \renewcommand{\parnote}[2][]{}%
79     \renewcommand\parnotemark[1]{}%
80 }{}{}
```

For MATHJAX:

```
81 \begin{warpMathJax}
82 \providecommand{\parnotename}{parnote}
83 \appto\LWR@syncnotenumbers{%
84     \addtocounter{parnotemark}{-1}% specific to parnotes
85     \LWR@synconenotenumber{LWRparnote}{\theparnotemark}%
86     \addtocounter{parnotemark}{1}% specific to parnotes
87 }
88 \appto\LWR@syncnotenames{\LWR@synconenotename{LWRparnote}{\parnotename}}
89 \CustomizeMathJax{\def\LWRparnote{1}}
90 \CustomizeMathJax{\newcommand{\parnote}[2][\LWRparnote]{{}^{\mathrm{#1}}}}
91 \CustomizeMathJax{\newcommand{\parnotemark}[1][\LWRparnote]{{}^{\mathrm{#1}}}}
92 \end{warpMathJax}
```

File 363 **`lwarp-parskip.sty`**

§ 475 Package **parskip**

parskip (*Pkg*) parskip is ignored.

for HTML output: Discard all options for lwarp-parskip.

1 \LWR@ProvidesPackageDrop{parskip}[2001/04/09]

File 364 **`lwarp-pbalance.sty`**

§ 476 Package **pbalance**

pbalance (*Pkg*) pbalance is ignored.

for HTML output:
1 \RequirePackage{balance}
2
3 \LWR@ProvidesPackageDrop{pbalance}[2022/07/28]

4 \newcommand\shrinkLastPage[1]{}
5 \newcommand\balancePageNum[1]{}
6 \newcommand\nopbalance{}

File 365 **`lwarp-pbox.sty`**

§ 477 Package **pbox**

(*Emulates or patches code by* SIMON LAW.)

pbox (*Pkg*) pbox is emulated.

for HTML output:
1 \LWR@ProvidesPackageDrop{pbox}[2011/12/07]

2 \NewDocumentCommand{\pbox}{O{t} O{} O{t} m +m}{%
3 \global\booltrue{LWR@minipagefullwidth}%
4 \parbox[#1][#2][#3]{#4}{#5}%
5 }
6
7 \newcommand{\settominwidth}[3][\columnwidth]{%
8 \settowidth{#2}{#3}%
9 }
10
11 \newcommand{\widthofpbox}[1]{%
12 \widthof{#1}%
13 }

File 366 **`lwarp-pdfcol.sty`**

§ 478   Package   # pdfcol

pdfcol (*Pkg*)   pdfcol is ignored.

1 `\LWR@ProvidesPackageDrop{pdfcol}[2018/11/01]`
                    2
                    3 `\ltx@newif\ifpdfcolAvailable`
                    4 `\pdfcolAvailablefalse`
                    5
                    6 `\def\pdfcolErrorNoStacks{`
                    7 `    \PackageInfo{lwarp-pdfcol}{Ignoring pdfcol for HTML output.}`
                    8 `}`
                    9
                   10 `\def\pdfcolInitStack#1{}%`
                   11
                   12 `\long\def\pdfcolIfStackExists#1#2#3{#3}%`
                   13
                   14 `\def\pdfcolSwitchStack#1{}%`
                   15
                   16 `\def\pdfcolSetCurrentColor{}%`
                   17
                   18 `\def\pdfcolSetCurrent#1{}%`

File 367   **`lwarp-pdfcolfoot.sty`**

§ 479   Package   # pdfcolfoot

pdfcolfoot (*Pkg*)   pdfcolfoot is ignored.

1 `\LWR@ProvidesPackageDrop{pdfcolfoot}[2016/05/16]`
                    2
                    3 `\newcommand*{\pdfcolfoot@switch}{}`
                    4
                    5 `\newcommand*{\pdfcolfoot@current}{}`

File 368   **`lwarp-pdfcolmk.sty`**

§ 480   Package   # pdfcolmk

pdfcolmk (*Pkg*)   pdfcolmk is ignored.

1 `\LWR@ProvidesPackageDrop{pdfcolmk}[2016/05/16]`

File 369   **`lwarp-pdfcolparallel.sty`**

§ 481   Package   # pdfcolparallel

pdfcolparallel (*Pkg*)   pdfcolparallel is ignored.

for HTML output: 1 \RequirePackage{keyval}%
2
3 \LWR@ProvidesPackageDropA{pdfcolparallel}{2016/05/16}

Pass options to parallel:

4 \DeclareOption*{%
5   \PassoptionsToPackage{\CurrentOption}{parallel}%
6 }

Process the options:

7 \LWR@ProvidesPackageDropB

Require parallel with the given options:

8 \RequirePackage{parallel}[2003/04/13]

Ignore the new key:

9 \define@key{parallel}{rulebetweencolor}{}

File 370   **`lwarp-pdfcolparcolumns.sty`**

§ 482   Package   # pdfcolparcolumns

pdfcolparcolumns (*Pkg*)   pdfcolparcolumns is ignored.

for HTML output: 1 \LWR@ProvidesPackageDropA{pdfcolparcolumns}{2016/05/16}

Pass options to parcolumns:

2 \DeclareOption*{%
3   \PassoptionsToPackage{\CurrentOption}{parcolumns}%
4 }

Process the options:

5 \LWR@ProvidesPackageDropB

Require parcolumns with the given options:

6 \RequirePackage{parcolumns}[2004/11/25]

Ignore the new key:

7 \define@key{LWRparcols}{rulebetweencolor}{}

File 371 **lwarp-pdfcomment.sty**

§ 483   Package **pdfcomment**

pdfcomment (*Pkg*)   pdfcomment is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{pdfcomment}[2016/06/13]

2 \newenvironment{pdfsidelinecomment}[2][]{}{}
3 \newcommand{\pdfcomment}[2][]{}
4 \newcommand{\pdfmargincomment}[2][]{}
5 \newcommand{\pdfmarkupcomment}[3][]{#2}
6 \newcommand{\pdffreetextcomment}[2][]{}
7 \newcommand{\pdfsquarecomment}[2][]{}
8 \newcommand{\pdfcirclecomment}[2][]{}
9 \newcommand{\pdflinecomment}[2][]{}
10 \newcommand{\pdftooltip}[3][]{#2}
11 \newcommand{\pdfcommentsetup}[2][]{}
12 \newcommand{\listofpdfcomments}[1][]{}
13 \newcommand{\setliststyle}[1]{}
14 \newcommand{\defineliststyle}[2]{}
15 \newcommand{\defineavatar}[2]{}
16 \newcommand{\definestyle}[2]{}

For MATHJAX:

17 \begin{warpMathJax}
18 \CustomizeMathJax{\newcommand{\pdfmarkupcomment}[3][]{#2}}
19 \CustomizeMathJax{\newcommand{\pdftooltip}[3][]{#2}}
20 \end{warpMathJax}

File 372   **lwarp-pdfcrypt.sty**

§ 484   Package **pdfcrypt**

pdfcrypt (*Pkg*)   pdfcrypt is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{pdfcrypt}[2016/05/16]

2 \newcommand*{\pdfcryptsetup}[1]{}

File 373   **lwarp-pdflscape.sty**

§ 485   Package **pdflscape**

pdflscape (*Pkg*)   pdflscape is ignored.

Discard all options for lwarp-pdflscape:

```
1 \LWR@ProvidesPackageDrop{pdflscape}[2019/12/05]
```

```
2 \let\landscape\relax
3 \let\endlandscape\relax
4
5 \newenvironment*{landscape}{}{}
```

File 374   `lwarp-pdfmarginpar.sty`

§ 486   Package   # pdfmarginpar

pdfmarginpar (*Pkg*)   pdfmarginpar is ignored.

```
1 \LWR@ProvidesPackageDrop{pdfmarginpar}[2011/08/05]
```

```
2 \newcommand{\pdfmarginpar}[2][]{}
3 \newcommand{\pdfmarginparset}[1]{}
```

File 375   `lwarp-pdfpages.sty`

§ 487   Package   # pdfpages

(*Emulates or patches code by* ANDREAS MATTHIAS.)

pdfpages (*Pkg*)   pdfpages is patched for use by lwarp.

Option `link` and `linkname` work:

```
\hyperlink{<filename>.pdf.<pagenubmer>}{some text}
\hyperlink{<linkname>.<pagenubmer>}{some text}
```

Options which make no sense in HTML are disabled.

```
1 \LWR@ProvidesPackagePass{pdfpages}[2024-10-28]
```

Disable option which have no meaning for HTML output:

```
2 \define@key{pdfpages}{fitpaper}[false]{}
3 \define@key{pdfpages}{landscape}[false]{}
4 \define@key{pdfpages}{openright}[false]{}
5 \define@key{pdfpages}{signature}{}
6 \define@key{pdfpages}{signature*}{}
7 \define@key{pdfpages}{booklet}[false]{}
8 \define@key{pdfpages}{rotateoversize}[false]{}
9 \define@key{pdfpages}{doublepages}[false]{}
10 \define@key{pdfpages}{doublepagestwist}[false]{}
11 \define@key{pdfpages}{doublepagestwistodd}[false]{}
12 \define@key{pdfpages}{doublepagestwist*}[false]{}
13 \define@key{pdfpages}{doublepagestwistodd*}[false]{}
```

14 \define@key{pdfpages}{duplicatepages}[2]{}
15 \define@key{pdfpages}{thread}[false]{}
16 \define@key{pdfpages}{threadname}{}
17 \define@key{pdfpages}{linkfit}{}
18 \define@key{pdfpages}{linktodoc}[false]{}
19 \define@key{pdfpages}{linktodocfit}{}
20 \define@key{pdfpages}{linkfilename}{}
21 \define@key{pdfpages}{survey}[false]{}
22 \define@key{pdfpages}{survey-nolink}[false]{}
23 \define@key{pdfpages}{newwindow}[false]{}

Use print mode while measuring the page numbers:

24 \xpretocmd{\AM@getpagecount}{\LWR@restoreorigformatting}{}{}

Emulate a bit of **eso-pic**:

25 \newif\ifESO@texcoord
26
27 \newcommand{\ESO@HookIIBG}{}
28
29 \renewcommand{\AM@AddToShipoutPicture}{\g@addto@macro\ESO@HookIIBG}
30
31 \renewcommand{\ClearShipoutPicture}{}

\LWR@esopic@newpage   At each \newpage.

32 \newcommand*{\LWR@esopic@newpage}{%

Is there something to draw?

33 \ifdefvoid{\ESO@HookIIBG}%
34 {}%
35 {%

If the `link` option was specified, add a hyper target:

36   \ifAM@link%
37   \hypertarget{\AM@linkname.\AM@page}{}%
38   \fi%

Draw inside a picture environment of the size of a virtual page:

39   \begingroup%
40   \setlength{\unitlength}{1in}%
41   \begin{picture}(8,10.5)%
42   \ESO@HookIIBG%
43   \end{picture}%
44   \endgroup%
45   \global\let\ESO@HookIIBG\@empty%
46 }
47 }

\AM@output   Patched to use \LWR@esopic@newpage.

48 \VerifyCommand[lwarp][pdfpages]{\AM@output@i}{99E594CC8DCDD915D63CB1C8E41BE427}
49
50 \xpatchcmd{\AM@output@i}
51   {\clearpage}%
52   {\LWR@esopic@newpage}
53   {}

```
54        {\LWR@patcherror{pdfpages}{AM@output-1}}
55
56 \xpatchcmd{\AM@output@i}
57        {\clearpage}%
58        {\LWR@esopic@newpage}
59        {}
60        {\LWR@patcherror{pdfpages}{AM@output-2}}
61
62 \xpatchcmd{\AM@output@i}
63        {\newpage}
64        {\LWR@esopic@newpage}
65        {}
66        {\LWR@patcherror{pdfpages}{AM@output-3}}
```

\includepdf   Patched to set the user's paper size.

```
67 \xpretocmd{\includepdf}{%
68        \begingroup%
69        \setlength{\paperwidth}{\LWR@userspaperwidth}%
70        \setlength{\paperheight}{\LWR@userspaperheight}%
71 }{}{}
72
73 \xapptocmd{\includepdf}{%
74        \endgroup%
75 }{}{}
```

\includepdfmerge   Patched to set the user's paper size.

```
76 \xpretocmd{\includepdfmerge}{%
77        \begingroup%
78        \setlength{\paperwidth}{\LWR@userspaperwidth}%
79        \setlength{\paperheight}{\LWR@userspaperheight}%
80 }{}{}
81
82 \xapptocmd{\includepdfmerge}{%
83        \endgroup%
84 }{}{}
```

\AM@hyper@begin@i   Hyper links are created by \LWR@esopic@newpage, so don't create them here:

```
85 \renewcommand{\AM@hyper@begin@i}{}
```

File 376   **lwarp-pdfprivacy.sty**

§ 488   Package   **pdfprivacy**

pdfprivacy (*Pkg*)   pdfprivacy is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{pdfprivacy}[2017/12/03]

File 377 **`lwarp-pdfrender.sty`**

§ 489   Package   # pdfrender

pdfrender (*Pkg*)   pdfrender is allowed during HTML, but it has no effect on HTML text output.
pdfrender is enabled for use with xfakebold, and it is enabled during HTML so
that it may be in use when an SVG math image is started. I.e. xfakebold's \setBold
may be used outside of a math expression and still be detected when the math
begins.

The lwarp-pdfrender package is present because it used to disable pdfrender, so
this newer version is to overwrite older versions.

for HTML output: 1 \LWR@ProvidesPackagePass{pdfrender}[2019/12/29]

File 378   **`lwarp-pdfsync.sty`**

§ 490   Package   # pdfsync

(*Emulates or patches code by* J. Laurens.)

pdfsync (*Pkg*)   pdfsync is ignored.

for HTML output: Discard all options for lwarp-pdfsync:

1 \LWR@ProvidesPackageDrop{pdfsync}[2008/01/26]

2 \newcommand*{\pdfsync}{}
3 \newcommand*{\pdfsyncstart}{}
4 \newcommand*{\pdfsyncstop}{}

File 379   **`lwarp-pdftricks.sty`**

§ 491   Package   # pdftricks

(*Emulates or patches code by* C. V. Radhakrishnan, C. V. Rajagopal, Antoine Chambert-Loir.)

pdftricks (*Pkg*)   pdftricks is patched for use by lwarp.

⚠ convert image files   The pdftricks image files <jobname>-fig*.pdf must be converted to .svg, or else a
missing file error will occur. The image files must also be converted again whenever
they change. To convert the images:

Enter ⇒   **`lwarpmk pdftosvg <jobname>-fig*.pdf`**

for HTML output: 1 \LWR@ProvidesPackagePass{pdftricks}[2003/08/10]

Reuse the print-mode images:

2 \def\PDFTfigname{\BaseJobname-fig\thepsfig}

If the .pdf images have not yet been converted to .svg then an error about a missing file will occur. Warn the user to convert the images.

```
3 \PackageWarning{lwarp-pdftricks}{%
4 When the pdftricks images change,
5 remember to convert PDF images to SVG using 'lwarpmk pdftosvg *-fig.pdf',
6 }
7
8 \AfterEndDocument{\typeout{***}}
9 \AfterEndDocument{\typeout{*** Note: If pdftricks images are not found, new, or updated,}}
10 \AfterEndDocument{\typeout{*** \space use 'lwarpmk pdftosvg \BaseJobname-fig*.pdf'}}
11 \AfterEndDocument{\typeout{***}}
```

File 380   **lwarp-pdfx.sty**

§ 492   Package   # pdfx

pdfx (*Pkg*)   pdfx is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{pdfx}[2017/05/18]

File 381   **lwarp-perpage.sty**

§ 493   Package   # perpage

(*Emulates or patches code by* DAVID KASTRUP.)

perpage (*Pkg*)   perpage is mostly ignored, but support is added for footnote counters.

There is no page number in HTML, so most counters are not reset. If the document redefines \the<countername> to include \theperpage, it is necessary to place that redefininition inside a warpprint environment to avoid modifying the HTML defintions.

\AddAbsoluteCounter must not be inside warpprint, as the counter must be added for HTML also, although it is not incremented.

footnote numbering   To have footnote numbers reset each time footnotes are printed:

    \setcounter{footnoteReset}{1}

For bigfoot, manyfoot, or perpage:

    \MakePerPage{footnoteX}
    — or —
    \MakeSortedPerPage{footnoteX}

The footnotes are reset when they are printed, according to section level as set by FootnoteDepth, which is not necessarily by HTML page. This is recommended for \alph, \Alph, or \fnsymbol footnotes, due to the limited number of symbols which are available.

for HTML output: 1 \LWR@ProvidesPackageDrop{perpage}[2014/10/25]

```
2 \newcommand\AddAbsoluteCounter[1]
3 {
4     \@ifundefined{c@abs#1}{%
5         \expandafter\newcount\csname c@abs#1\endcsname
6         \global\value{abs#1}\@ne
7 %       \global\expandafter\let\csname cl@abs#1\endcsname\@empty
8         \expandafter\xdef\csname theabs#1\endcsname{%
9 %           \noexpand\number \csname c@abs#1\endcsname
10        }%
11 %      \global\@namedef{c@pabs@#1}{\pp@cl@begin
12 %      \stepcounter{abs#1}%
13 %      \pp@cl@end}%
14 %      \@addtoreset{pabs@#1}{#1}
15    }
16    {}
17 }
18
19 \AddAbsoluteCounter{page}
20 \def\theabspage{1}
21
22 \newcommand*\MakePerPage[2][1]{%
23    \ifltxcounter{#2Reset}{%
24        \setcounter{#2Reset}{#1}%
25    }{
26
27 }%
28 }
29
30 \newcommand*\MakeSorted[1]{}
31
32 \newcommand*\MakeSortedPerPage[2][1]{%
33    \ifltxcounter{#2Reset}{%
34        \setcounter{#2Reset}{#1}%
35    }{
36 }%
37 }
38
39 \newcommand*{\theperpage}{1}
```

File 382   `lwarp-pfnote.sty`

§ 494   Package   # **pfnote**

pfnote (*Pkg*)   **pfnote** is ignored.

pfnote

⚠ pfnote numbers   While emulating **pfnote**, lwarp is not able to reset HTML footnote numbers per page number to match the printed version, as HTML has no concept of page numbers. lwarp therefore uses continuous footnote numbering even for **pfnote**.

for HTML output: 1 \LWR@ProvidesPackageDrop{pfnote}[1999/07/14]

File 383   `lwarp-phfqit.sty`

§ 495   Package   # phfqit

(*Emulates or patches code by* PHILIPPE FAIST.)

phfqit (*Pkg*)   phfqit is patched for use by lwarp.

for HTML output:
```
1 \LWR@ProvidesPackagePass{phfqit}[2017/08/16]

2 \LetLtxMacro\LWR@origbitstring\bitstring
3
4 \renewcommand\bitstring[1]{%
5 \InlineClass[%
6    text-decoration: overline underline ;
7 ]{bitstring}{#1}%
8 % \phfqit@bitstring{#1}%
9 }
10
11 \appto\LWR@restoreorigformatting{%
12 \LetLtxMacro\bitstring\LWR@origbitstring%
13 }
```

File 384   `lwarp-physics.sty`

§ 496   Package   # physics

(*Emulates or patches code by* SERGIO C. DE LA BARRERA.)

physics (*Pkg*)   physics works as-is for HTML with SVG math.

For MATHJAX, the MATHJAX V3 physics extension is used.

for HTML output:
```
1 \LWR@ProvidesPackagePass{physics}% No date is provided by the package.

2 \begin{warpMathJax}
3 \PackageNoteNoLine{lwarp, physics}{The MathJax v3 extension will be used}
4 \CustomizeMathJax{\require{physics}}
5 \end{warpMathJax}
```

File 385   `lwarp-physunits.sty`

§ 497   Package   # physunits

(*Emulates or patches code by* BRIAN W. MULLIGAN.)

physunits (*Pkg*)   physunits is supported as-is for SVG math, and is emulated for MATHJAX.

for HTML output:
```
1 \LWR@ProvidesPackagePass{physunits}[2020/03/26]
```

2 \begin{warpMathJax}
3 \LWR@infoprocessingmathjax{physunits}
4
5 \CustomizeMathJax{\newcommand{\micro}{\mu}}
6 \CustomizeMathJax{\newcommand{\V}[1][ ]{\,\mathrm{#1V}}}
7 \CustomizeMathJax{\newcommand{\Volt}[1][ ]{\,\mathrm{#1V}}}
8 \CustomizeMathJax{\newcommand{\Coulomb}[1][ ]{\,\mathrm{#1C}}}
9 \CustomizeMathJax{\newcommand{\esu}{\,\mathrm{esu}}}
10 \CustomizeMathJax{\newcommand{\Ohm}[1][ ]{\,\mathrm{#1\Omega}}}
11 \CustomizeMathJax{\newcommand{\Amp}[1][ ]{\,\mathrm{#1A}}}
12 \CustomizeMathJax{\newcommand{\Farad}[1][ ]{\,\mathrm{#1F}}}
13 \CustomizeMathJax{\newcommand{\Tesla}[1][ ]{\,\mathrm{#1T}}}
14 \CustomizeMathJax{\newcommand{\Gauss}[1][ ]{\,\mathrm{#1G}}}
15 \CustomizeMathJax{\newcommand{\Henry}[1][ ]{\,\mathrm{#1H}}}
16 \CustomizeMathJax{\newcommand{\eV}[1][ ]{\,\mathrm{#1eV}}}
17 \CustomizeMathJax{\newcommand{\keV}{\,\mathrm{keV}}}
18 \CustomizeMathJax{\newcommand{\MeV}{\,\mathrm{MeV}}}
19 \CustomizeMathJax{\newcommand{\J}[1][ ]{\,\mathrm{#1J}}}
20 \CustomizeMathJax{\newcommand{\Joule}[1][ ]{\,\mathrm{#1J}}}
21 \CustomizeMathJax{\newcommand{\erg}{\,\mathrm{erg}}}
22 \CustomizeMathJax{\newcommand{\kcal}{\,\mathrm{kcal}}}
23 \CustomizeMathJax{\newcommand{\Cal}{\,\mathrm{Cal}}}
24 \CustomizeMathJax{\newcommand{\calorie}[1][ ]{\,\mathrm{#1cal}}}
25 \CustomizeMathJax{\newcommand{\BTU}{\,\mathrm{BTU}}}
26 \CustomizeMathJax{\newcommand{\tnt}{\,\mathrm{ton\, of\, TNT}}}
27 \CustomizeMathJax{\newcommand{\Watt}[1][ ]{\,\mathrm{#1W}}}
28 \CustomizeMathJax{\newcommand{\hpi}{\,\mathrm{hp(I)}}}
29 \CustomizeMathJax{\newcommand{\hpm}{\,\mathrm{hp(M)}}}
30 \CustomizeMathJax{\newcommand{\hp}{\,\mathrm{hp}}}
31 \CustomizeMathJax{\newcommand{\meter}[1][ ]{\,\mathrm{#1m}}}
32 \CustomizeMathJax{\newcommand{\m}[1][ ]{\,\mathrm{#1m}}}
33 \CustomizeMathJax{\newcommand{\km}{\,\mathrm{km}}}
34 \CustomizeMathJax{\newcommand{\au}{\,\mathrm{au}}}
35 \CustomizeMathJax{\newcommand{\pc}[1][ ]{\,\mathrm{#1pc}}}
36 \CustomizeMathJax{\newcommand{\ly}[1][ ]{\,\mathrm{#1ly}}}
37 \CustomizeMathJax{\newcommand{\cm}{\,\mathrm{cm}}}
38 \CustomizeMathJax{\newcommand{\nm}{\,\mathrm{nm}}}
39 \CustomizeMathJax{\newcommand{\ft}{\,\mathrm{ft}}}
40 \CustomizeMathJax{\newcommand{\inch}{\,\mathrm{in}}}
41 \CustomizeMathJax{\newcommand{\mi}{\,\mathrm{mi}}}
42 \CustomizeMathJax{\newcommand{\s}[1][ ]{\,\mathrm{#1s}}}
43 \CustomizeMathJax{\newcommand{\Sec}[1][ ]{\,\mathrm{#1s}}}
44 \CustomizeMathJax{\newcommand{\Min}{\,\mathrm{min}}}
45 \CustomizeMathJax{\newcommand{\h}{\,\mathrm{h}}}
46 \CustomizeMathJax{\newcommand{\y}[1][ ]{\,\mathrm{#1y}}}
47 \CustomizeMathJax{\newcommand{\Day}{\,\mathrm{d}}}
48
49 \CustomizeMathJax{\newcommand{\gm}[1][ ]{\,\mathrm{#1g}}}
50 \CustomizeMathJax{\newcommand{\kg}{\,\mathrm{kg}}}
51 \CustomizeMathJax{\newcommand{\lb}{\,\mathrm{lb}}}
52 \CustomizeMathJax{\newcommand{\amu}{\,\mathrm{amu}}}
53 \CustomizeMathJax{\newcommand{\N}[1][ ]{\,\mathrm{#1N}}}
54 \CustomizeMathJax{\newcommand{\Newton}[1][ ]{\,\mathrm{#1N}}}
55 \CustomizeMathJax{\newcommand{\dyne}[1][ ]{\,\mathrm{#1dyn}}}
56 \CustomizeMathJax{\newcommand{\lbf}{\,\mathrm{lbf}}}
57 \CustomizeMathJax{\newcommand{\kmps}{\,\mathrm{km}\,\mathrm{s}^{-1}}}
58 \CustomizeMathJax{\newcommand{\kmph}{\,\mathrm{km}\,\mathrm{h}^{-1}}}
59 \CustomizeMathJax{\newcommand{\mps}[1][ ]{\,\mathrm{#1m}\,\mathrm{s}^{-1}}}
60 \CustomizeMathJax{\newcommand{\miph}{\,\mathrm{mi}\,\mathrm{h}^{-1}}}
61 \CustomizeMathJax{\newcommand{\kts}{\,\mathrm{kts}}}

```
62
63 \CustomizeMathJax{\newcommand{\mpss}[1][ ]{\,\mathrm{#1m}\,\mathrm{s}^{-2}}}
64 \CustomizeMathJax{\newcommand{\gacc}{\,\mathrm{g}}}
65 \CustomizeMathJax{\newcommand{\ftpss}{\,\mathrm{ft}\,\mathrm{s}^{-2}}}
66 \CustomizeMathJax{\newcommand{\K}[1][ ]{\,\mathrm{#1K}}}
67 \CustomizeMathJax{\newcommand{\Kelvin}[1][ ]{\,\mathrm{#1K}}}
68 \CustomizeMathJax{\newcommand{\Celcius}{\,^\circ{\mathrm{C}}}}
69 \CustomizeMathJax{\newcommand{\Rankine}{\,^\circ{\mathrm{R}}}}
70 \CustomizeMathJax{\newcommand{\Fahrenheit}{\,^\circ{\mathrm{F}}}}
71
72 \CustomizeMathJax{\newcommand{\rpm}{\,\mathrm{rev}\,\Min^{-1}}}
73
74 \CustomizeMathJax{\newcommand{\Hz}[1][ ]{\,\mathrm{#1Hz}}}
75 \CustomizeMathJax{\newcommand{\barP}[1][ ]{\,\mathrm{#1bar}}}
76 \CustomizeMathJax{\newcommand{\atm}{\,\mathrm{atm}}}
77 \CustomizeMathJax{\newcommand{\Pa}[1][ ]{\,\mathrm{#1Pa}}}
78 \CustomizeMathJax{\newcommand{\mmHg}{\,\mathrm{mmHg}}}
79 \CustomizeMathJax{\newcommand{\inHg}{\,\mathrm{inHg}}}
80 \CustomizeMathJax{\newcommand{\lbsi}{\,\mathrm{psi}}}
81 \CustomizeMathJax{\newcommand{\lbsf}{\,\mathrm{psf}}}
82 \CustomizeMathJax{\newcommand{\Ba}[1][ ]{\,\mathrm{#1Ba}}}
83 \CustomizeMathJax{\newcommand{\Torr}[1][ ]{\,\mathrm{#1Torr}}}
84 \CustomizeMathJax{\newcommand{\mol}{\,\mathrm{mol}}}
85 \end{warpMathJax}
```

File 386   **`lwarp-picinpar.sty`**

§ 498   Package   # picinpar

(*Emulates or patches code by* FRIEDHELM SOWA.)

picinpar (*Pkg*)   picinpar is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{picinpar}% No date is assigned.

The window is floated by a BlockClass style.

```
2 \long\def\LWR@HTML@window[#1,#2,#3,#4] {%
3     \if #2r%
4         \begin{BlockClass}[float:right](note){marginblock}%
5     \else%
6         \begin{BlockClass}[float:left](note){marginblock}%
7     \fi%
8     #3\par%
9     #4%
10    \end{BlockClass}%
11 }
12
13 \def\endLWR@HTML@window{}
14
15 \LWR@formattedenv{window}
```

The framepic and wframepic are placed inside a BlockClass of class framebox.

```
16 \def\LWR@HTML@framepic#1{%
17     \begin{BlockClass}{framebox}
18     \expandafter\box\csname #1box\endcsname%
```

```
19      \end{BlockClass}
20 }
21 \LWR@formatted{framepic}


22 \def\LWR@HTML@wframepic#1{%
23      \begin{BlockClass}{framebox}
24      \expandafter\box\csname #1box\endcsname%
25      \end{BlockClass}
26 }
27 \LWR@formatted{wframepic}
```

The caption is placed inside a BlockClass of class figurecaption.

```
28 \long\def\LWR@HTML@@makewincaption#1#2{%
29 \begin{BlockClass}{figurecaption}
30 #1: #2
31 \end{BlockClass}
32 }
33 \LWR@formatted{@makewincaption}
```

With HTML output, figwindow and tabwindow must not pre-decrement their counters.

```
34 \long\def\LWR@HTML@figwindow[#1,#2,#3,#4] {%
35 %       \advance\c@figure -1
36      \window[#1,#2,{#3},{\def\@captype{figure}%
37          \wincaption#4\par}] }
38
39 \def\endLWR@HTML@figwindow{\endwindow}
40
41 \LWR@formattedenv{figwindow}
```

For tabwindow, to change the catcode of &, \StartDefiningTabulars is used before absorbing the arguments, and \EndDefiningTabulars is used at the end of the environment.

```
42 \long\def\LWR@HTML@subtabwindow[#1,#2,#3,#4] {%
43 %       \advance\c@table -1
44      \window[#1,#2,{#3},{\def\@captype{table}%
45          \wincaption#4\par}] }
46
47 \newcommand*{\LWR@HTML@tabwindow}{%
48      \StartDefiningTabulars%
49      \LWR@HTML@subtabwindow%
50 }
51
52 \def\endLWR@HTML@tabwindow{%
53      \endwindow%
54      \StopDefiningTabulars%
55 }
56
57 \LWR@formattedenv{tabwindow}
```

File 387 **`lwarp-pifont.sty`**

§ 499   Package **pifont**

(*Emulates or patches code by* WALTER SCHMIDT.)

pifont (*Pkg*)   pifont is patched for use by lwarp.

Hashed inline images are used, as there may not be Unicode support for all icons.

for HTML output:
```
1 \LWR@ProvidesPackagePass{pifont}[2005/04/12]
```

```
2 \renewcommand{\Pisymbol}[2]{%
3     \begin{lateximage}*[Pisymbol][pisymbol#1#2]%
4     {\Pifont{#1}\char#2}%
5     \end{lateximage}%
6 }
7
8 \newcommand{\LWR@HTML@Pifill}[2]{
9     \Pisymbol{#1}{#2} \Pisymbol{#1}{#2} \Pisymbol{#1}{#2}
10 }
11 \LWR@formatted{Pifill}
12
13 \newcommand{\LWR@HTML@Piline}[2]{%
14     \par\noindent\hspace*{0.5in}
15     \Pifill{#1}{#2} \Pifill{#1}{#2} \Pifill{#1}{#2}
16 }
17 \LWR@formatted{Piline}
```

File 388   **`lwarp-pinlabel.sty`**

§ 500   Package **pinlabel**

(*Emulates or patches code by* COLIN ROURKE.)

pinlabel (*Pkg*)   pinlabel is patched for use by lwarp.

for HTML output:
```
1 \LWR@ProvidesPackagePass{pinlabel}% no date given
```

```
2 \xpretocmd\psfig
3     {\begin{lateximage}[-pinlabel-~\PackageDiagramAltText]?}%
4     {}
5     {\LWR@patcherror{pinlabel}{psfigA}}
6
7 \xapptocmd\psfig
8     {\end{lateximage}}
9     {}
10    {\LWR@patcherror{pinlabel}{psfigB}}
```

File 389 **`lwarp-placeins.sty`**

§ 501   Package   # placeins

(*Emulates or patches code by* DONALD ARSENEAU.)

placeins (*Pkg*)   placeins is ignored.

Discard all options for lwarp-placeins:

for HTML output: 1 \LWR@ProvidesPackageDrop{placeins}[2005/04/18]

2 \newcommand*{\FloatBarrier}{}

File 390   **`lwarp-plarydshln.sty`**

§ 502   Package   # plarydshln

plarydshln (*Pkg*)   plarydshln is emulated by lwarp-arydshln.

for HTML output: 1 \LWR@ProvidesPackageDrop{plarydshln}[2018/10/20]

2 \LWR@origRequirePackage{lwarp-arydshln}

File 391   **`lwarp-plext.sty`**

§ 503   Package   # plext

plext (*Pkg*)   plext is preloaded by jtarticle and related classes.

for HTML output: 1 \LWR@loadbefore{plext}
2
3 \LWR@ProvidesPackagePass{plext}[2017/07/21]

4 \let\tate\relax
5
6 \DeclareExpandableDocumentCommand{\rensuji}{s o m}{#3}
7
8 % \layoutfloat(width,height)[pos]#4
9 \DeclareDocumentCommand{\layoutfloat}{d() o m}{}
10
11 % \DeclareLayoutCaption{type} <dir>(width)[pos1pos2]
12 \DeclareDocumentCommand{\DeclareLayoutCaption}{m d<> d() o}{}
13
14 \LetLtxMacro\pcaption\caption
15
16 % \layoutcaption<dir>(width)[pos]
17 \DeclareDocumentCommand{\layoutcaption}{d<> d() o}{}
18
19 \let\captiondir\relax

Add the optional <t/y> direction:

```
20 \RenewDocumentEnvironment{LWR@HTML@minipage}{d<> O{t} O{} O{t} m}
21     {\LWR@HTML@sub@minipage{#2}{#3}{#4}{#5}}
22     {\endLWR@HTML@sub@minipage}
23
24 \RenewDocumentCommand{\LWR@HTML@parbox}{d<> O{t} O{} O{t} m +m}
25 {
26 \LWR@traceinfo{parbox of width #4}%
27 \begin{minipage}[#2][#3][#4]{#5}%
28 #6
29 \end{minipage}%
30 }
31
32 % \pbox <t/y> [width] [l/r] {contents}
33 \RenewDocumentCommand{\pbox}{d<> O{0pt} O{c} m}{%
34 \global\booltrue{LWR@minipagefullwidth}%
35 \parbox{#2}{#4}%
36 }
```

picture, as modified by pext, is encapsulated by the lwarp core.

File 392   **`lwarp-plextarydshln.sty`**

§ 504   Package   # plextarydshln

plextarydshln (*Pkg*)   plextarydshln is emulated by lwarp-arydshln.

for HTML output: 1 \LWR@ProvidesPackageDrop{plextarydshln}[2018/10/20]

2 \LWR@origRequirePackage{lwarp-arydshln}

File 393   **`lwarp-plextcolortbl.sty`**

§ 505   Package   # plextcolortbl

plextcolortbl (*Pkg*)   plextcolortbl is emulated by lwarp-colortbl.

for HTML output: 1 \LWR@ProvidesPackageDrop{plextcolortbl}[2018/09/19]

2 \LWR@origRequirePackage{lwarp-colortbl}

File 394   **`lwarp-plimsoll.sty`**

§ 506   Package   # plimsoll

(*Emulates or patches code by* PALLE JØRGENSEN.)

plimsoll (*Pkg*)   plimsoll is used as-is for SVG math, and emulated for MATHJAX.

The circ option is honored. For MATHJAX, \plimsollsans is the same as \plimsollroman.

for HTML output: 1 \LWR@ProvidesPackagePass{plimsoll}[2020/10/09]

```
2 \begin{warpMathJax}
3 \CustomizeMathJax{\newcommand{\plimsollroman}{\unicode{x029B5}}}
4
5 \CustomizeMathJax{\let\plimsoll\plimsollroman}
6 \CustomizeMathJax{\let\plimsollsans\plimsoll}
7
8 \ifdefstring{\stst}{^{\circ}}
9     {\CustomizeMathJax{\newcommand{\stst}{^{\circ}}}}
10    {\CustomizeMathJax{\newcommand{\stst}{^{\plimsoll}}}}
11 \end{warpMathJax}
```

File 395   **`lwarp-prelim2e.sty`**

§ 507   Package   # prelim2e

(*Emulates or patches code by* MARTIN SCHRÖDER.)

prelim2e (*Pkg*)   prelim2e is ignored.

for HTML output: Discard all options for lwarp-prelim2e:

```
1 \LWR@ProvidesPackageDrop{prelim2e}[2009/05/29]
```

```
2 \newcommand{\PrelimText}{}
3 \newcommand{\PrelimTextStyle}{}
4 \newcommand{\PrelimWords}{}
```

File 396   **`lwarp-prettyref.sty`**

§ 508   Package   # prettyref

(*Emulates or patches code by* KEVIN S. RULAND.)

prettyref (*Pkg*)   prettyref is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{prettyref}[1998/07/09]

```
2 \newrefformat{fig}{Figure \ref{#1}}
3 \newrefformat{tab}{Table \ref{#1}}
```

File 397   **`lwarp-preview.sty`**

§ 509   Package   # preview

preview (*Pkg*)   preview is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{preview}[2017/04/24]

```
2 \newenvironment{preview}{}{}
3 \newenvironment{nopreview}{}{}
4 \NewDocumentCommand{\PreviewMacro}{s o o +m}{}
5 \NewDocumentCommand{\PreviewEnvironment}{s o o +m}{}
6 \newcommand{\PreviewSnarfEnvironment}[2][]{}
7 \NewDocumentCommand{\PreviewOpen}{s o}{}
8 \NewDocumentCommand{\PreviewClose}{s o}{}
9 \let\ifPreview\iffalse% \fi for syntax highlighting
```

File 398   **`lwarp-psfrag.sty`**

§ 510   Package   # psfrag

(*Emulates or patches code by* Michael C. Grant, David Carlisle.)

psfrag (*Pkg*)   **psfrag** is patched for use by lwarp.

⚠   use psfrags   The psfrags environment is modified to use lateximage to encapsulate the image. Always use a psfrags environment to contain any local \psfrag macros and the associated \includegraphics or \epsfig calls. Outside of a psfrags environment, **psfrags** adjustments will not be seen by lwarp.

⚠   Tip: Use a mono-spaced font for the tags in the EPS file.

for HTML output: 1 `\LWR@ProvidesPackagePass{psfrag}[1998/04/11]`

A lateximage captures the modified image from the document.

```
2 \BeforeBeginEnvironment{psfrags}{%
3     \begin{lateximage}[-psfrags-~\PackageDiagramAltText]?%
4 }
5
6 \AfterEndEnvironment{psfrags}{\end{lateximage}}
```

File 399   **`lwarp-psfragx.sty`**

§ 511   Package   # psfragx

(*Emulates or patches code by* Pascal Kockaert.)

psfragx (*Pkg*)   **psfragx** is patched for use by lwarp.

for HTML output: 1 `\LWR@ProvidesPackagePass{psfragx}[2012/05/02]`

A lateximage captures the modified image from the document.

```
2 \VerifyCommand[lwarp][psfragx]{\pfx@includegraphicx}{45FCF58D66C0BFFC685913A78CADF20D}
3
4 \def\pfx@includegraphicx#1#2{%
5     \begin{lateximage}[-psfragx-~\PackageDiagramAltText]?%
6     \mbox{\pfx@overpix{#1}{#2}\endpfx@overpix}%
7     \end{lateximage}%
8 }
9
```

```
10 \VerifyCommand[lwarp][psfragx]{\@@@overpix}{DD69D71E9C551D4D568AE4269AAAC1C0}
11
12 \def\@@@overpix[#1]<#2>[#3]#4{%
13     \begin{lateximage}[-psfragx-~\PackageDiagramAltText]?%
14     \pfx@overpix{#1,ovpfgd={#2},ovpbgd={#3}}{#4}%
15 }
16
17 \VerifyCommand[lwarp][psfragx]{\endoverpix}{722C858D87F96798ABE0BAF89CB13373}
18
19 \def\endoverpix{%
20     \endpfx@overpix%
21     \end{lateximage}%
22 }
```

File 400   **lwarp-pst-eps.sty**

§ 512   Package   # pst-eps

(*Emulates or patches code by* Herbert Voss.)

pst-eps (*Pkg*)   pst-eps is patched for use by lwarp.

for HTML output: `1 \LWR@ProvidesPackagePass{pst-eps}[2005/05/20]`

```
2 \renewenvironment{TeXtoEPS}{}{}
3 \renewcommand{\PSTtoEPS}[3][]{}
```

File 401   **lwarp-pstool.sty**

§ 513   Package   # pstool

(*Emulates or patches code by* Zebb Prime, Will Robertson.)

pstool (*Pkg*)   pstool is patched for use by lwarp.

\graphicspath is ignored, and the file directory must be stated.

⚠ path and filename   The filename must not have a file extension.

Use

   Enter ⇒   **lwarpmk html**

followed by

   Enter ⇒   **lwarpmk limages**

.

for HTML output: `1 \LWR@ProvidesPackagePass{pstool}[2018/01/20]`

Each image is placed inside a lateximage to capture the results of psfrag.

`2 \renewcommand\pstool@alwaysprocess[3][]{%`

```
3      \begin{lateximage}[-pstool-~\PackageDiagramAltText]?%
4      \includegraphics{#2.pdf}%
5      \end{lateximage}%
6 }
7 \LetLtxMacro\pstool@neverprocess\pstool@alwaysprocess
8 \LetLtxMacro\pstool@maybeprocess\pstool@alwaysprocess
9
10 \renewcommand\pstool@@psfragfig[4]{%
11      \begin{lateximage}[-pstool-~\PackageDiagramAltText]?%
12      \includegraphics{#2.pdf}%
13      \end{lateximage}%
14 }
```

File 402   **`lwarp-pstricks.sty`**

§ 514   Package   # pstricks

(*Emulates or patches code by* TIMOTHY VAN ZANDT.)

pstricks (*Pkg*)   pstricks is patched for use by lwarp.

⚠ use pspicture   All pstricks content should be contained inside a pspicture environment.

for HTML output: `1 \LWR@ProvidesPackagePass{pstricks}[2018/01/06]`

```
2 \BeforeBeginEnvironment{pspicture}{%
3      \begin{lateximage}[pspicture]%
4 }
5 \AfterEndEnvironment{pspicture}{\end{lateximage}}
6
7 \BeforeBeginEnvironment{pspicture*}{%
8      \begin{lateximage}[pspicture]%
9 }
10 \AfterEndEnvironment{pspicture*}{\end{lateximage}}
```

File 403   **`lwarp-pxatbegshi.sty`**

§ 515   Package   # pxatbegshi

pxatbegshi (*Pkg*)   pxatbegshi is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{pxatbegshi}[2017/11/04]`

`2 \LWR@origRequirePackage{lwarp-atbegshi}`

File 404   **`lwarp-pxeveryshi.sty`**

§ 516   Package   # pxeveryshi

pxeveryshi (*Pkg*)   pxeveryshi is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{pxeveryshi}[2012/05/19]`

2 \LWR@origRequirePackage{lwarp-everyshi}

File 405 `lwarp-pxfonts.sty`

§ 517　Package **pxfonts**

(*Emulates or patches code by* YOUNG RYU.)

pxfonts (*Pkg*)　pxfonts is used as-is for SVG math, and is emulated for MATHJAX.

for HTML output:　1 \LWR@ProvidesPackagePass{pxfonts}[2008/01/22]

For MATHJAX:

```
2 \LWR@origRequirePackage{lwarp-common-mathjax-letters}
3
4 \begin{warpMathJax}
5 \LWR@infoprocessingmathjax{pxfonts}
6
7 \LWR@mathjax@addgreek@l@up{}{up}
8 \end{warpMathJax}
```

File 406 `lwarp-pxftnright.sty`

§ 518　Package **pxftnright**

pxftnright (*Pkg*)　pxftnright is ignored.

for HTML output:　1 \LWR@ProvidesPackageDrop{pxftnright}[2017/02/28]

2 \LWR@origRequirePackage{lwarp-ftnright}

File 407 `lwarp-pxjahyper.sty`

§ 519　Package **pxjahyper**

pxjahyper (*Pkg*)　pxjahyper is ignored.

for HTML output:　1 \LWR@ProvidesPackageDrop{pxjahyper}[2018/07/15]

File 408 `lwarp-quotchap.sty`

§ 520　Package **quotchap**

(*Emulates or patches code by* KARSTEN TINNEFELD, JAN KLEVER.)

quotchap (*Pkg*)　quotchap is emulated.

for HTML output:　1 \LWR@ProvidesPackageDrop{quotchap}[2019/07/09]

```
2 \newcommand{\@quotchap}{}
3 \newlength{\LWR@quotchapwidth}
4
5 \let\@printcites\relax
6
7 \newcommand*{\@iprintcites}{%
```

Place the quotes inside a <div> of class quotchap, of the maximum selected width:

```
8 \begin{BlockClass}[max-width: \LWR@printlength{\LWR@quotchapwidth}]{quotchap}
9 %\begin{minipage}{\LWR@quotchapwidth}
10 \@quotchap
11 %\end{minipage}
12 \end{BlockClass}
```

Deactivate the quote printing:

```
13 \global\let\@printcites\relax
14 }
15
16 \NewEnviron{savequote}[1][\linewidth]{%
```

Remember the width, adjusted for HTML, and make the length assignment global, per:
https://tex.stackexchange.com/questions/300823/
        why-is-setlength-ineffective-inside-a-tabular-environment

```
17 \setlength{\LWR@quotchapwidth}{#1*2}%
18 \global\LWR@quotchapwidth=\LWR@quotchapwidth%
```

Remember the body, and activate the quote printing:

```
19 \global\let\@quotchap\BODY
20 \global\let\@printcites\@iprintcites%
21 }
```

The quotation author is placed inside a <div> of class qauthor:

```
22 \newcommand{\qauthor}[1]{%
23     \LWR@stoppars%
24     \begin{BlockClass}{qauthor}%
25     {#1}%
26     \end{BlockClass}%
27     \LWR@startpars%
28 }
```

Fonts are ignored. Use CSS.

```
29 \newcommand{\qsetcnfont}[1]{}
30 \providecommand*{\quotefont}{}
31 \providecommand*{\qauthorfont}{}
```

File 409   **`lwarp-quoting.sty`**

Package   # quoting

(*Emulates or patches code by* THOMAS TITZ.)

quoting (*Pkg*)   quoting is patched for use by lwarp.

for HTML output: 1 `\LWR@ProvidesPackagePass{quoting}[2014/01/28]`

2 `\VerifyEnvironment[lwarp][quoting]{quoting}`
3 `  {AEC586766C9109C2889BDED4AE083C05}{8CE7FB71438699772DFD79A2BC803AB3}`
4
5 `\xpatchcmd{\quoting}{\quo@begintext}`
6 `  {\begin{LWR@blocktextcurrentfont}\quo@begintext}`
7 `  {}`
8 `  {\LWR@patcherror{quoting}{quoting}}`
9
10 `\xpatchcmd{\endquoting}{\quo@endtext}`
11 `  {\quo@endtext\end{LWR@blocktextcurrentfont}\LWR@stoppars}`
12 `  {}`
13 `  {\LWR@patcherror{quoting}{endquoting}}`

File 410   **`lwarp-ragged2e.sty`**

Package   # ragged2e

(*Emulates or patches code by* MARTIN SCHRÖDER.)

ragged2e (*Pkg*)   ragged2e is emulated.

Discard all options for lwarp-ragged2e:

for HTML output: 1 `\LWR@ProvidesPackageDrop{ragged2e}[2009/05/21]`

2 `\LetLtxMacro\Centering\centering`
3 `\LetLtxMacro\RaggedLeft\raggedleft`
4 `\LetLtxMacro\RaggedRight\raggedright`
5 `\newcommand*{\justifying}{}`
6 `\newlength{\CenteringLeftskip}`
7 `\newlength{\RaggedLeftLeftskip}`
8 `\newlength{\RaggedRightLeftskip}`
9 `\newlength{\CenteringRightskip}`
10 `\newlength{\RaggedLeftRightskip}`
11 `\newlength{\RaggedRightRightskip}`
12 `\newlength{\CenteringParfillskip}`
13 `\newlength{\RaggedLeftParfillskip}`
14 `\newlength{\RaggedRightParfillskip}`
15 `\newlength{\JustifyingParfillskip}`
16 `\newlength{\CenteringParindent}`
17 `\newlength{\RaggedLeftParindent}`
18 `\newlength{\RaggedRightParindent}`
19 `\newlength{\JustifyingParindent}`

```
20 \newenvironment*{Center}{\center}{\endcenter}
21 \newenvironment*{FlushLeft}{\flushleft}{\endflushleft}
22 \newenvironment*{FlushRight}{\flushright}{\endflushright}
23 \newenvironment*{justify}{\justifying}{\endjustifying}
```

File 411   **`lwarp-realscripts.sty`**

§ 523   Package   **realscripts**

(*Emulates or patches code by* WILL ROBERTSON.)

realscripts (*Pkg*)   realscripts is emulated. See `lwarp.css` for the <span> of class supsubscript.

for HTML output: 1 \LWR@ProvidesPackagePass{realscripts}[2016/02/13]


```
2 \ExplSyntaxOn
3
4 \DeclareDocumentCommand \LWR@HTML@realsubscript {m} {
5     \LWR@HTML@textsubscript{#1}
6 }
7
8 \LWR@formatted{realsubscript}
9
10
11 \DeclareDocumentCommand \LWR@HTML@realsuperscript {m} {
12     \LWR@HTML@textsuperscript{#1}
13 }
14
15 \LWR@formatted{realsuperscript}
16
17
18 \ExplSyntaxOff
19
20
21 \newcommand*{\LWR@realscriptsalign}{}
22
23 \newcommand*{\LWR@setrealscriptsalign}[1]{%
24     \renewcommand*{\LWR@realscriptsalign}{}%
25     \ifthenelse{\equal{#1}{c}}{%
26         \renewcommand{\LWR@realscriptsalign}{%
27             \LWR@print@mbox{text-align:center} ; %
28         }%
29     }{}%
30     \ifthenelse{\equal{#1}{r}}{%
31         \renewcommand{\LWR@realscriptsalign}{%
32             \LWR@print@mbox{text-align:right} ; %
33         }%
34     }{}%
35 }
36
37 \DeclareDocumentCommand \LWR@HTML@textsubsuperscript {s O{l} mm} {%
38     \LWR@setrealscriptsalign{#2}%
39     \InlineClass[\LWR@realscriptsalign]{supsubscript}{%
40         \textsuperscript{#4}\textsubscript{#3}%
41     }%
42 }
43 \LWR@formatted{textsubsuperscript}
```

```
44
45 \FilenameNullify{%
46     \RenewDocumentCommand{\textsuperscript}{s m}{}%
47     \RenewDocumentCommand{\textsubscript}{s m}{}%
48     \renewcommand{\fakesubscript}[1]{}%
49     \renewcommand{\fakesuperscript}[1]{}%
50     \renewcommand{\realsubscript}[1]{}%
51     \renewcommand{\realsuperscript}[1]{}%
52     \renewcommand{\textsubsuperscript}[2]{}%
53     \renewcommand{\textsupersubscript}[2]{}%
54 }
```

File 412   `lwarp-refcheck.sty`

§ 524   Package   **refcheck**

refcheck (*Pkg*)   refcheck is ignored.

for HTML output: 1 `\LWR@ProvidesPackageDrop{refcheck}[2013/02/14]`

```
2 \def\showrefnames{}
3 \def\norefnames{}
4 \def\showcitenames{}
5 \def\nocitenames{}
6 \def\setonmsgs{}
7 \def\setoffmsgs{}
8 \def\checkunlbld{}
9 \def\ignoreunlbld{}
10 \newcommand*{\refcheckxrdoc}[2][]{}
```

File 413   `lwarp-register.sty`

§ 525   Package   **register**

(*Emulates or patches code by* Matthew Lovell.)

register (*Pkg*)   register is patched for use by lwarp.

for HTML output: 1 `\LWR@ProvidesPackagePass{register}[2019/01/01]`

Not using \VerifyCommand here because these patches are not likely to be affected by changes in the original.

```
2 \xpatchcmd{\register}
3     {\centering}
4     {%
5         \begin{center}%
6         \begin{lateximage}[-register-~\PackageDiagramAltText]?%
7     }
8     {}
9     {\LWR@patcherror{register}{register}}
10
11 \xpatchcmd{\endregister}
```

```
12    {\leftskip}
13    {%
14        \end{lateximage}\end{center}%
15        \leftskip%
16    }%
17    {}
18    {\LWR@patcherror{register}{endregister}}
19
20 \expandafter\xapptocmd\csname register*\endcsname
21    {%
22        \begin{center}%
23        \begin{lateximage}[-register-~\PackageDiagramAltText]?%
24    }
25    {}
26    {\LWR@patcherror{register}{register*}}
27
28 \expandafter\xpatchcmd\csname endregister*\endcsname
29    {\leftskip}
30    {%
31        \end{lateximage}\end{center}%
32        \leftskip%
33    }%
34    {}
35    {\LWR@patcherror{register}{endregister*}}
36
37 \setlength{\regWidth}{5in}
```

File 414   **`lwarp-relsize.sty`**

§ 526   Package **relsize**

(*Emulates or patches code by* DONALD ARSENEAU, BERNIE COSELL, MATT SWIFT.)

relsize (*Pkg*)   relsize is patched for use by lwarp, and emulated for MATHJAX.

For HTML, only the inline macros are supported: \textlarger, \textsmaller, and \textscale. Each becomes an inline span of a modified font-size.

\relsize, \larger, \smaller, and \relscale are ignored.

While creating SVG math for HTML, the original definitions are temporarilty restored, and so should work as expected.

⚠ not small   The HTML browser's setting for minumum font size may limit how small the output will be displayed.

for HTML output: 1 \LWR@ProvidesPackagePass{relsize}[2013/03/29]

```
2 \let\LWR@origrelsize\relsize
3 \LetLtxMacro\LWR@origlarger\larger
4 \LetLtxMacro\LWR@origsmaller\smaller
5 \let\LWR@relscale\relscale
6 \LetLtxMacro\LWR@origtextlarger\textlarger
7 \LetLtxMacro\LWR@origtextsmaller\textsmaller
8 \let\LWR@textscale\textscale
9
10 \appto\LWR@restoreorigformatting{%
```

```
11 \let\relsize\LWR@origrelsize%
12 \LetLtxMacro\larger\LWR@origlarger%
13 \LetLtxMacro\smaller\LWR@origsmaller%
14 \let\relscale\LWR@relscale%
15 \LetLtxMacro\textlarger\LWR@origtextlarger%
16 \LetLtxMacro\textsmaller\LWR@origtextsmaller%
17 \let\textscale\LWR@textscale%
18 }
19
20 \newcounter{LWR@relsizetemp}
21
22 \renewcommand*{\relsize}[1]{}
23 \renewcommand*{\larger}[1][]{}
24 \renewcommand*{\smaller}[1][]{}
25 \renewcommand*{\relscale}[1]{}
26
27 \renewcommand*{\textlarger}[2][1]{%
28 \setcounter{LWR@relsizetemp}{100+(#1*20)}%
29 \InlineClass[font-size:\arabic{LWR@relsizetemp}\%]{textlarger}{#2}%
30 }
31
32 \renewcommand*{\textsmaller}[2][1]{%
33 \setcounter{LWR@relsizetemp}{100-(#1*20)}%
34 \InlineClass[font-size:\arabic{LWR@relsizetemp}\%]{textsmaller}{#2}%
35 }
36
37 \renewcommand*{\textscale}[2]{%
38 \setcounter{LWR@relsizetemp}{100*\real{#1}}%
39 \InlineClass[font-size:\arabic{LWR@relsizetemp}\%]{textscale}{#2}%
40 }
```

For MATHJAX:

```
41 \begin{warpMathJax}
42 \CustomizeMathJax{\newcommand{\mathlarger}[1]{#1}}
43 \CustomizeMathJax{\newcommand{\mathsmaller}[1]{#1}}
44 \end{warpMathJax}
```

File 415   **`lwarp-repeatindex.sty`**

§ 527   Package   # repeatindex

repeatindex (*Pkg*)   repeatindex is emulated for lwarp.

⚠   style file   lwarp must be used with a special style file:

>   \usepackage[makeindex,makeindexStyle={lwarp_repeatindex}]{lwarp}

where lwarp_repeatindex.ist may be copied from the following modified version of lwarp.ist:

```
preamble
"\\begin{theindex}
  \\providecommand*\\lettergroupDefault[1]{}
  \\providecommand*\\lettergroup[1]{%
      \\par\\textbf{#1}\\par
```

```
        \\nopagebreak
    }
"
headings_flag 1
heading_prefix "
  \\lettergroup{"
heading_suffix "}"
delim_0 "], \\hyperindexref{"
delim_1 ", \\hyperindexref{"
delim_2 ", \\hyperindexref{"
delim_n "}, \\hyperindexref{"
delim_r "} -- \\hyperindexref{"
delim_t "}"

item_0 "\n  \\item ["
```

(The modifications are the delim_0 and item_0 entries.)

for HTML output: 1 \LWR@ProvidesPackageDrop{repeatindex}[2001/10/13]

In the lwarp core, \LWR@indexitem is modified to accept the optional \item argument.

2 \RequirePackage{makeidx}
3 \def\entryprefix{\itshape}
4 \def\entrypostfix{\dots}

File 416   **lwarp-repltext.sty**

§ 528   Package   # repltext

repltext (*Pkg*)   repltext is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{repltext}[2020/09/25]

2 \newcommand{\repltext}[2]{#2}
3 \newcommand*{\prevrepl}{}

For MATHJAX:

4 \begin{warpMathJax}
5 \CustomizeMathJax{\newcommand{\repltext}[2]{#2}}
6 \end{warpMathJax}

File 417   **lwarp-resizegather.sty**

§ 529   Package   # resizegather

resizegather (*Pkg*)   resizegather is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{resizegather}[2016/05/16]

```
2 \newcommand*{\resizegathersetup}[1]{}
```

File 418   `lwarp-returntogrid.sty`

§ 530   Package   **returntogrid**

returntogrid (*Pkg*)   returntogrid is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{returntogrid}[2018/08/21]`

```
2 \NewDocumentCommand\returntogrid{ O {} }{}
3 \NewDocumentCommand\returntogridsetup { m } {}
4 \NewDocumentCommand\showdebugpagegrid {} {}
```

File 419   `lwarp-rlepsf.sty`

§ 531   Package   **rlepsf**

(*Emulates or patches code by* MICHAEL GREENE, COLIN ROURKE.)

rlepsf (*Pkg*)   rlepsf is patched for use by lwarp.

⚠ Rename the style file!   The file rlepsf.tex must be copied to rlepsf.sty for lwarp to detect and patch it.

for HTML output: `1 \LWR@ProvidesPackagePass{rlepsf}% No date given.`

```
2 \xpretocmd{\relabelbox}
3   {\begin{lateximage}}
4   {}
5   {\LWR@patcherror{rlepsf}{relabelbox}}
6
7 \xapptocmd{\endrelabelbox}
8   {\end{lateximage}}
9   {}
10   {\LWR@patcherror{rlepsf}{endrelabelbox}}
```

File 420   `lwarp-rmathbr.sty`

§ 532   Package   **rmathbr**

(*Emulates or patches code by* DENIS RYABOV.)

rmathbr (*Pkg*)   rmathbr is used as-is for svg math, and emulated for MATHJAX.

for HTML output: `1 \LWR@ProvidesPackagePass{rmathbr}[2020/12/11]`

```
2 \begin{warpMathJax}
3 \CustomizeMathJax{\def\*{~}}
4 \CustomizeMathJax{\newcommand{\cdott}{\cdot}}
5 \CustomizeMathJax{\newcommand{\nobr}{}}
6 \end{warpMathJax}
```

File 421 `lwarp-rmpage.sty`

§ 533   Package   # rmpage

rmpage (*Pkg*)   rmpage is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{rmpage}[1997/09/29]

File 422 `lwarp-romanbar.sty`

§ 534   Package   # romanbar

(*Emulates or patches code by* H.-MARTIN MÜNCH.)

romanbar (*Pkg*)   romanbar is patched for use by lwarp.

An inline class with an overline and underline is used.

for HTML output: 1 \LWR@ProvidesPackagePass{romanbar}[2012/01/01]

```
2 \DeclareRobustCommand{\Roman@bar}[1]{% #1 is in Roman, i.e. MMXII
3 \InlineClass[%
4     text-decoration: overline underline ;
5 ]{romanbar}{#1}%
6 }
```

File 423 `lwarp-romanbarpagenumber.sty`

§ 535   Package   # romanbarpagenumber

romanbarpagenumber (*Pkg*)   romanbarpagenumber is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{romanbarpagenumber}[2015/02/06]

File 424 `lwarp-rotating.sty`

§ 536   Package   # rotating

(*Emulates or patches code by* ROBIN FAIRBAIRNS, SEBASTIAN RAHTZ, LEONOR BARROCA.)

rotating (*Pkg*)   rotating is emulated.

All rotations are ignored in HTML output.

for HTML output: 1 \LWR@ProvidesPackagePass{rotating}[2016/08/11]
2 \RequirePackage{graphicx}

```
3 \LetLtxMacro\LWR@HTML@sidewaystable\table
4 \let\endLWR@HTML@sidewaystable\endtable
5 \LWR@formattedenv{sidewaystable}
6
7 \LetLtxMacro\LWR@HTML@sidewaysfigure\figure
8 \let\endLWR@HTML@sidewaysfigure\endfigure
9 \LWR@formattedenv{sidewaysfigure}
10
11 \newenvironment*{LWR@HTML@sideways}{}{}
12 \LWR@formattedenv{sideways}
13
14 \newenvironment*{LWR@HTML@turn}[1]{}{}
15 \LWR@formattedenv{turn}
16
17 \newenvironment*{LWR@HTML@rotate}[1]{}{}
18 \LWR@formattedenv{rotate}
19
20 \NewDocumentCommand{\LWR@HTML@turnbox}{m +m}{#2}
21 \LWR@formatted{turnbox}
22
23 \let\LWR@HTML@rotcaption\caption
24 \LWR@formatted{rotcaption}
25
26 \let\LWR@HTML@@makerotcaption\@makecaption
27 \LWR@formatted{@makerotcaption}
```

File 425   **`lwarp-rotfloat.sty`**

§ 537   Package   **rotfloat**

(*Emulates or patches code by* AXEL SOMMERFELDT.)

rotfloat (*Pkg*)   rotfloat is emulated.

```
1 \LWR@ProvidesPackageDrop{rotfloat}[2004/01/04]
2
3 \RequirePackage{float}
4 \RequirePackage{rotating}
```

\newfloat   {⟨*1: type*⟩} {⟨*2: placement*⟩} {⟨*3: ext*⟩} [⟨*4: within*⟩]

Emulates the \newfloat command from the float package. Sideways floats are
\let to the same as regular floats.

"placement" is ignored.

```
5 \RenewDocumentCommand{\newfloat}{m m m o}{%
6 \IfValueTF{#4}%
7 {%
8     \DeclareFloatingEnvironment[fileext=#3,within=#4]{#1}%
9 }%
10 {%
11     \DeclareFloatingEnvironment[fileext=#3]{#1}%
12 }%
13 \csletcs{sideways#1}{#1}%
14 \csletcs{endsideways#1}{end#1}%
```

Remember the float style:

```
15 \csedef{LWR@floatstyle@#1}{\LWR@floatstyle}%
16 \csedef{LWR@floatstyle@sideways#1}{\LWR@floatstyle}%
```

newfloat package automatically creates the \listof command for new floats, but float does not, so remove \listof here in case it is manually created later:

```
17 \cslet{listof#1s}\relax%
18 \cslet{listof#1es}\relax%
19 \cslet{listofsideways#1s}\relax%
20 \cslet{listofsideways#1es}\relax%
21 }
```

File 426   **lwarp-rviewport.sty**

§ 538   Package   # rviewport

rviewport (*Pkg*)   rviewport is honored inside a lateximage, and otherwise ignored for HTML output.

If rviewport is important for an image, enclose the image inside a lateximage environment.

for HTML output: 1 \LWR@ProvidesPackagePass{rviewport}[2011/08/27]

2 \define@key{igraph}{rviewport}{}

File 427   **lwarp-savetrees.sty**

§ 539   Package   # savetrees

savetrees (*Pkg*)   savetrees is ignored.

for HTML output: Discard all options for lwarp-savetrees:

1 \LWR@ProvidesPackageDrop{savetrees}[2016/04/13]

File 428   **lwarp-scalefnt.sty**

§ 540   Package   # scalefnt

(*Emulates or patches code by* D. CARLISLE.)

scalefnt (*Pkg*)   scalefnt is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{scalefnt}

2 \DeclareRobustCommand\scalefont[1]{}

File 429 **`lwarp-scalerel.sty`**

§ 541   Package   # scalerel

(*Emulates or patches code by* STEVEN B. SEGLETES.)

scalerel (*Pkg*)   scalerel is used as-is for SVG math, and is emulated and ignored for MATHJAX.

for HTML output: 1 `\LWR@ProvidesPackagePass{scalerel}[2016/12/29]`

For MATHJAX:

```
2 \begin{warpMathJax}
3 \LWR@infoprocessingmathjax{scalerel}
4
5 \CustomizeMathJax{\newcommand{\scalerel}{\ifstar{\scalerelplain}{\scalerelplus}}}
6 \CustomizeMathJax{\newcommand{\scalerelplain}[3][]{#2}}
7 \CustomizeMathJax{\newcommand{\scalerelplus}[3][]{#2#3}}
8 \CustomizeMathJax{\newcommand{\stretchrel}{\ifstar{\stretchrelplain}{\stretchrelplus}}}
9 \CustomizeMathJax{\newcommand{\stretchrelplain}[3][]{#2}}
10 \CustomizeMathJax{\newcommand{\stretchrelplus}[3][]{#2#3}}
11 \CustomizeMathJax{\newcommand{\scaleto}[3][]{#2}}
12 \CustomizeMathJax{\newcommand{\stretchto}[3][]{#2}}
13 \CustomizeMathJax{\newcommand{\scaleleftright}[4][]{#2#3#4}}
14 \CustomizeMathJax{\newcommand{\stretchleftright}[4][]{#2#3#4}}
15 \CustomizeMathJax{\newcommand{\hstretch}[2]{#2}}
16 \CustomizeMathJax{\newcommand{\vstretch}[2]{#2}}
17 \CustomizeMathJax{\newcommand{\scaleobj}[2]{#2}}
18 \CustomizeMathJax{\newcommand{\ThisStyle}[1]{#1}}
19 \CustomizeMathJax{\newcommand{\SavedStyle}{}}
20 \CustomizeMathJax{\def\scriptstyleScaleFactor{.7}}
21 \CustomizeMathJax{\def\scriptscriptstyleScaleFactor{.5}}
22 \CustomizeMathJax{\newcommand{\discernmathstyle}{}}
23 \CustomizeMathJax{\newcommand{\ignoremathstyle}[1][T]{}}
24 \CustomizeMathJax{\newcommand{\Isnextbyte}[3][v]{}}
25 \end{warpMathJax}
```

File 430   **`lwarp-schemata.sty`**

§ 542   Package   # schemata

(*Emulates or patches code by* CHARLES P. SCHAUM.)

schemata (*Pkg*)   schemata is patched for use by lwarp.

for HTML output: 1 `\LWR@ProvidesPackagePass{schemata}[2020/11/23]`

```
2 \LetLtxMacro\LWR@schemata@origschema\schema
3 \LetLtxMacro\LWR@schemata@origSchema\Schema
4
5 \renewcommand{\schema}[3][open]{%
6     \begin{lateximage}[-schemata-~\PackageDiagramAltText]?%
7     \LWR@print@normalsize%
```

```
 8        \LWR@schemata@origschema[#1]{#2}{#3}%
 9        \end{lateximage}%
10 }
11
12 \renewcommand{\Schema}[5][open]{%
13        \begin{lateximage}[-schemata-~\PackageDiagramAltText]?%
14        \LWR@print@normalsize%
15        \LWR@schemata@origSchema[#1]{#2}{#3}{#4}{#5}%
16        \end{lateximage}%
17 }
```

File 431   **`lwarp-scrextend.sty`**

§ 543   Package   **scrextend**

scrextend (*Pkg*)   scrextend is emulated.

This package may be loaded standalone, but is also loaded automatically if koma-script classes are in use. \DeclareDocumentCommand is used to overwrite the koma-script definitions.

for HTML output: 1 \LWR@ProvidesPackageDrop{scrextend}[2020/01/24]

```
 2 \DeclareDocumentCommand{\setkomafont}{m m}{}
 3 \DeclareDocumentCommand{\addkomafont}{m m}{}
 4 \DeclareDocumentCommand{\usekomafont}{m}{}
 5
 6 \DeclareDocumentCommand{\usefontofkomafont}{m}{}
 7 \DeclareDocumentCommand{\useencodingofkomafont}{m}{}
 8 \DeclareDocumentCommand{\usesizeofkomafont}{m}{}
 9 \DeclareDocumentCommand{\usefamilyofkomafont}{m}{}
10 \DeclareDocumentCommand{\useseriesofkomafont}{m}{}
11 \DeclareDocumentCommand{\useshapeofkomafont}{m}{}
12
13 \providecommand*{\coverpagetopmargin}{}
14 \providecommand*{\coverpagebottommargin}{}
15 \providecommand*{\coverpageleftmargin}{}
16 \providecommand*{\coverpagerightmargin}{}
17
```

Title page:

```
18 \AtBeginDocument{
19        \let\LWR@koma@orig@maketitle\maketitle
20        \DeclareDocumentCommand{\maketitle}{o}{\LWR@koma@orig@maketitle}
21 }
22
23 \providecommand*{\@maketitle}{}
24 \renewrobustcmd{\@maketitle}{%
25        \ifdefvoid{\@titlehead}{}{%
26            \begin{BlockClass}{titlehead}%
27            \@titlehead%
28            \end{BlockClass}%
29        }%
30        \ifdefvoid{\@subject}{}{%
31            \begin{BlockClass}{subject}%
```

```
32        \@subject%
33        \end{BlockClass}%
34     }%
35     \LWR@stoppars%
36     \LWR@htmltag{\LWR@tagtitle}%
37     \@title%
38     \LWR@htmltag{\LWR@tagtitleend}%
39     \ifdefvoid{\@subtitle}{}{%
40        \begin{BlockClass}{subtitle}%
41        \@subtitle%
42        \end{BlockClass}%
43     }%
44     \LWR@startpars%
45     \begin{BlockClass}{author}%

46     \renewcommand*{\cr}{}%
47     \renewcommand*{\crcr}{}%
48     \renewcommand*{\noalign}{}%

49        \renewcommand{\and}{%
50           \end{BlockClass}%
51           \begin{BlockClass}{oneauthor}%
52        }%
53        \begin{BlockClass}{oneauthor}%
54           \@author%
55        \end{BlockClass}%
56     \end{BlockClass}%
57     \begin{BlockClass}{titledate}%
58     \@date%
59     \end{BlockClass}%
60     \ifdefvoid{\@published}{}{%
61        \begin{BlockClass}{published}%
62        \@published%
63        \end{BlockClass}%
64     }%
65 }
66
67 \AddSubtitlePublished
68
69 \DeclareDocumentCommand{\extratitle}{m}{}
70 \DeclareDocumentCommand{\frontispiece}{m}{}
71
72 \def\@titlehead{}%
73 \DeclareDocumentCommand{\titlehead}{m}{\gdef\@titlehead{#1}}%
74
75 \def\@subject{}%
76 \DeclareDocumentCommand{\subject}{m}{\gdef\@subject{#1}}%
77
78 % \subtitle and \published are defined by \AddSubtitlePublished
79
80 \DeclareDocumentCommand{\publishers}{m}{\published{#1}}
81
82 \DeclareDocumentCommand{\uppertitleback}{m}{}
83 \DeclareDocumentCommand{\lowertitleback}{m}{}
84 \DeclareDocumentCommand{\dedication}{m}{}
85
86 \DeclareDocumentCommand{\ifthispageodd}{m m}{#1}
87
88 \DeclareDocumentCommand{\cleardoublepageusingstyle}{m}{}
89 \DeclareDocumentCommand{\cleardoubleemptypage}{}{}
```

```
 90 \DeclareDocumentCommand{\cleardoubleplainpage}{}{}
 91 \DeclareDocumentCommand{\cleardoublestandardpage}{}{}
 92 \DeclareDocumentCommand{\cleardoubleoddpage}{}{}
 93 \DeclareDocumentCommand{\cleardoubleoddpageusingstyle}{m}{}
 94 \DeclareDocumentCommand{\cleardoubleoddemptypage}{}{}
 95 \DeclareDocumentCommand{\cleardoubleoddplainpage}{}{}
 96 \DeclareDocumentCommand{\cleardoubleoddstandardpage}{}{}
 97 \DeclareDocumentCommand{\cleardoubleevenpage}{}{}
 98 \DeclareDocumentCommand{\cleardoubleevenpageusingstyle}{m}{}
 99 \DeclareDocumentCommand{\cleardoubleevenemptypage}{}{}
100 \DeclareDocumentCommand{\cleardoubleevenplainpage}{}{}
101 \DeclareDocumentCommand{\cleardoubleevenstandardpage}{}{}
102
103 \DeclareDocumentCommand{\multiplefootnoteseparator}{}{%
104   \begingroup\let\thefootnotemark\multfootsep\@makefnmark\endgroup
105 }
106
107 \DeclareDocumentCommand{\multfootsep}{}{,}
108
109 \DeclareDocumentCommand{\footref}{m}{%
110   \begingroup
111     \unrestored@protected@xdef\@thefnmark{\ref{#1}}%
112   \endgroup
113   \@footnotemark
114 }
115
116 \DeclareDocumentCommand{\deffootnote}{o m m m}{}
117 \DeclareDocumentCommand{\deffootnotemark}{m}{}
118 \DeclareDocumentCommand{\setfootnoterule}{o m}{}
119 \DeclareDocumentCommand{\raggedfootnote}{}{}


120 \DeclareDocumentCommand{\dictum}{o m}{
121 \begin{LWR@BlockClassWP}{\LWR@print@mbox{text-align:right}}{}{dictum}
122     #2
123     \IfValueT{#1}
124     {
125         \LWR@stoppars%
126         \ifbool{FormatWP}
127     {\begin{BlockClass}[\LWR@print@mbox{border-top:} 1px solid gray]{dictumauthor}}
128         {\begin{BlockClass}{dictumauthor}}
129         \dictumauthorformat{#1}
130         \end{BlockClass}
131     }
132 \end{LWR@BlockClassWP}
133 }
134
135 \DeclareDocumentCommand{\dictumwidth}{}{}
136 \DeclareDocumentCommand{\dictumauthorformat}{m}{(#1)}
137 \DeclareDocumentCommand{\dictumrule}{}{}
138 \DeclareDocumentCommand{\raggeddictum}{}{}
139 \DeclareDocumentCommand{\raggeddictumtext}{}{}
140 \DeclareDocumentCommand{\raggeddictumauthor}{}{}
141
142 \DeclareDocumentEnvironment{labeling}{o m}
143 {%
144 \def\sc@septext{#1}%
145 \list{}{}%
146 \let\makelabel\labelinglabel%
147 }
148 {
```

```
149 \endlist
150 }
151
152 \DeclareDocumentCommand{\labelinglabel}{m}{%
153 #1 \qquad \sc@septext%
154 }
155
156 \let\addmargin\relax
157 \let\endaddmargin\relax
158 \cslet{addmargin*}{\relax}
159 \cslet{endaddmargin*}{\relax}


160 \NewDocumentEnvironment{addmargin}{s O{} m}
161 {
162 \LWR@stoppars%
163 \setlength{\LWR@templengthtwo}{#3}
164 \ifblank{#2}
165 {
166     \begin{BlockClass}[
167         \LWR@print@mbox{margin-left:\LWR@printlength{\LWR@templengthtwo}} ;
168         \LWR@print@mbox{margin-right:\LWR@printlength{\LWR@templengthtwo}}
169     ]{addmargin}
170 }
171 {
172     \setlength{\LWR@templengthone}{#2}
173     \begin{BlockClass}[
174         \LWR@print@mbox{margin-left:\LWR@printlength{\LWR@templengthone}} ;
175         \LWR@print@mbox{margin-right:\LWR@printlength{\LWR@templengthtwo}}
176     ]{addmargin}
177 }
178 }
179 {\end{BlockClass}\LWR@startpars}
```

Ref to create a starred environment:
https://tex.stackexchange.com/questions/45401/
        use-the-s-star-argument-with-newdocumentenvironment

```
180
181 \ExplSyntaxOn
182 \cs_new:cpn {addmargin*} {\addmargin*}
183 \cs_new_eq:cN {endaddmargin*} \endaddmargin
184 \ExplSyntaxOff
185
186 \DeclareDocumentCommand{\marginline}{m}{\marginpar{#1}}
```

File 432 **lwarp-scrhack.sty**

§ 544   Package **scrhack**

scrhack (*Pkg*)   scrhack is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{scrhack}[2018/03/30]

File 433 **`lwarp-scrlayer.sty`**

§ 545   Package **scrlayer**

(*Emulates or patches code by* Markus Kohm.)

scrlayer (*Pkg*)   scrlayer is emulated.

⚠ Not fully tested!   Please send bug reports!

for HTML output:

```
1 \LWR@ProvidesPackageDrop{scrlayer}[2018/03/30]
```

```
2 \newcommand*{\DeclareSectionNumberDepth}[2]{}
3 \newcommand*{\DeclareLayer}[2][]{}
4 \newcommand*{\DeclareNewLayer}[2][]{}
5 \newcommand*{\ProvideLayer}[2][]{}
6 \newcommand*{\RedeclareLayer}[2][]{}
7 \newcommand*{\ModifyLayer}[2][]{}
8 \newcommand*{\layerhalign}{}
9 \newcommand*{\layervalign}{}
10 \newcommand*{\layerxoffset}{}
11 \newcommand*{\layeryoffset}{}
12 \newcommand*{\layerwidth}{}
13 \newcommand*{\layerheight}{}
14 \providecommand*{\LenToUnit}[1]{\strip@pt\dimexpr#1*\p@/\unitlength}
15 \newcommand*{\putUL}[1]{}
16 \newcommand*{\putUR}[1]{}
17 \newcommand*{\putLL}[1]{}
18 \newcommand*{\putLR}[1]{}
19 \newcommand*{\putC}[1]{}
20 \newcommand*{\GetLayerContents}[1]{}
21 \newcommand{\IfLayerExists}[3]{#3}
22 \newcommand*{\DestroyLayer}[1]{}
23 \newcommand*{\layercontentsmeasure}{}
24 \newcommand*{\currentpagestyle}{}
25 \newcommand*{\BeforeSelectAnyPageStyle}[1]{}
26 \newcommand*{\AfterSelectAnyPageStyle}[1]{}
27 \newcommand*{\DeclarePageStyleAlias}[2]{}
28 \newcommand*{\DeclareNewPageStyleAlias}[2]{}
29 \newcommand*{\ProvidePageStyleAlias}[2]{}
30 \newcommand*{\RedeclarePageStyleAlias}[2]{}
31 \newcommand*{\DestroyPageStyleAlias}[1]{}
32 \newcommand*{\GetRealPageStyle}[1]{}
33 \newcommand*{\DeclarePageStyleByLayers}[3][]{}
34 \newcommand*{\DeclareNewPageStyleByLayers}[3][]{}
35 \newcommand*{\ProvidePageStyleByLayers}[3][]{}
36 \newcommand*{\RedeclarePageStyleByLayers}[3][]{}
37 \NewDocumentCommand{\ForEachLayerOfPageStyle}{s m m}{}
38 \newcommand*{\AddLayersToPageStyle}[2]{}
39 \newcommand*{\AddLayersAtBeginOfPageStyle}[2]{}
40 \newcommand*{\AddLayersAtEndOfPageStyle}[2]{}
41 \newcommand*{\RemoveLayersFromPageStyle}[2]{}
42 \newcommand*{\AddLayersToPageStyleBeforeLayer}[3]{}
43 \newcommand*{\AddLayersToPageStyleAfterLayer}[3]{}
44 \newcommand*{\UnifyLayersAtPageStyle}[1]{}
45 \newcommand*{\ModifyLayerPageStyleOptions}[2]{}
```

```
46 \newcommand*{\AddToLayerPageStyleOptions}[2]{}
47 \newcommand{\IfLayerPageStyleExists}[3]{#3}
48 \newcommand{\IfRealLayerPageStyleExists}[3]{#3}
49 \newcommand{\IfLayerAtPageStyle}[4]{#4}
50 \newcommand{\IfSomeLayerAtPageStyle}[4]{#4}
51 \newcommand{\IfLayersAtPageStyle}[4]{#4}
52 \newcommand*{\DestroyRealLayerPageStyle}[1]{}
53 \@ifundefined{footheight}{\newlength\footheight}{}
54 \DeclareDocumentCommand{\automark}{s o m}{}
55 \DeclareDocumentCommand{\manualmark}{}{}
56 \DeclareDocumentCommand{\MakeMarkcase}{m}{#1}
```

```
57 \newcommand{\partmarkformat}{}
58 \if@chapter
59 \newcommand{\chaptermarkformat}{}
60 \fi
61 \newcommand{\sectionmarkformat}{}
62 \DeclareDocumentCommand{\GenericMarkFormat}{m}{}
```

```
63 \newcommand*{\@mkleft}[1]{}
64 \newcommand*{\@mkright}[1]{}
65 \newcommand*{\@mkdouble}[1]{}
66 \newcommand*{\@mkboth}[2]{}
67 \newcommand*{\scrlayerInitInterface}[1][]{}
68 \newcommand{\scrlayerAddToInterface}[3][]{}
69 \newcommand{\scrlayerAddCsToInterface}[3][]{}
70 \newcommand{\scrlayerOnAutoRemoveInterface}[2][]{}
```

File 434   **`lwarp-scrlayer-notecolumn.sty`**

§ 546   Package   # scrlayer-notecolumn

(*Emulates or patches code by* Markus Kohm.)

scrlayer-notecolumn (*Pkg*)   scrlayer-notecolumn is emulated.

⚠   Not fully tested!   Please send bug reports!

for HTML output: 1 \LWR@ProvidesPackageDrop{scrlayer-notecolumn}[2018/02/02]

```
2 \newcommand*{\DeclareNoteColumn}[2][]{}
3 \newcommand*{\DeclareNewNoteColumn}[2][]{}
4 \newcommand*{\ProvideNoteColumn}[2][]{}
5 \newcommand*{\RedeclareNoteColumn}[2][]{}
6 \NewDocumentCommand{\makenote}{s o m}{\marginpar{#3}}
7 \newcommand*{\syncwithnotecolumn}[1][]{}
8 \newcommand*{\syncwithnotecolumns}[1][]{}
9 \newcommand*{\clearnotecolumn}[1][]{}
10 \newcommand*{\clearnotecolumns}[1][]{}
```

File 435   **`lwarp-scrlayer-scrpage.sty`**

§ 547   Package   # scrlayer-scrpage

(*Emulates or patches code by* Markus Kohm.)

scrlayer-scrpage (*Pkg*) **scrlayer-scrpage** is ignored.

⚠ Not fully tested! Please send bug reports!

for HTML output: 1 \LWR@ProvidesPackageDrop{scrlayer-scrpage}[2018/03/30]

2 \@ifundefined{footheight}{\newlength\footheight}{}
3 \NewDocumentCommand{\lehead}{s o m}{}
4 \NewDocumentCommand{\cehead}{s o m}{}
5 \NewDocumentCommand{\rehead}{s o m}{}
6 \NewDocumentCommand{\lohead}{s o m}{}
7 \NewDocumentCommand{\cohead}{s o m}{}
8 \NewDocumentCommand{\rohead}{s o m}{}
9 \NewDocumentCommand{\lefoot}{s o m}{}
10 \NewDocumentCommand{\cefoot}{s o m}{}
11 \NewDocumentCommand{\refoot}{s o m}{}
12 \NewDocumentCommand{\lofoot}{s o m}{}
13 \NewDocumentCommand{\cofoot}{s o m}{}
14 \NewDocumentCommand{\rofoot}{s o m}{}
15 \NewDocumentCommand{\ohead}{s o m}{}
16 \NewDocumentCommand{\chead}{s o m}{}
17 \NewDocumentCommand{\ihead}{s o m}{}
18 \NewDocumentCommand{\ofoot}{s o m}{}
19 \NewDocumentCommand{\cfoot}{s o m}{}
20 \NewDocumentCommand{\ifoot}{s o m}{}

21 \NewDocumentCommand{\automark}{som}{}
22 \newcommand*{\manualmark}{}

23 \DeclareDocumentCommand{\MakeMarkcase}{m}{#1}

24 \let\headmark\leftmark
25 \providecommand{\pnumfont}{\normalfont}%
26 \DeclareRobustCommand\pagemark{{\pnumfont{\thepage}}}%

27 \newcommand*{\defpairofpagestyles}[3][]{}
28 \newcommand*{\newpairofpagestyles}[3][]{}
29 \newcommand*{\renewpairofpagestyles}[3][]{}
30 \newcommand*{\providepairofpagestyles}[3][]{}

31 \newcommand*{\clearmainofpairofpagestyles}{}
32 \newcommand*{\clearplainofpairofpagestyles}{}
33 \newcommand*{\clearpairofpagestyles}{}
34 \newcommand*{\clearscrheadings}{}
35 \newcommand*{\clearscrheadfoot}{}
36 \newcommand*{\clearscrplain}{}

37 \NewDocumentCommand{\deftriplepagestyle}{m o o m m m m m m}{}
38 \NewDocumentCommand{\newtriplepagestyle}{m o o m m m m m m}{}
39 \NewDocumentCommand{\renewtriplepagestyle}{m o o m m m m m m}{}
40 \NewDocumentCommand{\providetriplepagestyle}{m o o m m m m m m}{}
41 \newcommand*{\defpagestyle}[3]{}
42 \newcommand*{\newpagestyle}[3]{}
43 \newcommand*{\providepagestyle}[3]{}
44 \newcommand*{\renewpagestyle}[3]{}

File 436 **`lwarp-scrpage2.sty`**

§ 548 Package **scrpage2**

(*Emulates or patches code by* Markus Kohm.)

scrpage2 (*Pkg*) scrpage2 is ignored.

⚠ Not fully tested! Please send bug reports!

for HTML output:

```
1 \LWR@ProvidesPackageDrop{scrpage2}[2018/03/30]
```

```
2 \@ifundefined{footheight}{\newlength\footheight}{}
3 \NewDocumentCommand{\lehead}{o m}{}
4 \NewDocumentCommand{\cehead}{o m}{}
5 \NewDocumentCommand{\rehead}{o m}{}
6 \NewDocumentCommand{\lohead}{o m}{}
7 \NewDocumentCommand{\cohead}{o m}{}
8 \NewDocumentCommand{\rohead}{o m}{}
9 \NewDocumentCommand{\lefoot}{o m}{}
10 \NewDocumentCommand{\cefoot}{o m}{}
11 \NewDocumentCommand{\refoot}{o m}{}
12 \NewDocumentCommand{\lofoot}{o m}{}
13 \NewDocumentCommand{\cofoot}{o m}{}
14 \NewDocumentCommand{\rofoot}{o m}{}
15 \NewDocumentCommand{\ohead}{o m}{}
16 \NewDocumentCommand{\chead}{o m}{}
17 \NewDocumentCommand{\ihead}{o m}{}
18 \NewDocumentCommand{\ofoot}{o m}{}
19 \NewDocumentCommand{\cfoot}{o m}{}
20 \NewDocumentCommand{\ifoot}{o m}{}
21 \DeclareDocumentCommand{\automark}{o m}{}
22 \DeclareDocumentCommand{\manualmark}{}{}
23 \DeclareDocumentCommand{\MakeMarkcase}{m}{#1}
24 \NewDocumentCommand{\deftripstyle}{m o o m m m m m m}{}
25 \NewDocumentCommand{\defpagestyle}{s m m m}{}
26 \NewDocumentCommand{\newpagestyle}{s m m m}{}
27 \NewDocumentCommand{\renewpagestyle}{s m m m}{}
28 \NewDocumentCommand{\providepagestyle}{s m m m}{}
29 \newcommand{\partmarkformat}{}
30 \if@chapter
31 \newcommand{\chaptermarkformat}{}
32 \fi
33 \newcommand{\sectionmarkformat}{}
34 \newcommand{\subsectionmarkformat}{}
35 \newcommand{\subsubsectionmarkformat}{}
36 \newcommand{\paragraphmarkformat}{}
37 \newcommand{\subparagraphmarkformat}{}
38
39 \newcommand*{\clearscrheadings}{}
40 \newcommand*{\clearscrheadfoot}{}
41 \newcommand*{\clearscrplain}{}
```

File 437 **`lwarp-section.sty`**

§ 549 Package **section**

section (*Pkg*) section is ignored.

(*Emulates or patches code by* OLIVER PRETZEL.)

for HTML output: 1 `\LWR@ProvidesPackageDrop{section}`

```
2 \ifx\chapter\undefined
3  \def\chsize{\Large}\def\hdsize{\huge}\else
4  \def\chsize{\huge}\def\hdsize{\Huge}
5 \fi
6 \let\ttsize\LARGE
7 \let\ausize\large
8 \let\dasize\large
9 \let\secsize\Large
10 \let\subsize\large
11 \let\hdpos\raggedright
12 \newcounter{hddepth}
13 \let\fpind\relax
14 \def\ttfnt{}
15 \def\hdfnt{}
16 \def\fefnt{}
17 \def\thfnt{}
18 \def\pgfnt{}
19 \def\hmkfnt{}
20 \let\mkcse\uppercase
21 \def\hddot{}
22 \def\cpdot{:}
23 \def\nmdot{}
24 \ifx\secindent\undefined
25  \newdimen\secindent
26  \newskip\secpreskp
27  \newskip\secpstskp
28  \newdimen\subindent
29  \newskip\subpreskp
30  \newskip\subpstskp
31  \newskip\parpstskp
32  \newcount\c@hddepth
33 \fi
```

File 438 **`lwarp-sectionbreak.sty`**

§ 550 Package **sectionbreak**

(*Emulates or patches code by* MICHAL HOFTICH.)

sectionbreak (*Pkg*)   sectionbreak is patched for use by lwarp.

for HTML output: 1 `\LWR@ProvidesPackagePass{sectionbreak}[2018-01-03]`

```
2 \renewcommand\asterism{\HTMLunicode{2042}}
3
4 \renewcommand\pre@sectionbreak{}
5 \renewcommand\post@sectionbreak{}
6
7 \renewcommand\print@sectionbreak[1]{%
8 \begin{center}
9 #1
10 \end{center}
11 }
12
```

File 439   **`lwarp-sectsty.sty`**

§ 551   Package   # sectsty

(*Emulates or patches code by* ROWLAND MCDONNELL.)

sectsty (*Pkg*)   sectsty is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{sectsty}[2002/02/25]`

```
2 \newcommand*{\partfont}          [1] {}
3 \newcommand*{\partnumberfont}    [1] {}
4 \newcommand*{\parttitlefont}     [1] {}
5 \newcommand*{\chapterfont}       [1] {}
6 \newcommand*{\chapternumberfont} [1] {}
7 \newcommand*{\chaptertitlefont}  [1] {}
8 \newcommand*{\sectionfont}       [1] {}
9 \newcommand*{\subsectionfont}    [1] {}
10 \newcommand*{\subsubsectionfont} [1] {}
11 \newcommand*{\paragraphfont}     [1] {}
12 \newcommand*{\subparagraphfont}  [1] {}
13 \newcommand*{\minisecfont}  [1] {}
14 \newcommand*{\allsectionsfont}[1] {}
15 \newcommand{\nohang}{}
```

\sectionrule is only to be used in *font commands, thus it is ignored.

```
16 \newcommand*{\sectionrule}[5]{}
17
18 \def\ulemheading#1#2{}
```

File 440   **`lwarp-selectp.sty`**

§ 552   Package   # selectp

selectp (*Pkg*)   selectp is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{selectp}% no date given`

```
2 \newcommand*{\outputonly}[1]{}
```

File 441 **`lwarp-semantic-markup.sty`**

§ 553   Package   # semantic-markup

(*Emulates or patches code by* ANDREW A. CASHNER.)

semantic-markup (*Pkg*)   semantic-markup is patched for use by lwarp.

⚠   If using the endnotes option, add \theendnotes where desired.

for HTML output: 1 \LWR@ProvidesPackagePass{semantic-markup}[2018/05/21]

The endnotes must be printed by the user before the end of the document, since the end is after the HTML footer, etc.

```
2 \ifendnotes
3 \RenewDocumentCommand{\SetupEndnotes}{}{%
4     \let\footnote=\endnote
5 %    \AtEndDocument{\DoBeforeEndnotes{\EndnoteFont\theendnotes}}%
6 }
7 \fi
```

HTML unicode characters from musicography are used.

```
8 \RequirePackage{musicography}
9
10 \let\fl\musFlat
11 \let\sh\musSharp
12 \let\na\musNatural
```

The \musfig is placed inside a hashed image, with a simple alt tag.

```
13 \RequirePackage{amsmath}
14
15 \RenewDocumentCommand{\musfig}{ m m }{%
16     \LWR@subsingledollar*%
17         {#1/#2}% alt tag
18         {musfig}% addl' hashing
19         {% contents
20             \LWR@origensuredmath{%
21                 \genfrac{}{}{0pt}{1}{\text{#1}}{\text{#2}}%
22             }%
23         }%
24 }
```

The \meter is taken from musicography, and becomes a hashed image with a simple alt tag.

```
25 \RenewDocumentCommand{\meter}{ m m }{%
26     \musMeter{#1}{#2}%
27 }
```

File 442 **`lwarp-seqsplit.sty`**

§ 554   Package   **seqsplit**

(*Emulates or patches code by* Boris Veytsman.)

seqsplit (*Pkg*)   seqsplit is patched for use by lwarp.

For HTML output, the results are similar to print mode, and respond to window size.

⚠ SVG math results   For SVG math, the output differs from print mode in that the contents are formatted in a minipage, which is then inline with the surrounding math.

For MATHJAX, the contents are used as-is.

for HTML output: 1 \LWR@ProvidesPackagePass{seqsplit}[2006/08/07]

Special handling because lwarp uses a box for SVG math, which does not normally allow line breaks, so a print-mode minipage must be used to allow line breaks. The minipage will not be wrapped inline with any surrounding math.

```
2 \begin{warpHTML}
3 \LetLtxMacro\LWR@orig@seqsplit\seqsplit
4
5 \renewcommand*{\seqsplit}[1]{%
6     \ifmmode%
7         \begin{LWR@print@minipage}{6in}%
8         \LWR@orig@seqsplit{#1}%
9         \end{LWR@print@minipage}%
10    \else%
11        \InlineClass[word-wrap:break-word]{seqsplit}{\LWR@orig@seqsplit{#1}}%
12    \fi
13 }
```

Between characters, an empty HTML comment is placed to allow a line wrap in the HTML source, without adding spaces in the output.

```
14 \AtBeginDocument{
15     \newcommand*{\LWR@HTML@seqinsert}{%
16         \LWR@htmlcomment{ }%
17     }
18     \LWR@formatted{seqinsert}
19 }
20 \end{warpHTML}
21
22 \begin{warpMathJax}
23 \CustomizeMathJax{\newcommand{\seqsplit}[1]{#1}}
24 \end{warpMathJax}
```

File 443 **`lwarp-setspace.sty`**

§ 555   Package **setspace**

(*Emulates or patches code by* ROBIN FAIRBAIRNS.)

setspace (*Pkg*)   setspace is emulated.

Discard all options for lwarp-setspace:

for HTML output:
```
1 \LWR@ProvidesPackageDrop{setspace}[2011/12/19]
2
3 \newcommand*{\setstretch}[1]{}
4 \newcommand*{\SetSinglespace}[1]{}
5 \newcommand*{\singlespacing}{}
6 \newcommand*{\onehalfspacing}{}
7 \newcommand*{\doublespacing}{}
8
9 \newenvironment*{singlespace}
10 {
11 \LWR@forcenewpage
12 \BlockClass{singlespace}
13 }
14 {\endBlockClass}
15
16 \newenvironment*{singlespace*}
17 {
18 \LWR@forcenewpage
19 \BlockClass{singlespace}
20 }
21 {\endBlockClass}
22
23 \newenvironment*{spacing}[1]{
24
25 }{
26
27 }
28
29 \newenvironment*{onehalfspace}
30 {
31 \LWR@forcenewpage
32 \BlockClass{onehalfspace}
33 }
34 {\endBlockClass}
35
36 \newenvironment*{doublespace}
37 {
38 \LWR@forcenewpage
39 \BlockClass{doublespace}
40 }
41 {\endBlockClass}
```

File 444   `lwarp-shadethm.sty`

§ 556   Package   # shadethm

(*Emulates or patches code by* JIM HEFFERON.)

shadethm (*Pkg*)   shadethm is patched for use by lwarp.

for HTML output: 1 `\LWR@ProvidesPackagePass{shadethm}[1999/11/23]`

2 `\newenvironment{LWR@HTML@shadebox}`
3 `{%`
4 `    \convertcolorspec{named}{shadethmcolor}{HTML}\LWR@tempcolor%`
5 `    \convertcolorspec{named}{shaderulecolor}{HTML}\LWR@tempcolortwo%`
6 `    \begin{BlockClass}[%`
7 `        background: \LWR@origpound\LWR@tempcolor ;`
8 `        border: 1px solid \LWR@origpound\LWR@tempcolortwo ;`
9 `    ]{shadebox}`
10 `}%`
11 `{\end{BlockClass}}`
12 `\LWR@formattedenv{shadebox}`

File 445   `lwarp-shadow.sty`

§ 557   Package   # shadow

(*Emulates or patches code by* MAURO ORLANDINI.)

shadow (*Pkg*)   shadow is emulated.

for HTML output: Discard all options for lwarp-shadow:

1 `\LWR@ProvidesPackageDrop{shadow}[2003/02/19]`

2 `\newdimen\sboxsep`
3 `\newdimen\sboxrule`
4 `\newdimen\sdim`
5
6 `\newcommand{\shabox}[1]{%`
7 `\InlineClass{shabox}{#1}%`
8 `}`

File 446   `lwarp-shapepar.sty`

§ 558   Package   # shapepar

(*Emulates or patches code by* DONALD ARSENEAU.)

shapepar (*Pkg*)   shapepar is patched for use by lwarp. Shapes appear in print mode, as well as inside a lateximage, but are ignored for HTML.

1 `\LWR@ProvidesPackagePass{shapepar}[2013/03/26]`

2 `\newcommand*{\LWR@HTML@shapepar}[2][]{}`
3 `\LWR@formatted{shapepar}`
4
5 `\NewDocumentCommand{\LWR@HTML@cutout}{m d()}{}`
6 `\LWR@formatted{cutout}`

File 447   **`lwarp-showidx.sty`**

§ 559   Package   # showidx

showidx (*Pkg*)   showidx is ignored.

Discard all options for lwarp-showidx:

1 `\LWR@ProvidesPackageDrop{showidx}[2014/09/29]`

`\@wrindex` is redefined `\AtBeginDocument` by the lwarp core.

File 448   **`lwarp-showkeys.sty`**

§ 560   Package   # showkeys

(*Emulates or patches code by* David Carlisle, Morten Høgholm.)

showkeys (*Pkg*)   showkeys is ignored.

Discard all options for lwarp-showkeys:

1 `\LWR@ProvidesPackageDrop{showkeys}[2014/10/28]`

2 `\NewDocumentCommand{\showkeys}{s}{}`

File 449   **`lwarp-showlabels.sty`**

§ 561   Package   # showlabels

showlabels (*Pkg*)   showlabels is ignored.

1 `\LWR@ProvidesPackageDrop{showlabels}[2021/10/27]`

2 `\providecommand{\showlabelfont}{}`
3 `\providecommand{\showlabelsetlabel}[1]{}`
4 `\newcommand*{\showlabels}[2][]{}`
5 `\newcommand*{\showlabelrefline}{}`
6 `\newcommand*{\showlabelsinline}{}`

File 450   `lwarp-showtags.sty`

§ 562   Package   # showtags

showtags (*Pkg*)   **showtags** is ignored.

1 \LWR@ProvidesPackageDrop{showtags}% no version is given

2 \newcommand{\thecitetag}[1]{}

File 451   `lwarp-shuffle.sty`

§ 563   Package   # shuffle

(*Emulates or patches code by* JULIAN GILBEY AND ANTOINE LEJAY.)

shuffle (*Pkg*)   **shuffle** is emulated for SVG math, and also emulated for MATHJAX.

The font used for **shuffle** may not render correctly when converted to SVG math, so a `picture` environment drawing is used instead.

For MATHJAX, the Unicode character is used, and for \cshuffle a \bar is added.

1 \LWR@ProvidesPackageDrop{shuffle}[2008/10/27]
2 \LWR@origRequirePackage{lwarp-common-mathjax-overlaysymbols}

3 \newcommand*{\LWR@shuffle@start}{%
4   \hspace*{.2em}
5   \begin{picture}(.75,0.65)
6   \setlength{\unitlength}{1em}
7   \put(0,0){\line(1,0){.75}}
8   \put(0,0){\line(0,1){.5}}
9   \put(.375,0){\line(0,1){.5}}
10   \put(.75,0){\line(0,1){.5}}
11 }
12
13 \newcommand*{\LWR@shuffle@finish}{%
14   \end{picture}
15   \hspace*{.75em}
16   \hspace*{.2em}
17 }
18
19 \newcommand*{\shuffle}{%
20   \LWR@shuffle@start%
21   \LWR@shuffle@finish%
22 }
23
24 \newcommand*{\cshuffle}{%
25   \LWR@shuffle@start%
26   \put(.05,.65){\line(1,0){.65}}%
27   \LWR@shuffle@finish%
28 }

```
29 \begin{warpMathJax}
30 \CustomizeMathJax{\newcommand{\shuffle}{\mathbin{\unicode{0x29E2}}}}
31 \CustomizeMathJax{\newcommand{\cshuffle}{%
32    \mathbin{\LWRoverlaysymbols{\raise{.6ex}{-}}{\unicode{0x29E2}}}%
33 }}
34 \end{warpMathJax}
```

File 452   **`lwarp-sidecap.sty`**

§ 564   Package   # sidecap

(*Emulates or patches code by* ROLF NIEPRASCHK, HUBERT GÄSSLEIN.)

sidecap (*Pkg*)   sidecap is emulated.

for HTML output: Discard all options for lwarp-sidecap.

1 \LWR@ProvidesPackageDrop{sidecap}[2003/06/06]

See:
http://tex.stackexchange.com/questions/45401/
use-the-s-star-argument-with-newdocumentenvironment
regarding the creation of starred environments with xparse.

```
2 \NewDocumentEnvironment{SCtable}{soo}
3 {\IfValueTF{#3}{\table[#3]}{\table}}
4 {\endtable}
5
6 \ExplSyntaxOn
7 \cs_new:cpn {SCtable*} {\SCtable*}
8 \cs_new_eq:cN {endSCtable*} \endSCtable
9 \ExplSyntaxOff
10
11
12 \NewDocumentEnvironment{SCfigure}{soo}
13 {\IfValueTF{#3}{\figure[#3]}{\figure}}
14 {\endfigure}
15
16 \ExplSyntaxOn
17 \cs_new:cpn {SCfigure*} {\SCfigure*}
18 \cs_new_eq:cN {endSCfigure*} \endSCfigure
19 \ExplSyntaxOff
20
21
22 \newenvironment*{wide}{}{}
```

File 453   **`lwarp-sidenotes.sty`**

§ 565   Package   # sidenotes

(*Emulates or patches code by* ANDY THOMAS, OLIVER SCHEBAUM.)

sidenotes (*Pkg*)   Patched for lwarp.

for HTML output:

Load the original package:

```
1 \LWR@ProvidesPackagePass{sidenotes}
```

The following patch sidenotes for use with lwarp.

An ARIA note role is not assigned since the caption is an important part of the figure.

\sidecaption   * [⟨*entry*⟩] [⟨*offset*⟩] {⟨*text*⟩}

```
2 \VerifyCommand[lwarp][sidenotes]{\sidecaption}{2EFE2196F612943BCF13746EC12E69D6}
3
4 \RenewDocumentCommand \sidecaption {s o o m}
5 {
6     \LWR@stoppars
7     \begingroup
8  \captionsetup{style=sidecaption}%
9  \IfBooleanTF{#1}
10 { % starred
11    \begin{BlockClass}[border:none ; box-shadow:none]{marginblock}%
12    \caption*{#4}%
13    \end{BlockClass}
14 }
15 { % unstarred
16 \IfNoValueOrEmptyTF{#2}
17   {\def\@sidenotes@sidecaption@tof{#4}}
18   {\def\@sidenotes@sidecaption@tof{#2}}
19   \begin{BlockClass}[border:none ; box-shadow:none]{marginblock}%
20   \caption[\@sidenotes@sidecaption@tof]{#4}
21   \end{BlockClass}
22 }
23   \endgroup
24   \LWR@startpars
25 }
```

Borrowed from the lwarp version of keyfloat:

```
26 \NewDocumentEnvironment{KFLTsidenotes@marginfloat}{O{-1.2ex} m}
27 {% start
28    \LWR@BlockClassWP{float:right; width:2in; margin:10pt}{}{marginblock}%
29    \renewcommand*{\@captype}{#2}%
30 }
31 {%
32    \endLWR@BlockClassWP%
33 }
34
35 \RenewDocumentEnvironment{marginfigure}{o}
36  {\begin{KFLTsidenotes@marginfloat}{figure}}
37  {\end{KFLTsidenotes@marginfloat}}
38
39 \RenewDocumentEnvironment{margintable}{o}
40  {\begin{KFLTsidenotes@marginfloat}{table}}
41  {\end{KFLTsidenotes@marginfloat}}
```

The following were changed by sidenotes, and now are reset back to their lwarp-supported originals:

Restoring the definition from the LaTeX 2ε article.cls source:

```
42 \renewenvironment{figure*}
43              {\@dblfloat{figure}}
44              {\end@dblfloat}
45
46 \renewenvironment{table*}
47              {\@dblfloat{table}}
48              {\end@dblfloat}
```

For MathJax:

⚠ Note that sidenotes does not support \sidenote inside math in print mode. Use \sidenotemark and \sidenotetext instead.

```
49 \begin{warpMathJax}
50 \providecommand{\sidenotename}{sidenote}
51 \appto\LWR@syncnotenumbers{\LWR@synconenotenumber{LWRsidenote}{\thesidenote}}
52 \appto\LWR@syncnotenames{\LWR@synconenotename{LWRsidenote}{\sidenotename}}
53 \CustomizeMathJax{\def\LWRsidenote{1}}
54 \CustomizeMathJax{\newcommand{\sidenotemark}[1][\LWRsidenote]{{}^{\mathrm{#1}}}}
55 \end{warpMathJax}
```

The following is not defined since is not allowed inside math in print mode, and also would have to be modified to parse the optional offset argument:

$\quad$ \CustomizeMathJax{\newcommand{\sidenote}[2][\LWRsidenote]{{}^{\mathrm{#1}}}}

File 454   **`lwarp-simplebnf.sty`**

§ 566   Package   # simplebnf

(*Emulates or patches code by* JAY LEE.)

simplebnf (*Pkg*)   simplebnf is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{simplebnf}[2023-11-25]

The entire object is placed inside a lateximage whose alt text is the LaTeX source bnf expression.

```
2 \ExplSyntaxOn
3
4 \VerifyEnvironment[lwarp][simplebnf]{bnf}
5    {A7E8911B9291D4EB7C1CD8366CD75341}{3B45D7D9107687D718F5303B6632776C}
6
7 \RenewDocumentEnvironment { bnf } { d() O{llcll} +b }
8  {
9     \begin{lateximage}[bnf:\space\detokenize{#3}]?%
10
11     \IfNoValueF { #1 }
12       { \keys_set:nn { simplebnf } { #1 } }
13
14     \__simplebnf_build_grammar:n { #3 }
15
```

```
16        \begin{@simplebnf_tblr_env}[expand=\l__simplebnf_table_tl]{#2}
17          \tl_use:N \l__simplebnf_table_tl
18        \end{@simplebnf_tblr_env}
19      \end{lateximage}%                              lwarp
20    }
21    { }
22
23 \VerifyEnvironment[lwarp][simplebnf]{bnfgrammar}
24    {E7326E6CAE6E35827E866B4A08C5CEA8}{A9B27A2478E8BD67B19E94ECF8A44F14}
25
26 \RenewDocumentEnvironment { bnfgrammar } { O{llcll} O{[^\|]}\|[^\|]} O{\|\|} +b }
27    {
28      \msg_warning:nn { simplebnf } { dep }
29      \begin{center}
30        \begin{lateximage}[bnf:\space\detokenize{#4}]?%
31          \begin{tabular}{#1}
32            \@dep__simplebnf_typeset_grammar:nnn { #2 } { #3 } { #4 }
33            \tl_use:N \l__simplebnf_table_tl
34          \end{tabular}
35        \end{lateximage}%                            lwarp
36      \end{center}
37    }
38    { }
39
40 \ExplSyntaxOff
```

File 455   **`lwarp-SIunits.sty`**

§ 567   Package   **SIunits**

(*Emulates or patches code by* MARCEL HELDOORN.)

SIunits (*Pkg*)   SIunits is patched for use by lwarp.

For SVG math, it is recommended to use \unit where possible, which combines the entire expression into a single lateximage, and adds the alt tag containing the LATEX code, allowing for copy/paste. When units are used outside of the \unit macro, each unit macro will have its own lateximage, and each will have the alt tag set according to \MathImageAltText, which defaults to (math image).

For MATHJAX, individual units used in text will appear as SVG images, since \ensuremath is used in the original defintions, and \ensuremath often has expressions which do not work well in MATHJAX, so it is always forced to an SVG image. If, however, \unit is used, the result is expressed with MATHJAX instead of an SVG image.

for HTML output:
```
1 \LWR@ProvidesPackagePass{SIunits}[2007/12/02]
```

Patched for copy/paste with the HTML alt tag:

```
2 \ifbool{mathjax}{
3   \DeclareRobustCommand{\LWR@HTML@unit}[2]{%
4       \begingroup%
5       \boolfalse{LWR@HTMLsanitize@tmpb@removebackslash}%
6       \LWR@subsingledollar*%  lwarp
7       {% alt tag
8           \textbackslash{}unit%
```

```
 9              \{\LWR@HTMLsanitizedetokenized{\detokenize{#1}}\}%
10              \{ \LWR@HTMLsanitizedetokenized{\detokenize{#2}}\}% extra space
11          }%
12          {SIunits}% add'l hashing
13          {%
14              #1\,{#2}%
15          }% contents
16          \endgroup%
17      }
18 }{% not MathJax
19     \DeclareRobustCommand{\LWR@HTML@unit}[2]{%
20          \@inunitcommandtrue%    original
21          \LWR@subsingledollar*%  lwarp
22          {% alt tag
23          \textbackslash{}unit\{\LWR@HTMLsanitizedetokenized{\detokenize{#1}}\}%
24              \{ \LWR@HTMLsanitizedetokenized{\detokenize{#2}}\}% extra space
25          }%
26          {SIunits}% add'l hashing
27          {%
28              \LWR@origensuredmath{%  lwarp modification
29                  \SI@fstyle{%
30                      {#1}\@qsk\period@active{#2}%
31                  }%  original
32              }%
33          }% contents
34          \@inunitcommandfalse%   original
35      }
36 }% not MathJax
37 \LWR@formatted{unit}
```

For MATHJAX:

```
38 \begin{warpMathJax}
39 \LWR@infoprocessingmathjax{SIunits}
40
41 \CustomizeMathJax{\newcommand{\one}{}}
42 \CustomizeMathJax{\newcommand{\meter}{\metre}}
43 \CustomizeMathJax{\newcommand{\deka}{\deca}}
44 \CustomizeMathJax{\newcommand{\dekad}{\decad}}
45 \CustomizeMathJax{\newcommand{\per}{/}}
46 \CustomizeMathJax{\newcommand{\usk}{\;}}
47 \CustomizeMathJax{\newcommand{\unit}[2]{#1\,{#2}}}
48 \CustomizeMathJax{\newcommand{\power}[2]{#1^{#2}}}
49
50 \AtBeginDocument{%
51 \if@redefsquare
52  \CustomizeMathJax{\renewcommand{\square}[1]{\power{#1}{2}}}
53 \else
54   \if@defsquaren
55    \CustomizeMathJax{\newcommand{\squaren}[1]{\power{#1}{2}}}
56   \else
57     \CustomizeMathJax{\renewcommand{\square}[1]{\power{#1}{2}}}
58   \fi %\if@defsquaren
59 \fi   %\if@redefsquare
60 }     %\AtBeginDocument
61
62 \CustomizeMathJax{\newcommand{\squared}{^{2}}}
63 \CustomizeMathJax{\newcommand{\cubic}[1]{\power{#1}{3}}}
64 \CustomizeMathJax{\newcommand{\cubed}{^{3}}}
65 \CustomizeMathJax{\newcommand{\fourth}[1]{\power{#1}{4}}}
```

```
66 \CustomizeMathJax{\newcommand{\reciprocal}[1]{\power{#1}{-1}}}
67 \CustomizeMathJax{\newcommand{\rp}{\reciprocal}}
68 \CustomizeMathJax{\newcommand{\rpsquare}[1]{\power{#1}{-2}}}
69 \CustomizeMathJax{\newcommand{\rpsquared}{^{-2}}}
70 \CustomizeMathJax{\newcommand{\rpcubic}[1]{\power{#1}{-3}}}
71 \CustomizeMathJax{\newcommand{\rpcubed}{^{-3}}}
72 \CustomizeMathJax{\newcommand{\rpfourth}[1]{\power{#1}{-4}}}
73 \CustomizeMathJax{\newcommand{\yocto}{\mathrm{y}}}
74 \CustomizeMathJax{\newcommand{\zepto}{\mathrm{z}}}
75 \CustomizeMathJax{\newcommand{\atto}{\mathrm{a}}}
76 \CustomizeMathJax{\newcommand{\femto}{\mathrm{f}}}
77 \CustomizeMathJax{\newcommand{\pico}{\mathrm{p}}}
78 \CustomizeMathJax{\newcommand{\nano}{\mathrm{n}}}
79 \CustomizeMathJax{\newcommand{\micro}{\mathrm{\unicode{x00B5}}}}
80 \CustomizeMathJax{\newcommand{\milli}{\mathrm{m}}}
81 \CustomizeMathJax{\newcommand{\centi}{\mathrm{c}}}
82 \CustomizeMathJax{\newcommand{\deci}{\mathrm{d}}}
83 \CustomizeMathJax{\newcommand{\deca}{\mathrm{da}}}
84 \CustomizeMathJax{\newcommand{\hecto}{\mathrm{h}}}
85 \CustomizeMathJax{\newcommand{\kilo}{\mathrm{k}}}
86 \CustomizeMathJax{\newcommand{\mega}{\mathrm{M}}}
87 \CustomizeMathJax{\newcommand{\giga}{\mathrm{G}}}
88 \CustomizeMathJax{\newcommand{\tera}{\mathrm{T}}}
89 \CustomizeMathJax{\newcommand{\peta}{\mathrm{P}}}
90 \CustomizeMathJax{\newcommand{\exa}{\mathrm{E}}}
91 \CustomizeMathJax{\newcommand{\zetta}{\mathrm{Z}}}
92 \CustomizeMathJax{\newcommand{\yotta}{\mathrm{Y}}}
93 \CustomizeMathJax{\newcommand{\yoctod}{\power{10}{-24}}}
94 \CustomizeMathJax{\newcommand{\zeptod}{\power{10}{-21}}}
95 \CustomizeMathJax{\newcommand{\attod}{\power{10}{-18}}}
96 \CustomizeMathJax{\newcommand{\femtod}{\power{10}{-15}}}
97 \CustomizeMathJax{\newcommand{\picod}{\power{10}{-12}}}
98 \CustomizeMathJax{\newcommand{\nanod}{\power{10}{-9}}}
99 \CustomizeMathJax{\newcommand{\microd}{\power{10}{-6}}}
100 \CustomizeMathJax{\newcommand{\millid}{\power{10}{-3}}}
101 \CustomizeMathJax{\newcommand{\centid}{\power{10}{-2}}}
102 \CustomizeMathJax{\newcommand{\decid}{\power{10}{-1}}}
103 \CustomizeMathJax{\newcommand{\decad}{\power{10}{1}}}
104 \CustomizeMathJax{\newcommand{\hectod}{\power{10}{2}}}
105 \CustomizeMathJax{\newcommand{\kilod}{\power{10}{3}}}
106 \CustomizeMathJax{\newcommand{\megad}{\power{10}{6}}}
107 \CustomizeMathJax{\newcommand{\gigad}{\power{10}{9}}}
108 \CustomizeMathJax{\newcommand{\terad}{\power{10}{12}}}
109 \CustomizeMathJax{\newcommand{\petad}{\power{10}{15}}}
110 \CustomizeMathJax{\newcommand{\exad}{\power{10}{18}}}
111 \CustomizeMathJax{\newcommand{\zettad}{\power{10}{21}}}
112 \CustomizeMathJax{\newcommand{\yottad}{\power{10}{24}}}
113 \CustomizeMathJax{\newcommand{\gram}{\mathrm{g}}}
114 \CustomizeMathJax{\newcommand{\metre}{\mathrm{m}}}
115 \CustomizeMathJax{\newcommand{\kilogram}{\kilo\gram}}
116 \CustomizeMathJax{\newcommand{\second}{\mathrm{s}}}
117 \CustomizeMathJax{\newcommand{\ampere}{\mathrm{A}}}
118 \CustomizeMathJax{\newcommand{\kelvin}{\mathrm{K}}}
119 \CustomizeMathJax{\newcommand{\mole}{\mathrm{mol}}}
120 \CustomizeMathJax{\newcommand{\candela}{\mathrm{cd}}}
121 \CustomizeMathJax{\newcommand{\radian}{\mathrm{rad}}}
122 \CustomizeMathJax{\newcommand{\steradian}{\mathrm{sr}}}
123 \CustomizeMathJax{\newcommand{\hertz}{\mathrm{Hz}}}
124 \CustomizeMathJax{\newcommand{\newton}{\mathrm{N}}}
125 \CustomizeMathJax{\newcommand{\pascal}{\mathrm{Pa}}}
```

```
126 \CustomizeMathJax{\newcommand{\joule}{\mathrm{J}}}
127 \CustomizeMathJax{\newcommand{\watt}{\mathrm{W}}}
128 \CustomizeMathJax{\newcommand{\coulomb}{\mathrm{C}}}
129 \CustomizeMathJax{\newcommand{\volt}{\mathrm{V}}}
130 \CustomizeMathJax{\newcommand{\farad}{\mathrm{F}}}
131 \CustomizeMathJax{\newcommand{\ohm}{\mathrm{\Omega}}}
132 \CustomizeMathJax{\newcommand{\siemens}{\mathrm{S}}}
133 \CustomizeMathJax{\newcommand{\weber}{\mathrm{Wb}}}
134 \CustomizeMathJax{\newcommand{\tesla}{\mathrm{T}}}
135 \CustomizeMathJax{\newcommand{\henry}{\mathrm{H}}}
136 \CustomizeMathJax{\newcommand{\degreecelsius}{\mathrm{\unicode{x2103}}}}
137 \CustomizeMathJax{\newcommand{\celsius}{\degreecelsius}}
138 \CustomizeMathJax{\newcommand{\lumen}{\mathrm{lm}}}
139 \CustomizeMathJax{\newcommand{\lux}{\mathrm{lx}}}
140 \CustomizeMathJax{\newcommand{\becquerel}{\mathrm{Bq}}}
141 \CustomizeMathJax{\newcommand{\sievert}{\mathrm{Sv}}}
142 \CustomizeMathJax{\newcommand{\katal}{\mathrm{kat}}}
143
144 \ifdef{\radianbase}{
145 \CustomizeMathJax{\newcommand{\radianbase}%
146         {\metre\usk\reciprocal\metre}}
147 \CustomizeMathJax{\newcommand{\steradianbase}%
148         {\squaremetre\usk\rpsquare\metre}}
149 \CustomizeMathJax{\newcommand{\hertzbase}%
150         {\reciprocal\second}}
151 \CustomizeMathJax{\newcommand{\newtonbase}%
152         {\metre\usk\kilogram\usk\second\rpsquared}}
153 \CustomizeMathJax{\newcommand{\pascalbase}%
154         {\reciprocal\metre\usk\kilogram\usk\second\rpsquared}}
155 \CustomizeMathJax{\newcommand{\joulebase}%
156         {\squaremetre\usk\kilogram\usk\second\rpsquared}}
157 \CustomizeMathJax{\newcommand{\wattbase}%
158         {\squaremetre\usk\kilogram\usk\rpcubic\second}}
159 \CustomizeMathJax{\newcommand{\coulombbase}%
160         {\ampere\usk\second}}
161 \CustomizeMathJax{\newcommand{\voltbase}%
162         {\squaremetre\usk\kilogram\usk\rpcubic\second\usk\reciprocal\ampere}}
163 \CustomizeMathJax{\newcommand{\faradbase}%
164     {\rpsquare\metre\usk\reciprocal\kilogram\usk\fourth\second\usk\ampere\squared}}
165 \CustomizeMathJax{\newcommand{\ohmbase}%
166         {\squaremetre\usk\kilogram\usk\rpcubic\second\usk\rpsquare\ampere}}
167 \CustomizeMathJax{\newcommand{\siemensbase}%
168     {\rpsquare\metre\usk\reciprocal\kilogram\usk\cubic\second\usk\ampere\squared}}
169 \CustomizeMathJax{\newcommand{\weberbase}%
170     {\squaremetre\usk\kilogram\usk\second\rpsquared\usk\reciprocal\ampere}}
171 \CustomizeMathJax{\newcommand{\teslabase}%
172         {\kilogram\usk\second\rpsquared\usk\reciprocal\ampere}}
173 \CustomizeMathJax{\newcommand{\henrybase}%
174     {\squaremetre\usk\kilogram\usk\second\rpsquared\usk\rpsquare\ampere}}
175 \CustomizeMathJax{\newcommand{\celsiusbase}%
176         {\kelvin}}
177 \CustomizeMathJax{\newcommand{\lumenbase}%
178         {\candela\usk\squaremetre\usk\rpsquare\metre}}
179 \CustomizeMathJax{\newcommand{\luxbase}%
180         {\candela\usk\squaremetre\usk\rpfourth\metre}}
181 \CustomizeMathJax{\newcommand{\becquerelbase}%
182         {\hertzbase}}
183 \CustomizeMathJax{\newcommand{\graybase}%
184         {\squaremetre\usk\second\rpsquared}}
185 \CustomizeMathJax{\newcommand{\sievertbase}%
```

```
186         {\graybase}}
187 \CustomizeMathJax{\newcommand{\katalbase}%
188         {\rp\second\usk\mole }}
189 }{}
190
191 \ifdef{\derradian}{
192 \CustomizeMathJax{\newcommand{\derradian}%
193         {\metre\usk\reciprocal\metre}}
194 \CustomizeMathJax{\newcommand{\dersteradian}%
195         {\squaremetre\usk\rpsquare\metre}}
196 \CustomizeMathJax{\newcommand{\derhertz}%
197         {\reciprocal\second}}
198 \CustomizeMathJax{\newcommand{\dernewton}%
199         {\metre\usk\kilogram\usk\second\rpsquared}}
200 \CustomizeMathJax{\newcommand{\derpascal}%
201         {\newton\usk\rpsquare\metre}}
202 \CustomizeMathJax{\newcommand{\derjoule}%
203         {\newton\usk\metre}}
204 \CustomizeMathJax{\newcommand{\derwatt}%
205         {\joule\usk\reciprocal\second}}
206 \CustomizeMathJax{\newcommand{\dercoulomb}%
207         {\ampere\usk\second}}
208 \CustomizeMathJax{\newcommand{\dervolt}%
209         {\watt\usk\reciprocal\ampere}}
210 \CustomizeMathJax{\newcommand{\derfarad}%
211         {\coulomb\usk\reciprocal\volt}}
212 \CustomizeMathJax{\newcommand{\derohm}%
213         {\volt\usk\reciprocal\ampere}}
214 \CustomizeMathJax{\newcommand{\dersiemens}%
215         {\ampere\usk\reciprocal\volt}}
216 \CustomizeMathJax{\newcommand{\derweber}%
217       {\squaremetre\usk\kilogram\usk\second\rpsquared\usk\reciprocal\ampere}}
218 \CustomizeMathJax{\newcommand{\dertesla}%
219         {\weber\usk\rpsquare\metre}}
220 \CustomizeMathJax{\newcommand{\derhenry}%
221         {\weber\usk\reciprocal\ampere}}
222 \CustomizeMathJax{\newcommand{\dercelsius}%
223         {\kelvin}}
224 \CustomizeMathJax{\newcommand{\derlumen}%
225         {\candela\usk\steradian}}
226 \CustomizeMathJax{\newcommand{\derlux}%
227         {\lumen\usk\rpsquare\metre}}
228 \CustomizeMathJax{\newcommand{\derbecquerel}%
229         {\derhertz}}
230 \CustomizeMathJax{\newcommand{\dergray}%
231         {\joule\usk\reciprocal\kilogram}}
232 \CustomizeMathJax{\newcommand{\dersievert}%
233         {\dergray}}
234 \CustomizeMathJax{\newcommand{\derkatal}%
235         {\katalbase}}
236 }{}
237
238 \CustomizeMathJax{\newcommand{\minute}{\mathrm{min}}}
239 \CustomizeMathJax{\newcommand{\hour}{\mathrm{h}}}
240 \CustomizeMathJax{\newcommand{\dday}{\mathrm{d}}}
241 \CustomizeMathJax{\newcommand{\degree}{\mathrm{^\circ}}}
242 \CustomizeMathJax{\newcommand{\paminute}{^\prime}}
243 \CustomizeMathJax{\newcommand{\arcminute}{^\prime}}
244 \CustomizeMathJax{\newcommand{\pasecond}{^{\prime\prime}}}
245 \CustomizeMathJax{\newcommand{\arcsecond}{^{\prime\prime}}}
```

```
246 \CustomizeMathJax{\newcommand{\ton}{\mathrm{t}}}
247 \CustomizeMathJax{\newcommand{\tonne}{\mathrm{t}}}
248 \CustomizeMathJax{\newcommand{\liter}{\mathrm{L}}}
249 \CustomizeMathJax{\newcommand{\litre}{\mathrm{l}}}
250 \CustomizeMathJax{\newcommand{\neper}{\mathrm{Np}}}
251 \CustomizeMathJax{\newcommand{\bel}{\mathrm{B}}}
252 \CustomizeMathJax{\newcommand{\curie}{\mathrm{Ci}}}
253 \CustomizeMathJax{\newcommand{\rad}{\mathrm{rad}}}
254 \CustomizeMathJax{\newcommand{\arad}{\mathrm{rd}}}
255 \CustomizeMathJax{\newcommand{\rem}{\mathrm{rem}}}
256 \CustomizeMathJax{\newcommand{\roentgen}{\mathrm{R}}}
257 \CustomizeMathJax{\newcommand{\electronvolt}{\mathrm{\mathrm{eV}}}}
258 \CustomizeMathJax{\newcommand{\atomicmass}{\mathrm{u}}}
259 \CustomizeMathJax{\newcommand{\atomicmassunit}{\mathrm{u}}}
260 \CustomizeMathJax{\newcommand{\dalton}{\mathrm{Da}}}
261 \CustomizeMathJax{\newcommand{\are}{\mathrm{a}}}
262 \CustomizeMathJax{\newcommand{\hectare}{\mathrm{\hecto\are}}}
263 \CustomizeMathJax{\newcommand{\barn}{\mathrm{b}}}
264 \CustomizeMathJax{\newcommand{\bbar}{\mathrm{bar}}}
265 \CustomizeMathJax{\newcommand{\gal}{\mathrm{Gal}}}
266 \CustomizeMathJax{\newcommand{\angstrom}{\mathrm{\unicode{x212B}}}}
267 \CustomizeMathJax{\newcommand{\rperminute}{\mathrm{r}\per\minute}}
268 \CustomizeMathJax{\newcommand{\rpersecond}{\mathrm{r}\per\second}}
269 \CustomizeMathJax{\newcommand{\squaremetre}{\power{\metre}{2}}}
270 \CustomizeMathJax{\newcommand{\cubicmetre}{\cubic\metre}}
271 \CustomizeMathJax{\newcommand{\graypersecond}{\gray\per\second}}
272 \CustomizeMathJax{\newcommand{\graypersecondnp}{\gray\usk\reciprocal\second}}
273 \CustomizeMathJax{\newcommand{\metrepersquaresecond}{\metre\per\second\squared}}
274 \CustomizeMathJax{\newcommand{\metrepersquaresecondnp}{\metre\usk\second\rpsquared}}
275 \CustomizeMathJax{\newcommand{\joulepermole}{\joule\per\mole}}
276 \CustomizeMathJax{\newcommand{\joulepermolenp}{\joule\usk\reciprocal\mole}}
277 \CustomizeMathJax{\newcommand{\molepercubicmetre}{\mole\per\cubic\metre}}
278 \CustomizeMathJax{\newcommand{\molepercubicmetrenp}{\mole\usk\rpcubic\metre}}
279 \CustomizeMathJax{\newcommand{\radianpersquaresecond}{\radian\per\second\squared}}
280 \CustomizeMathJax{\newcommand{\radianpersquaresecondnp}{\radian\usk\second\rpsquared}}
281 \CustomizeMathJax{\newcommand{\kilogramsquaremetrepersecond}{%
282     \kilogram\usk\squaremetre\per\second%
283 }}
284 \CustomizeMathJax{\newcommand{\kilogramsquaremetrepersecondnp}{%
285     \kilogram\usk\squaremetre\usk\reciprocal\second%
286 }}
287 \CustomizeMathJax{\newcommand{\radianpersecond}{\radian\per\second}}
288 \CustomizeMathJax{\newcommand{\radianpersecondnp}{\radian\usk\reciprocal\second}}
289 \CustomizeMathJax{\newcommand{\squaremetrepercubicmetre}{\squaremetre\per\cubic\metre}}
290 \CustomizeMathJax{\newcommand{\squaremetrepercubicmetrenp}{%
291     \squaremetre\usk\rpcubic\metre%
292 }}
293 \CustomizeMathJax{\newcommand{\katalpercubicmetre}{\katal\per\cubic\metre}}
294 \CustomizeMathJax{\newcommand{\katalpercubicmetrenp}{\katal\usk\rpcubic\metre}}
295 \CustomizeMathJax{\newcommand{\coulombpermol}{\coulomb\per\mole}}
296 \CustomizeMathJax{\newcommand{\coulombpermolnp}{\coulomb\usk\reciprocal\mole}}
297 \CustomizeMathJax{\newcommand{\amperepersquaremetre}{\ampere\per\squaremetre}}
298 \CustomizeMathJax{\newcommand{\amperepersquaremetrenp}{\ampere\usk\rpsquare\metre}}
299 \CustomizeMathJax{\newcommand{\kilogrampercubicmetre}{\kilogram\per\cubic\metre}}
300 \CustomizeMathJax{\newcommand{\kilogrampercubicmetrenp}{\kilogram\usk\rpcubic\metre}}
301 \CustomizeMathJax{\newcommand{\squaremetrepernewtonsecond}{%
302     \squaremetre\per\newton\usk\second%
303 }}
304 \CustomizeMathJax{\newcommand{\squaremetrepernewtonsecondnp}{%
305     \squaremetre\usk\reciprocal\newton\usk\reciprocal\second%
```

```
306 }}
307 \CustomizeMathJax{\newcommand{\pascalsecond}{\pascal\usk\second}}
308 \CustomizeMathJax{\newcommand{\coulombpercubicmetre}{\coulomb\per\cubic\metre}}
309 \CustomizeMathJax{\newcommand{\coulombpercubicmetrenp}{\coulomb\usk\rpcubic\metre}}
310 \CustomizeMathJax{\newcommand{\amperemetresecond}{\ampere\usk\metre\usk\second}}
311 \CustomizeMathJax{\newcommand{\voltpermetre}{\volt\per\metre}}
312 \CustomizeMathJax{\newcommand{\voltpermetrenp}{\volt\usk\reciprocal\metre}}
313 \CustomizeMathJax{\newcommand{\coulombpersquaremetre}{\coulomb\per\squaremetre}}
314 \CustomizeMathJax{\newcommand{\coulombpersquaremetrenp}{\coulomb\usk\rpsquare\metre}}
315 \CustomizeMathJax{\newcommand{\faradpermetre}{\farad\per\metre}}
316 \CustomizeMathJax{\newcommand{\faradpermetrenp}{\farad\usk\reciprocal\metre}}
317 \CustomizeMathJax{\newcommand{\ohmmetre}{\ohm\usk\metre}}
318 \CustomizeMathJax{\newcommand{\kilowatthour}{\kilo\watt\hour}}
319 \CustomizeMathJax{\newcommand{\wattpersquaremetre}{\watt\per\squaremetre}}
320 \CustomizeMathJax{\newcommand{\wattpersquaremetrenp}{\watt\usk\rpsquare\metre}}
321 \CustomizeMathJax{\newcommand{\joulepersquaremetre}{\joule\per\squaremetre}}
322 \CustomizeMathJax{\newcommand{\joulepersquaremetrenp}{\joule\usk\rpsquare\metre}}
323 \CustomizeMathJax{\newcommand{\newtonpercubicmetre}{\newton\per\cubic\metre}}
324 \CustomizeMathJax{\newcommand{\newtonpercubicmetrenp}{\newton\usk\rpcubic\metre}}
325 \CustomizeMathJax{\newcommand{\newtonperkilogram}{\newton\per\kilogram}}
326 \CustomizeMathJax{\newcommand{\newtonperkilogramnp}{\newton\usk\reciprocal\kilogram}}
327 \CustomizeMathJax{\newcommand{\jouleperkelvin}{\joule\per\kelvin}}
328 \CustomizeMathJax{\newcommand{\jouleperkelvinnp}{\joule\usk\reciprocal\kelvin}}
329 \CustomizeMathJax{\newcommand{\jouleperkilogram}{\joule\per\kilogram}}
330 \CustomizeMathJax{\newcommand{\jouleperkilogramnp}{\joule\usk\reciprocal\kilogram}}
331 \CustomizeMathJax{\newcommand{\coulombperkilogram}{\coulomb\per\kilogram}}
332 \CustomizeMathJax{\newcommand{\coulombperkilogramnp}{\coulomb\usk\reciprocal\kilogram}}
333 \CustomizeMathJax{\newcommand{\squaremetrepersecond}{\squaremetre\per\second}}
334 \CustomizeMathJax{\newcommand{\squaremetrepersecondnp}{%
335     \squaremetre\usk\reciprocal\second%
336 }}
337 \CustomizeMathJax{\newcommand{\squaremetrepersquaresecond}{%
338     \squaremetre\per\second\squared%
339 }}
340 \CustomizeMathJax{\newcommand{\squaremetrepersquaresecondnp}{%
341     \squaremetre\usk\second\rpsquared%
342 }}
343 \CustomizeMathJax{\newcommand{\kilogrammetrepersecond}{%
344     \kilogram\usk\metre\per\second%
345 }}
346 \CustomizeMathJax{\newcommand{\kilogrammetrepersecondnp}{%
347     \kilogram\usk\metre\usk\reciprocal\second%
348 }}
349 \CustomizeMathJax{\newcommand{\candelapersquaremetre}{\candela\per\squaremetre}}
350 \CustomizeMathJax{\newcommand{\candelapersquaremetrenp}{\candela\usk\rpsquare\metre}}
351 \CustomizeMathJax{\newcommand{\amperepermetre}{\ampere\per\metre}}
352 \CustomizeMathJax{\newcommand{\amperepermetrenp}{\ampere\usk\reciprocal\metre}}
353 \CustomizeMathJax{\newcommand{\joulepertesla}{\joule\per\tesla}}
354 \CustomizeMathJax{\newcommand{\jouleperteslanp}{\joule\usk\reciprocal\tesla}}
355 \CustomizeMathJax{\newcommand{\henrypermetre}{\henry\per\metre}}
356 \CustomizeMathJax{\newcommand{\henrypermetrenp}{\henry\usk\reciprocal\metre}}
357 \CustomizeMathJax{\newcommand{\kilogrampersecond}{\kilogram\per\second}}
358 \CustomizeMathJax{\newcommand{\kilogrampersecondnp}{\kilogram\usk\reciprocal\second}}
359 \CustomizeMathJax{\newcommand{\kilogrampersquaremetresecond}{%
360     \kilogram\per\squaremetre\usk\second%
361 }}
362 \CustomizeMathJax{\newcommand{\kilogrampersquaremetresecondnp}{%
363     \kilogram\usk\rpsquare\metre\usk\reciprocal\second%
364 }}
365 \CustomizeMathJax{\newcommand{\kilogrampersquaremetre}{\kilogram\per\squaremetre}}
```

366 `\CustomizeMathJax{\newcommand{\kilogrampersquaremetrenp}{\kilogram\usk\rpsquare\metre}}`
367 `\CustomizeMathJax{\newcommand{\kilogrampermetre}{\kilogram\per\metre}}`
368 `\CustomizeMathJax{\newcommand{\kilogrampermetrenp}{\kilogram\usk\reciprocal\metre}}`
369 `\CustomizeMathJax{\newcommand{\joulepermolekelvin}{\joule\per\mole\usk\kelvin}}`
370 `\CustomizeMathJax{\newcommand{\joulepermolekelvinnp}{%`
371 `    \joule\usk\reciprocal\mole\usk\reciprocal\kelvin%`
372 `}}`
373 `\CustomizeMathJax{\newcommand{\kilogramperkilomole}{\kilogram\per\kilo\mole}}`
374 `\CustomizeMathJax{\newcommand{\kilogramperkilomolenp}{%`
375 `    \kilogram\usk\kilo\reciprocal\mole%`
376 `}}`
377 `\CustomizeMathJax{\newcommand{\kilogramsquaremetre}{\kilogram\usk\squaremetre}}`
378 `\CustomizeMathJax{\newcommand{\kilogramsquaremetrenp}{\kilogramsquaremetre}}`
379 `\CustomizeMathJax{\newcommand{\kilogrammetrepersquaresecond}{%`
380 `    \kilogram\usk\metre\per\second\squared%`
381 `}}`
382 `\CustomizeMathJax{\newcommand{\kilogrammetrepersquaresecondnp}{%`
383 `    \kilogram\usk\metre\usk\second\rpsquared%`
384 `}}`
385 `\CustomizeMathJax{\newcommand{\newtonpersquaremetre}{\newton\per\squaremetre}}`
386 `\CustomizeMathJax{\newcommand{\newtonpersquaremetrenp}{\newton\usk\rpsquare\metre}}`
387 `\CustomizeMathJax{\newcommand{\persquaremetresecond}{1\per\squaremetre\usk\second}}`
388 `\CustomizeMathJax{\newcommand{\persquaremetresecondnp}{%`
389 `    \rpsquare\metre\usk\reciprocal\second%`
390 `}}`
391 `\CustomizeMathJax{\newcommand{\wattperkilogram}{\watt\per\kilogram}}`
392 `\CustomizeMathJax{\newcommand{\wattperkilogramnp}{\watt\usk\reciprocal\kilogram}}`
393 `\CustomizeMathJax{\newcommand{\wattpercubicmetre}{\watt\per\cubic\metre}}`
394 `\CustomizeMathJax{\newcommand{\wattpercubicmetrenp}{\watt\usk\rpcubic\metre}}`
395 `\CustomizeMathJax{\newcommand{\wattpersquaremetresteradian}{%`
396 `    \watt\per\squaremetre\usk\steradian%`
397 `}}`
398 `\CustomizeMathJax{\newcommand{\wattpersquaremetresteradiannp}{%`
399 `    \watt\usk\rpsquare\metre\usk\rp\steradian%`
400 `}}`
401 `\CustomizeMathJax{\newcommand{\jouleperkilogramkelvin}{\joule\per\kilogram\usk\kelvin}}`
402 `\CustomizeMathJax{\newcommand{\jouleperkilogramkelvinnp}{%`
403 `    \joule\usk\reciprocal\kilogram\usk\reciprocal\kelvin%`
404 `}}`
405 `\CustomizeMathJax{\newcommand{\squaremetreperkilogram}{\squaremetre\per\kilogram}}`
406 `\CustomizeMathJax{\newcommand{\rpsquaremetreperkilogram}{%`
407 `    \squaremetre\usk\reciprocal\kilogram%`
408 `}}`
409 `\CustomizeMathJax{\newcommand{\cubicmetreperkilogram}{\cubic\metre\per\kilogram}}`
410 `\CustomizeMathJax{\newcommand{\rpcubicmetreperkilogram}{%`
411 `    \cubic\metre\usk\reciprocal\kilogram%`
412 `}}`
413 `\CustomizeMathJax{\newcommand{\newtonpermetre}{\newton\per\metre}}`
414 `\CustomizeMathJax{\newcommand{\newtonpermetrenp}{\newton\usk\reciprocal\metre}}`
415 `\CustomizeMathJax{\newcommand{\Celsius}{\unicode{x2103}}}`
416 `\CustomizeMathJax{\newcommand{\wattpermetrekelvin}{\watt\per\metre\usk\kelvin}}`
417 `\CustomizeMathJax{\newcommand{\wattpermetrekelvinnp}{%`
418 `    \watt\usk\reciprocal\metre\usk\reciprocal\kelvin%`
419 `}}`
420 `\CustomizeMathJax{\newcommand{\newtonmetre}{\newton\usk\metre}`
421 `\CustomizeMathJax{\newcommand{\newtonmetrenp}{\newtonmetre}}}`
422 `\CustomizeMathJax{\newcommand{\squaremetrepercubicsecond}{%`
423 `    \squaremetre\per\cubic\second%`
424 `}}`
425 `\CustomizeMathJax{\newcommand{\squaremetrepercubicsecondnp}{%`

```
426        \squaremetre\usk\rpcubic\second%
427 }}
428 \CustomizeMathJax{\newcommand{\metrepersecond}{\metre\per\second}}
429 \CustomizeMathJax{\newcommand{\metrepersecondnp}{\metre\usk\reciprocal\second}}
430 \CustomizeMathJax{\newcommand{\joulepercubicmetre}{\joule\per\cubicmetre}}
431 \CustomizeMathJax{\newcommand{\joulepercubicmetrenp}{\joule\usk\rpcubic\metre}}
432 \CustomizeMathJax{\newcommand{\kilogrampercubicmetrecoulomb}{%
433        \kilogram\per\cubic\metre\usk\coulomb%
434 }}
435 \CustomizeMathJax{\newcommand{\kilogrampercubicmetrecoulombnp}{%
436        \kilogram\usk\rpcubic\metre\usk\reciprocal\coulomb%
437 }}
438 \CustomizeMathJax{\newcommand{\cubicmetrepersecond}{\cubicmetre\per\second}}
439 \CustomizeMathJax{\newcommand{\rpcubicmetrepersecond}{\cubicmetre\usk\reciprocal\second}}
440 \CustomizeMathJax{\newcommand{\kilogrampersecondcubicmetre}{%
441        \kilogram\per\second\usk\cubicmetre%
442 }}
443 \CustomizeMathJax{\newcommand{\kilogrampersecondcubicmetrenp}{%
444        \kilogram\usk\reciprocal\second\usk\rpcubic\metre%
445 }}
446 \end{warpMathJax}
```

File 456   **`lwarp-siunitx.sty`**

§ 568   Package   **siunitx**

(*Emulates or patches code by* JOSEPH WRIGHT.)

siunitx (*Pkg*)   siunitx is patched for use by lwarp, and is emulated for MATHJAX.

for HTML output:
```
1 \providecommand\DeclareRelease[3]{}
2 \providecommand\DeclareCurrentRelease[2]{}
3
4 \DeclareRelease{2}{2010-05-23}{lwarp-siunitx-v2.sty}
5 \DeclareRelease{v2}{2010-05-23}{lwarp-siunitx-v2.sty}
6 \DeclareCurrentRelease{}{2021-05-17}
7
8 \RequirePackage{xcolor}% for \convertcolorspec
9
10 \LWR@ProvidesPackagePass{siunitx}[2025-01-21]
11
12 \ExplSyntaxOn
13
14 \VerifyCommand[lwarp][siunitx]{\siunitx_number_format:nN}{76C110239433A6FC6441704392C40200}
15
16 \cs_set_protected:Npn \siunitx_number_format:nN #1#2
17   {
18     \group_begin:
19       \bool_if:NTF \l_siunitx_number_parse_bool
20         {
21           \siunitx_number_parse:nN {#1} \l__siunitx_number_parsed_tl
22           \tl_if_empty:NF \l__siunitx_number_parsed_tl
23             {
24             \siunitx_number_process:NN \l__siunitx_number_parsed_tl \l__siunitx_number_parsed_tl
25                \tl_set:Nx \l__siunitx_number_outputted_tl
26                  { \siunitx_number_output:N \l__siunitx_number_parsed_tl }
27             }
```

```
27            }
28            {
29                 \tl_set:Nn \l__siunitx_number_outputted_tl
30                 {
31                     \boolfalse{mathjax}%                        lwarp
32                     \LWR@subsingledollar{%                      lwarp
33                         \textbackslash( % space                lwarp ALT text
34                         \LWR@HTMLsanitizedetokenized{%
35                             \detokenize{#1}%
36                         } \textbackslash)%                      lwarp
37                     }%
38                     {siunitx unparsed}%                         lwarp add'l hashing
39                     {\ensuremath{#1}}%                          lwarp
40                 }
41            }
42        \exp_args:NNNV \group_end:
43        \tl_set:Nn #2 \l__siunitx_number_outputted_tl
44    }

45 \VerifyCommand[lwarp][siunitx]{\__siunitx_compound_unparsed:n}{C6CACB29BD6BD43225E174AD2FBD750C}
46
47 \cs_set_protected:Npn \__siunitx_compound_unparsed:n #1
48    {
49        \tl_if_blank:nF {#1}
50            { \seq_put_right:Nn \l__siunitx_compound_tmp_seq
51                {
52                    \boolfalse{mathjax}%                    lwarp
53                    \LWR@subsingledollar{%                  lwarp
54                        \textbackslash( % space            lwarp ALT tag
55                        \LWR@HTMLsanitizedetokenized{%
56                            \detokenize{#1}%
57                        } \textbackslash)%                  lwarp
58                    }%
59                    {siunitx unparsed}%                     lwarp add'l hashing
60                    {\ensuremath{#1}}%                      lwarp
61                }
62            }
63    }
```

\LWR@siunitx@mathrm   {⟨*text*⟩}

If in text mode, use \textrm instead. Avoids crashing while using \mathrm in text mode.

```
64 \LetLtxMacro\LWR@siunitx@orig@mathrm\mathrm
65
66 \newcommand*{\LWR@siunitx@mathrm}[1]{%
67    \ifmmode{\LWR@siunitx@orig@mathrm{#1}}\else{#1}\fi%
68 }
```

If not in a lateximage, always use text mode. Ignore current text font if resetting text family, series, and shape.

```
69 \VerifyCommand[lwarp][siunitx]{\__siunitx_print_aux:nn}{4D81B79284057560BC61DF11EF723AC8}
70
71 \cs_set_protected:Npn \__siunitx_print_aux:nn #1#2
72    {
73        \LetLtxMacro\mathrm\LWR@siunitx@mathrm%                     lwarp
74        \tl_if_empty:oF {#2}
75            {
```

```
76        \tl_if_empty:cTF { l__siunitx_print_ #1 _color_tl }
77          { \use:n }
78         { \ExpandArgs { v } \textcolor { l__siunitx_print_ #1 _color_tl } }
79            {
80                \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%        lwarp
81                {
82                    \use:c
83                      {
84                           __siunitx_print _ #1 :n
85                      }
86                    {#2}
87                }
88                {
89                    \bool_lazy_all:nTF%      lwarp
90                      {
91                          {\l__siunitx_print_text_family_bool}
92                          {\l__siunitx_print_text_series_bool}
93                          {\l__siunitx_print_text_shape_bool}
94                      }
95                    {% No font control if reset-text-family/series/shape
96                        \use:c
97                          {
98                          siunitx_print_%                            lwarp
99                             text%                                   lwarp
100                             :n%                                    lwarp
101                          }%                                        lwarp
102                          {#2}%                                     lwarp
103                    }
104                    {
105                        \LWR@textcurrentfont{%                      lwarp
106                          \use:c
107                            {
108                            siunitx_print_%                         lwarp
109                               text%                                lwarp
110                               :n%                                  lwarp
111                            }%                                      lwarp
112                            {#2}%                                   lwarp
113                        }
114                    }
115                }
116            }
117    }
118 }
```

To determine whether to make a complex root be italic or upright, \l__siunitx_complex_output_root_tl
is compared to \LWR@siunitx@complexrm<i/j>, and the css style for i,j is set to
ijit or ijup.

```
119 \newcommand*{\LWR@siunitx@complexrootstyle}{textrm}
120
121 \newcommand*{\LWR@siunitx@complexrmi}{\mathrm{i}}
122 \newcommand*{\LWR@siunitx@complexrmj}{\mathrm{j}}
123
124 \newcommand*{\LWR@siunitx@setcomplexroot}{%
125    \renewcommand*{\LWR@siunitx@complexrootstyle}{ijit}%
126    \ifdefequal{\l__siunitx_complex_output_root_tl}{\LWR@siunitx@complexrmi}%
127        {\renewcommand*{\LWR@siunitx@complexrootstyle}{ijup}}%
128        {}%
129    \ifdefequal{\l__siunitx_complex_output_root_tl}{\LWR@siunitx@complexrmj}%
130        {\renewcommand*{\LWR@siunitx@complexrootstyle}{ijup}}%
```

```
131          {}%
132 }

133 \VerifyCommand[lwarp][siunitx]{\__siunitx_complex_format_cartesian_auxii:n}
134     {DC0897DBE172C13B6F7282D266BE1156}

135
136 \cs_set_protected:Npn \__siunitx_complex_format_cartesian_auxii:n #1
137   {
138       \LWR@siunitx@setcomplexroot%                               lwarp
139       \__siunitx_complex_format_cartesian_units:n {#1}
140       \tl_if_empty:NF \l__siunitx_complex_real_tl
141       { \exp_after:wN \__siunitx_complex_drop_exponent:nnnnnnn \l__siunitx_complex_real_tl }
142     \exp_after:wN \__siunitx_complex_format_sign:nnnnnnn \l__siunitx_complex_img_tl
143     \tl_set:Nx \l__siunitx_complex_tmp_tl
144       { \siunitx_number_output:NN \l__siunitx_complex_img_tl \q_nil }
145     \exp_after:wN \__siunitx_complex_extract_exponent:w \l__siunitx_complex_tmp_tl \q_stop
146     \tl_set:Nx \l__siunitx_complex_tmp_tl
147       {
148           \bool_lazy_or:nnTF
149             {
150               \bool_lazy_and_p:nn
151                 { \l_siunitx_number_bracket_ambiguous_bool }
152                 { ! \tl_if_empty_p:N \l__siunitx_complex_exp_tl }
153             }
154             {
155               ! \bool_lazy_any_p:n
156                 {
157                   { \tl_if_blank_p:n {#1} }
158                   { \tl_if_empty_p:N \l__siunitx_complex_real_tl }
159                   { \tl_if_empty_p:N \l__siunitx_complex_img_tl }
160                 }
161             }
162           { \__siunitx_complex_format_bracket:n }
163           { \use:n }
164           {
165             \siunitx_number_output:N \l__siunitx_complex_real_tl
166             \exp_not:V \l__siunitx_complex_sign_tl
167             \bool_if:NF \l__siunitx_complex_root_after_bool
168               {
169                 \InlineClass{\LWR@siunitx@complexrootstyle}%     lwarp
170                   {
171                       \exp_not:V \l__siunitx_complex_output_root_tl
172                   }
173               }
174             \exp_not:V \l__siunitx_complex_tmp_tl
175             \bool_if:NT \l__siunitx_complex_root_after_bool
176               {
177                 \InlineClass{\LWR@siunitx@complexrootstyle}%     lwarp
178                   {
179                       \exp_not:V \l__siunitx_complex_output_root_tl
180                   }
181               }
182           }
183         \exp_not:V \l__siunitx_complex_exp_tl
184     }
185   }
```

{⟨*1: decimal part of angle*⟩} {⟨*2: ?*⟩} {⟨*3: integer part of angle*⟩} {⟨*4: decimal point char*⟩} {⟨*5: deg/min/sec char*⟩}

```
186 \VerifyCommand[lwarp][siunitx]{\__siunitx_angle_arc_print_auxiv:nnnnn}%
187     {BA9FCDF6D4868323669F0F8C61412508}
188 \cs_set_protected:Npn \__siunitx_angle_arc_print_auxiv:nnnnn #1#2#3#4#5
189   {
190     \mode_if_math:TF
191       { \bool_set_true:N \l__siunitx_angle_tmp_bool }
192       { \bool_set_false:N \l__siunitx_angle_tmp_bool }
193     \siunitx_print_number:n {#3}
194     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%          lwarp
195     {%                                                      lwarp: lateximage
196     \tl_if_blank:nTF {#1}% decimal part of angle
197       {
198         \siunitx_print_number:n {#2}%
199         \__siunitx_angle_arc_print_auxix:n {#5}% deg/min/sec char
200       }
201       {
202         \hbox_set:Nn \l__siunitx_angle_marker_box
203           {
204             \__siunitx_angle_arc_print_auxv:n
205               { \siunitx_print_number:n {#4} }% decimal point char
206           }
207         \hbox_set:Nn \l__siunitx_angle_unit_box
208           {
209             \__siunitx_angle_arc_print_auxv:n
210               {
211             \siunitx_unit_format:nN {#5} \l__siunitx_angle_tmp_tl% deg/min/sec char
212               \siunitx_print_unit:V \l__siunitx_angle_tmp_tl
213               \skip_horizontal:n { -\scriptspace }
214               }
215           }
216         \dim_compare:nNnTF { \box_wd:N \l__siunitx_angle_marker_box } >
217             { \box_wd:N \l__siunitx_angle_unit_box }
218             {
219               \__siunitx_angle_arc_print_auxvi:NN
220                 \l__siunitx_angle_marker_box
221                 \l__siunitx_angle_unit_box
222             }
223             {
224               \__siunitx_angle_arc_print_auxvi:NN
225                 \l__siunitx_angle_unit_box
226                 \l__siunitx_angle_marker_box
227             }
228         \hbox_set_to_wd:Nnn \l__siunitx_angle_marker_box
229             \l__siunitx_angle_tmp_dim
230             {
231               \hbox_overlap_right:n
232                 { \box_use_drop:N \l__siunitx_angle_marker_box }
233               \hbox_overlap_right:n
234                 { \box_use_drop:N \l__siunitx_angle_unit_box }
235               \tex_hfil:D
236             }
237         \box_use:N \l__siunitx_angle_marker_box
238         \skip_horizontal:N \scriptspace
239         \siunitx_print_number:n {#1#2}% decimal part of angle
240       }
241       }% lwarp: latex image
```

{⟨*1: decimal part of angle*⟩} {⟨*2: ?*⟩} {⟨*3: integer part of angle*⟩} {⟨*4: decimal point char*⟩} {⟨*5: deg/min/sec char*⟩}

```
242        {% lwarp: not latex iamge
243      \tl_if_blank:nTF {#1}% decimal part of angle
244        {
245          \siunitx_print_number:n {#2}%
246          \__siunitx_angle_arc_print_auxix:n {#5}% deg/min/sec char
247        }
248        {
249 %        \hbox_set:Nn \l__siunitx_angle_marker_box
250 %          {
251              \__siunitx_angle_arc_print_auxv:n
252                { \siunitx_print_number:n {#4} }% decimal point char
253 %          }
254 %        \hbox_set:Nn \l__siunitx_angle_unit_box
255 %          {
256              \__siunitx_angle_arc_print_auxv:n
257                {
258          \siunitx_unit_format:nN {#5} \l__siunitx_angle_tmp_tl% deg/min/sec char
259                \siunitx_print_unit:V \l__siunitx_angle_tmp_tl
260 %              \skip_horizontal:n { -\scriptspace }
261                }
262 %          }
263 %        \dim_compare:nNnTF { \box_wd:N \l__siunitx_angle_marker_box } >
264 %          { \box_wd:N \l__siunitx_angle_unit_box }
265 %          {
266 %            \__siunitx_angle_arc_print_auxvi:NN
267 %              \l__siunitx_angle_marker_box
268 %              \l__siunitx_angle_unit_box
269 %          }
270 %          {
271 %            \__siunitx_angle_arc_print_auxvi:NN
272 %              \l__siunitx_angle_unit_box
273 %              \l__siunitx_angle_marker_box
274 %          }
275 %        \hbox_set_to_wd:Nnn \l__siunitx_angle_marker_box
276 %          \l__siunitx_angle_tmp_dim
277 %          {
278 %            \hbox_overlap_right:n
279 %              { \box_use_drop:N \l__siunitx_angle_marker_box }
280 %            \hbox_overlap_right:n
281 %              { \box_use_drop:N \l__siunitx_angle_unit_box }
282 %            \tex_hfil:D
283 %          }
284 %        \box_use:N \l__siunitx_angle_marker_box
285 %        \skip_horizontal:N \scriptspace
286          \siunitx_print_number:n {#1#2}% decimal part of angle
287        }
288      }% lwarp: not latex image
289    }
```

```
290 \VerifyCommand[lwarp][siunitx]{\__siunitx_angle_arc_print_auxix:n}
291    {603807F80B4E40084CC07396DBC89FC1}
292
293 \cs_set_protected:Npn \__siunitx_angle_arc_print_auxix:n #1
294   {
295     \group_begin:
296       \siunitx_unit_options_apply:n {#1}
```

```
297        \keys_set:nn { siunitx }{quantity-product={}}%      lwarp
298        \siunitx_unit_format:nN {#1} \l__siunitx_angle_tmp_tl
299        \siunitx_quantity_print:nV { } \l__siunitx_angle_tmp_tl
300      \group_end:
301    }
```

If not in a lateximage, print a simple inline fraction, avoiding the use of SVG math:

```
302 \VerifyCommand[lwarp][siunitx]{\__siunitx_print_text_fraction:Nnn}
303     {F47521F256C661719258012969E7AE04}
304
305 \cs_set_protected:Npn \__siunitx_print_text_fraction:Nnn #1#2#3
306   {
307     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%  lwarp
308     {%
309       \ensuremath
310       {
311         #1
312           { \mbox { \__siunitx_print_text_replace:n {#2} } }
313           { \mbox { \__siunitx_print_text_replace:n {#3} } }
314       }
315     }%
316     {%                                                         lwarp
317         { \mbox { \__siunitx_print_text_replace:n {#2} } }%   lwarp
318         /%                                                     lwarp
319         { \mbox { \__siunitx_print_text_replace:n {#3} } }%   lwarp
320     }%      lwarp
321   }
```

If not in a lateximage, print a \textsubscript:

```
322 \VerifyCommand[lwarp][siunitx]{\__siunitx_unit_format_qualifier_subscript:}
323     {543B01848C00E4089F0E0C53988F6A28}
324
325 \cs_set_protected:Npn \__siunitx_unit_format_qualifier_subscript:
326   {
327     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%        lwarp
328     {%
329       \__siunitx_unit_format_font:
330       \tl_set:Nx \l__siunitx_unit_part_tl
331       {
332           \c__siunitx_unit_math_subscript_tl
333           {
334               \exp_not:V \l_siunitx_unit_font_tl
335               { \exp_not:V \l__siunitx_unit_part_tl }
336           }
337       }
338     }
339     {%      lwarp simplified for HTML:
340       \__siunitx_unit_format_font:
341       \tl_set:Nx \l__siunitx_unit_part_tl
342       {
343           \textsubscript
344           {
345               \exp_not:V \l_siunitx_unit_font_tl
346               { \exp_not:V \l__siunitx_unit_part_tl }
347           }
348       }
349     }
350   }
```

```
351 \VerifyCommand[lwarp][siunitx]{\siunitx_quantity:nn}
352     {AEF3237DB5107FE46437AF1D3ABD03DE}
353
354 \cs_set_protected:Npn \siunitx_quantity:nn #1#2
355   {
356     \group_begin:
357       \siunitx_unit_options_apply:n {#2}
358       \tl_if_blank:nTF {#1}
359         {
360           \siunitx_unit_format:nN {#2} \l__siunitx_quantity_unit_tl
361           \siunitx_print_unit:V \l__siunitx_quantity_unit_tl
362         }
363         {
364           \bool_if:NTF \l_siunitx_number_parse_bool
365             { \__siunitx_quantity_parsed:nn {#1} {#2} }
366             {
367               \tl_set:Nn \l__siunitx_quantity_number_tl {
368                 \boolfalse{mathjax}%                      lwarp
369                 \LWR@subsingledollar{%                    lwarp
370                   \textbackslash( % space                lwaro ALT tag
371                   \LWR@HTMLsanitizedetokenized{%
372                     \detokenize{#1}%
373                   } \textbackslash)%                      lwarp
374                 }%
375                 {siunitx unparsed}%                       lwarp add'l hashing
376                 {\ensuremath{#1}}%                        lwarp
377               }
378               \siunitx_unit_format:nN {#2} \l__siunitx_quantity_unit_tl
379               \siunitx_quantity_print:VV
380                 \l__siunitx_quantity_number_tl \l__siunitx_quantity_unit_tl
381             }
382         }
383     \group_end:
384   }
```

\cancel for HTML does not work yet.

```
385 \newcommand*{\LWR@siunitx@nocancel}[1]{%
386   \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
387     {\cancel{#1}}%  SVG
388     {#1}%           HTML
389 }
390
391 \AtBeginDocument{
392 \__siunitx_unit_set_symbolic:Npnn \cancel
393   { }
394 %   { \__siunitx_unit_parse_special:n { \cancel } }
395   { \__siunitx_unit_parse_special:n { \LWR@siunitx@nocancel } }%    lwarp
396 }
```

For HTML, use a simple unaligned \num:

```
397 \newcommand{\LWR@HTML@tablenum}[2][]{\num[#1]{#2}}
398 \LWR@formatted{tablenum}
```

For HTML, the S column is simplified to a c column. Keys are set locally, allowing drop-exponent, etc.

```
399 \AtBeginDocument{
```

```
400 \HTMLnewcolumntype{S}[1][]{>{\begingroup\sisetup{#1}}c<{\endgroup}}
401 }
```

To define simplified units for HTML:

\HTMLDeclareSIUnit [⟨*options*⟩] {⟨*name*⟩} {⟨*definition*⟩}

```
402 \NewDocumentCommand{\HTMLDeclareSIUnit}{o +m m}
403 {
404     \ifcsdef{ __siunitx_unit_ \token_to_str:N #2 :w }
405         {}
406         {
407             \PackageError{lwarp}
408                 {%
409                     First~use\MessageBreak
410                         \space\space\protect\DeclareSIUnit{
411                             \token_to_str:N#2}{...}\MessageBreak
412                     before~using\MessageBreak
413                         \space\space\protect\HTMLDeclareSIUnit{
414                             \token_to_str:N#2}{...}%
415                 }
416                 {%
417                     See~the~Lwarp~manual~section~about~special~cases,~
418                     regarding~siunitx.%
419                 }
420         }
421     \csNewCommandCopycs
422         { __orig_siunitx_unit_ \token_to_str:N #2 :w }
423         { __siunitx_unit_ \token_to_str:N #2 :w }
424     \DeclareSIUnit[#1]{#2}
425         {
426             \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}
427                 {\csuse{ __orig_siunitx_unit_ \token_to_str:N #2 :w }}
428                 {#3}
429         }
430 }
```

```
431 \ExplSyntaxOff
```

HTML versions for existing units:

```
432 \AtBeginDocument{
433 \HTMLDeclareSIUnit\celsius{\LWR@siunitx@textcelsius}
434 \HTMLDeclareSIUnit\arcminute{\LWR@siunitx@textprime}
435 \HTMLDeclareSIUnit\arcsecond{\LWR@siunitx@textdblprime}
436 \HTMLDeclareSIUnit\elementarycharge{\textit{e}}
437 %
438 \HTMLDeclareSIUnit\clight{\text{\textit{c}\textsubscript{0}}}
439 \HTMLDeclareSIUnit\bohr{\text{\textit{a}\textsubscript{0}}}
440 \HTMLDeclareSIUnit\electronmass{\text{\textit{m}\textsubscript{e}}}
441 \HTMLDeclareSIUnit\hartree{\text{\textit{E}\textsubscript{h}}}
442 \HTMLDeclareSIUnit\planckbar{\LWR@siunitx@textplanckbar}
443 }% \AtBeginDocument
```

Initial options:

```
444 \AtBeginDocument{
445 \sisetup{
```

```
446    per-mode=symbol,      % fraction is not seen by pdftotext
447    angle-symbol-degree = {\LWR@siunitx@textdegree},
448    angle-symbol-minute = {\LWR@siunitx@textprime} ,
449    angle-symbol-second = {\LWR@siunitx@textdblprime} ,
450 }
451 }
```

Load late paches for lltjp-siunitx:

```
452 \AtBeginDocument{
453 \ifdef{\ltj@allalchar}
454    {\LWR@origRequirePackage{lwarp-lltjp-siunitx}}
455    {}
456 }
```

For MATHJAX:

```
457 \LWR@origRequirePackage{lwarp-common-mathjax-siunitx}
458
459 \begin{warpMathJax}
460 \CustomizeMathJax{\let\unit\si}
461 \CustomizeMathJax{\let\qty\SI}
462 \CustomizeMathJax{\let\qtylist\SIlist}
463 \CustomizeMathJax{\let\qtyrange\SIrange}
464 \CustomizeMathJax{\let\numproduct\num}
465 \CustomizeMathJax{\let\qtyproduct\SI}
466 \CustomizeMathJax{\let\complexnum\num}
467 \CustomizeMathJax{\newcommand{\complexqty}[3][]{(\complexnum{#2})\si{#3}}}
468 \end{warpMathJax}
```

Pass range-phrase to common-mathjax-siunitx:

```
469 \ExplSyntaxOn
470 \AtBeginDocument{
471 \edef\LWR@siunitx@rangephrase{\l_siunitx_range_phrase_tl}
472 \expandafter\CustomizeMathJax\expandafter{%
473    \expandafter\def\expandafter\LWRsiunitxrangephrase%
474    \expandafter{\LWR@siunitx@rangephrase}%
475 }
476 }
477 \ExplSyntaxOff
```

File 457   **`lwarp-siunitx-v2.sty`**

§ 569   Package   # siunitx-v2

(*Emulates or patches code by* JOSEPH WRIGHT.)

siunitx-v2 (*Pkg*)   siunitx-v2 is patched for use by lwarp, and is emulated for MATHJAX.

siunitx is well supported by lwarp.

Limitations   Some general limitations:

fractions   Due to *pdftotext* limitations, fraction output is replaced by symbol output for
per-mode and quotient-mode.

\cancel is not currently supported for siunitx v3.

Negative values are not automatically colored.

⚠   tabular   Tabular S and s columns are rendered as simple c columns, although key settings
will be set. If using scientific notation, table-format, table-align-uncertainty,
drop-exponent, etc.. use \tablenum for each cell. This is especially required for
⚠   drop-exponent   drop-exponent, without which the value will be shown incorrectly.

⚠   table-auto-round   table-auto-round is ignored.

Math rendering   Math may be rendered in several ways in the same document:

For math mode with svg display: The original siunitx code is used while gener-
ating the svg image.

For html text mode: lwarp uses siunitx code patched for html, and simplified
units.

For math expressions while using MATHJAX: A limited emulation is used. Most
functions work reasonably well, but many options cannot be emulated. The
result usually looks fine, and otherwise is enough to get the meaning across.

Custom units   siunitx allows customized units:

\DeclareSIUnit   {⟨*name*⟩} {⟨*definition*⟩}

\DeclareSIUnit declares a version of the unit for the print version. This is also
used when the unit is printed in svg math or a lateximage. It is also used for
html if an html-specific version is not defined with \HTMLDeclareSIUnit.

    \DeclareSIUnit\myunit{\ensuremath{\text{m}_y}}

\HTMLDeclareSIUnit   {⟨*name*⟩} {⟨*definition*⟩}

⚠   v3 only!   Use this after the print unit has been defined. For siunitx v3, \HTMLDeclareSIUnit
declares a simplified version of the unit for html, for example if the print-mode
unit uses TEX boxes or \ensuremath:

    \HTMLDeclareSIUnit\myunit{\text{m}\textsubscript{\textit{y}}}

It is also possible to provide a custom unit for MATHJAX:

    \CustomizeMathJax{\newcommand{\myunit}{\text{m}_y}}

Predefined units Most units work as-is with HTML. For the following units, lwarp has already set \HTMLDeclareSIUnit: \celsius, \arcminute, \arcsecond, \elementarycharge, \clight, \bohr, \electronmass, \hartree, \planckbar.

⚠ MathJax

Document modifications required for MATHJAX

⚠ \sisetup
- Place \sisetup in the preamble before \begin{document}. Changes made later may be ignored, especially with MATHJAX. The MATHJAX emulation also ignores most macro options.

⚠ complex numbers
- Complex numbers are displayed as entered, ignoring output-complex-root.

custom units
- Custom units may be added with \CustomizeMathJax. For example, from lwarp-common-mathjax-siunitx:

      \CustomizeMathJax{\newcommand{\hartree}{\mathit{E}_{\mathrm{h}}}}
      \CustomizeMathJax{\newcommand{\angstrom}{\mathrm{\unicode{x212B}}}}

⚠ unit spacing
- Units work better using ~ between units instead of using periods.

⚠ \square, \cubic
- To square or cube compound units, enclose the following compound units in braces:

      \cubic{\centi\meter}

  Single units do not require braces.

- For \numlist, the argument is printed as text as-is, so use space between semicolons for improved readability.

⚠ Missing $ inserted
- If using parse-numbers = false, also use \num or \qty. siunitx=siunitx>Missing $ inserted.

Also see MATHJAX option, section 8.7.5.

for HTML output:
```
1 \RequirePackage{xcolor}% for \convertcolorspec
2
3 \LWR@ProvidesPackagePass{siunitx}[=v2]% 2021-04-17
```

```
4 \AtBeginDocument{% in case textcomp was not loaded
5     \DeclareSIUnit\bohr{\textit{a}\textsubscript{0}}
6     \DeclareSIUnit\clight{\textit{c}\textsubscript{0}}
7     \DeclareSIUnit\elementarycharge{\textit{e}}
8     \DeclareSIUnit\electronmass{\textit{m}\textsubscript{e}}
9     \DeclareSIUnit\hartree{\textit{E}\textsubscript{h}}
10    \DeclareSIUnit\planckbar{\LWR@siunitx@textplanckbar}
11 }% AtBeginDocument
```

Support the S and s column types:

```
12 \AtBeginDocument{
13 \HTMLnewcolumntype{S}[1][]{>{\begingroup\sisetup{#1}}c<{\endgroup}}
14 \HTMLnewcolumntype{s}[1][]{>{\begingroup\sisetup{#1}}c<{\endgroup}}
15 }
```

\@ensuredmath is not supported inside an \hbox, so it must temporarily be restored to its original. Similar for \mbox. SVG math is created explicitly when necessary, using \LWR@subsingledollar.

```
16
17 \ExplSyntaxOn
18 %
```

Modified to use the print version of \@ensuredmath to avoid having a lateximage each time.

```
19 \AtBeginDocument{
20 \cs_set_protected:Npn \__siunitx_print_text:
21   {
22     \LetLtxMacro\@ensuredmath\LWR@origensuredmath%          lwarp
23     \tl_replace_all:Nnn \l__siunitx_print_arg_tl { - }
24       { \textminus }
25     \__siunitx_print_text_aux:
26     \tl_replace_all:Nnn \l__siunitx_print_arg_tl { \mp }
27       { \ensuremath { \mp } }
28     \tl_remove_all:Nn \l__siunitx_print_arg_tl { \mathord }
29     \cs_set_eq:NN \PrintSubscript   \__siunitx_print_text_sub:n
30     \cs_set_eq:NN \PrintSuperscript \__siunitx_print_text_super:n
31     \__siunitx_print_text_aux:NnN
32       _ { math_subscript }   \__siunitx_print_text_sub:n
33       _ { active }           \__siunitx_print_text_sub:n
34       ^ { math_superscript } \__siunitx_print_text_super:n
35       ^ { active }           \__siunitx_print_text_super:n
36     \q_recursion_tail ?  ?
37     \q_recursion_stop
38     \l__siunitx_print_arg_tl
39   }
40 }
```

Modified to set set HTML \textcolor if not black:

```
41 \cs_new_protected:Npn \LWR@HTML@__siunitx_print_aux:
42   {
43     \text
44       {
45         \__siunitx_ensure_ltr:n
46           {
47             \color@begingroup
48 %
49             \__siunitx_print_color:
50             \__siunitx_font_shape:
51             \__siunitx_font_weight:
52             \use:c
53               {
54                 __siunitx_ \l__siunitx_print_type_tl _
55                 text \l__siunitx_font_family_tl :
56               }
57 %             \bool_if:NTF \l__siunitx_font_math_mode_bool
58 %               {
59 %                 \__siunitx_print_math:
60 %               }
61               {
62                 \LWR@findcurrenttextcolor% lwarp
63                 \ifdefstring{\LWR@tempcolor}{000000}% lwarp
64                   {\__siunitx_print_text:}% lwarp
65                   {% lwarp
66                     \LWR@textcurrentcolor{% lwarp
```

```
67                              \__siunitx_print_text:
68                        }% lwarp
69                     }% lwarp
70              }
71           \color@endgroup
72 %
73        }
74     }
75   }
76 \LWR@formatted{__siunitx_print_aux:}
77
78 \cs_new_protected:Npn \LWR@HTML@__siunitx_set_math_fam:n #1 {
79   \group_begin:
80 %    \LetLtxMacro\@ensuredmath\LWR@origensuredmath% lwarp
81 %    \LetLtxMacro\mbox\LWR@print@mbox% lwarp
82 %    \hbox_set:Nn \l__siunitx_tmp_box
83 %      {
84        \ensuremath
85          {
86            \use:c { math #1 }
87              {
88                \int_const:cn { c__siunitx_math #1 _int } { \fam }
89              }
90          }
91 %      }
92   \group_end:
93 }
94 \LWR@formatted{__siunitx_set_math_fam:n}
95
96 \cs_new_protected:Npn \LWR@HTML@__siunitx_combined_output:n #1 {
97 %
98     \group_begin:% lwarp
99   \bool_if:NTF \l__siunitx_number_parse_bool
100    {
101      \tl_clear:N \l__siunitx_number_out_tl
102      \bool_set_false:N \l__siunitx_number_compound_bool
103      \__siunitx_number_output_parse:n {#1}
104    }
105    {
```

For parse-numbers=false:

```
106          \__siunitx_unit_output_pre_print:
107          \begingroup%                          lwarp
108            \boolfalse{mathjax}%                 lwarp
109 %        \__siunitx_print:nn { number } { \ensuremath {#1} }
110            \LWR@subsingledollar%        lwarp
111              {%  alt text
112                  \textbackslash( % space
113                  \LWR@HTMLsanitizedetokenized{%
114                      \detokenize{#1}%
115                  } \textbackslash)%         lwarp
116              }
117              {siunitx}%  addl hashing
118              {%
119                  \__siunitx_print:nn { number } {%
120                      \LWR@origensuredmath{#1}%
121                  }%
122              }%                               lwarp
123      \endgroup%                               lwarp
```

```
124        \__siunitx_unit_output_print:
125      }
126    \group_end:% lwarp
127 %
128 }
129 \LWR@formatted{__siunitx_combined_output:n}
```

For parse-numbers=false:

```
130 \cs_new_protected:Npn \LWR@HTML@__siunitx_range_numbers_aux:n #1
131   {
132     \bool_if:NTF \l__siunitx_number_parse_bool
133       {
134         \tl_clear:N \l__siunitx_number_out_tl
135         \tl_clear:N \l__siunitx_number_out_saved_tl
136         \bool_set_false:N \l__siunitx_number_compound_bool
137         \__siunitx_number_output_parse:n {#1}
138         \bool_if:NT \l__siunitx_number_compound_bool
139           { \msg_error:nnx { siunitx } { multi-part-range } {#1} }
140       }
141       {
142         \__siunitx_unit_output_pre_print:
143         \begingroup%       lwarp
144            \boolfalse{mathjax}%                      lwarp
145 %           \__siunitx_print:nn { number } {#1}
146                \LWR@subsingledollar%                 lwarp
147                   {%  alt text
148                      \textbackslash( % space
149                      \LWR@HTMLsanitizedetokenized{%
150                         \detokenize{#1}%
151                      } \textbackslash)%              lwarp
152                   }%
153                {siunitx}%  addl hashing
154                {%
155                      \__siunitx_print:nn { number } {%
156                         \LWR@origensuredmath{#1}%
157                      } %                             lwarp
158                }%                                    lwarp
159         \endgroup%                                   lwarp
160         \__siunitx_unit_output_print:
161       }
162   }
163 \LWR@formatted{__siunitx_range_numbers_aux:n}
```

For parse-numbers=false:

```
164 \cs_new_protected:Npn \LWR@HTML@__siunitx_angle_print_direct_aux:nn #1#2 {
165   \tl_if_empty:nF {#1}
166     {
167       \tl_set:Nn \l__siunitx_unit_tl {#2}
168         \begingroup%                                lwarp
169            \boolfalse{mathjax}%                     lwarp
170 %           \__siunitx_print:nn { number } {#1}
171                \LWR@subsingledollar{%               lwarp
172                   \textbackslash( % space
173                   \LWR@HTMLsanitizedetokenized{%
174                      \detokenize{#1}%
175                   } \textbackslash)%                lwarp
176                }%
177                {siunitx}%
```

```
178                    {%
179                        \__siunitx_print:nn { number } {
180                            \LWR@origensuredmath{#1}%
181                        }%                                        lwarp
182                    }%                                            lwarp
183            \endgroup%                                            lwarp
184        \__siunitx_unit_output_print:
185    }
186 }
187 \LWR@formatted{__siunitx_angle_print_direct_aux:nn}
188 %
```

For quotients, the fraction code is replaced by the symbol code:

```
189 \cs_new_protected:Npn \LWR@HTML@__siunitx_number_output_quotient_fraction: {
190    \bool_set_true:N \l__siunitx_number_compound_bool
191    \__siunitx_number_output_quotient_aux_i:
192    \tl_set_eq:NN \l__siunitx_number_out_tl
193        \l__siunitx_number_numerator_tl
194    \tl_put_right:NV \l__siunitx_number_out_tl \l__siunitx_output_quotient_tl
195    \tl_put_right:NV \l__siunitx_number_out_tl
196        \l__siunitx_number_denominator_tl
197    \__siunitx_number_output_single_aux:
198 }
199 \LWR@formatted{__siunitx_number_output_quotient_fraction:}
```

For units, the fraction code is replaced by the symbol code:

```
200 \cs_new_protected:Npn \LWR@HTML@__siunitx_unit_format_fraction_fraction: {
201    \__siunitx_unit_format_fraction_symbol_aux:
202    \int_compare:nNnT { \l__siunitx_unit_denominator_int } > { 1 }
203        {
204            \bool_if:NT \l__siunitx_unit_denominator_bracket_bool
205                {
206                \tl_put_left:NV \l__siunitx_unit_denominator_tl \l__siunitx_bracket_open_tl
207                \tl_put_right:NV \l__siunitx_unit_denominator_tl \l__siunitx_bracket_close_tl
208                }
209        }
210    \tl_set_eq:NN \l__siunitx_unit_tl \l__siunitx_unit_numerator_tl
211    \tl_put_right:NV \l__siunitx_unit_tl \l__siunitx_per_symbol_tl
212    \tl_put_right:NV \l__siunitx_unit_tl \l__siunitx_unit_denominator_tl
213 }
214 \LWR@formatted{__siunitx_unit_format_fraction_fraction:}
```

```
215 \cs_new_protected:Npn \LWR@HTML@__siunitx_angle_print_astronomy_aux: {
216    \prop_get:NnNT \l__siunitx_number_out_prop { mantissa-integer }
217        \l__siunitx_tmpa_tl
218        { \__siunitx_print:nV { number } \l__siunitx_tmpa_tl }
219    \ifnumcomp{\value{LWR@leximagedepth}}{>}{0}% lwarp
220    {% leximage
221    \hbox_set:Nn \l__siunitx_angle_marker_box
222        {
223            \__siunitx_print:nn { number } { { \l__siunitx_output_decimal_tl } }
224        }
225    \hbox_set:Nn \l__siunitx_angle_unit_box
226        {
227            \__siunitx_print:nV { unit } \l__siunitx_unit_tl
228            \skip_horizontal:n { -\scriptspace }
229        }
```

```
230  \__siunitx_angle_print_astronomy_aux:n { marker }
231  \__siunitx_angle_print_astronomy_aux:n { unit }
232  \hbox_set:Nn \l__siunitx_angle_marker_box
233    {
234      \box_use:N \l__siunitx_angle_marker_box
235      \box_use:N \l__siunitx_angle_unit_box
236    }
237  \dim_compare:nNnTF
238    { \l__siunitx_angle_marker_dim } > { \l__siunitx_angle_unit_dim }
239    { \__siunitx_angle_print_astronomy_marker: }
240    { \__siunitx_angle_print_astronomy_unit: }
241  }% lateximage
242  {% not a lateximage
243      \__siunitx_print:nn { number } { { \l__siunitx_output_decimal_tl } }
244      \__siunitx_print:nV { unit } \l__siunitx_unit_tl
245  }% not a lateximage
246  \prop_get:NnNT \l__siunitx_number_out_prop { mantissa-decimal }
247    \l__siunitx_tmpa_tl
248    { \__siunitx_print:nV { number } \l__siunitx_tmpa_tl }
249 }
250 \LWR@formatted{__siunitx_angle_print_astronomy_aux:}


251 \cs_new_protected:Npn \LWR@HTML@__siunitx_textsuperscript:n #1 {\textsuperscript{#1}}
252 \LWR@formatted{__siunitx_textsuperscript:n}
253
254 \cs_new_eq:NN \LWR@HTML@__siunitx_print_text_super:n \textsuperscript
255 \LWR@formatted{__siunitx_print_text_super:n}
256
257 \cs_new_eq:NN \LWR@HTML@__siunitx_print_text_sub:n \textsubscript
258 \LWR@formatted{__siunitx_print_text_sub:n}
```

\LWR@origenduresmath is added here in case the user asks for \mathrm, etc. for
output-exponent-marker.

```
259 \cs_new_protected:Npn \LWR@HTML@__siunitx_number_format_final_exponent: {
260   \prop_get:NnN \l__siunitx_number_out_prop { exponent }
261     \l__siunitx_tmpa_tl
262   \tl_if_empty:NTF \l__siunitx_output_exponent_tl
263     {
264       \tl_set:Nx \l__siunitx_tmpa_tl
265       { ^ { \exp_not:V \l__siunitx_tmpa_tl } }
266       \tl_put_left:NV \l__siunitx_tmpa_tl \l__siunitx_exponent_base_tl
267     }
268     {
269       \tl_set:Nx  \l__siunitx_tmpa_tl
270         {
271           \LWR@origensuredmath{%     lwarp
272             \exp_not:V \l__siunitx_output_exponent_tl
273           }%                         lwarp
274           \exp_not:N \mathord
275           \exp_not:V  \l__siunitx_tmpa_tl
276         }
277     }
278   \prop_put:NnV \l__siunitx_number_out_prop { exponent-result }
279     \l__siunitx_tmpa_tl
280 }
281 \LWR@formatted{__siunitx_number_format_final_exponent:}
```

`\LWR@origensuredmath` is added here to avoid using an image for the exponent product.

```
282 \cs_new_protected:Npn \LWR@HTML@__siunitx_number_format_final_combined: {
283   \__siunitx_number_format_brackets:n { mantissa }
284   \prop_get:NnN \l__siunitx_number_out_prop { mantissa-result }
285     \l__siunitx_tmpa_tl
286   \tl_if_empty:NT \l__siunitx_output_exponent_tl
287     {
288       \tl_put_right:Nx \l__siunitx_tmpa_tl
289       {
290         \exp_not:N \LWR@origensuredmath%        lwarp
291         {
292           \bool_if:NTF \l__siunitx_tight_bool
293             { {  \exp_not:V \l__siunitx_exponent_product_tl } }
294             { { } \exp_not:V \l__siunitx_exponent_product_tl { } }
295         }
296       }
297     }
298   \prop_get:NnN \l__siunitx_number_out_prop { exponent-result }
299     \l__siunitx_tmpb_tl
300   \tl_put_right:NV \l__siunitx_tmpa_tl \l__siunitx_tmpb_tl
301   \prop_put:NnV \l__siunitx_number_out_prop { result }
302     \l__siunitx_tmpa_tl
303   \prop_put:Nnn \l__siunitx_number_out_prop
304     { result-bracket-exponent } { true }
305 }
306 \LWR@formatted{__siunitx_number_format_final_combined:}
```

`\LWR@origensuredmath` is added here to avoid using an image for the exponent product.

```
307 \cs_new_protected:Npn \LWR@HTML@__siunitx_number_output_parts_aux: {
308   \bool_if:NTF \l__siunitx_multi_repeat_bool
309     {
310       \prop_if_in:NnT \l__siunitx_number_out_prop { mantissa-result }
311         {
312           \__siunitx_number_output_parts_aux:n { mantissa }
313           \__siunitx_number_output_parts_aux:n { complex }
314         }
315       \prop_get:NnNT \l__siunitx_number_out_prop { exponent-result }
316         \l__siunitx_tmpa_tl
317         {
318           \prop_if_in:NnT \l__siunitx_number_out_prop { mantissa-result }
319             {
320               \tl_put_left:Nx \l__siunitx_tmpa_tl
321               {
322                 \exp_not:N \LWR@origensuredmath
323                 {
324                   \bool_if:NTF \l__siunitx_tight_bool
325                     { {  \exp_not:V \l__siunitx_exponent_product_tl } }
326                     { { } \exp_not:V \l__siunitx_exponent_product_tl { } }
327                 }
328               }
329             \prop_put:NnV \l__siunitx_number_out_prop { exponent }
330               \l__siunitx_tmpa_tl
331             }
332           \__siunitx_number_output_parts_print:n { exponent }
333         }
334     }
```

```
335     { \__siunitx_number_output_single: }
336 }
337 \LWR@formatted{__siunitx_number_output_parts_aux:}
```

\LWR@origensuredmath is added here to avoid using an image for the exponent product.

```
338 \cs_new_protected:Npn \LWR@HTML@__siunitx_unit_output_print: {
339   \int_compare:nNnF { \l__siunitx_unit_prefix_int } = { 0 }
340     {
341       \tl_set:Nx \l__siunitx_tmpa_tl
342         {
343           \bool_if:NTF \l__siunitx_tight_bool
344             {
345               \exp_not:N \LWR@origensuredmath%       lwarp
346                 { { \exp_not:V \l__siunitx_exponent_product_tl } }
347             }
348             {
349               \exp_not:N \LWR@origensuredmath%       lwarp
350                 { { } \exp_not:V \l__siunitx_exponent_product_tl { } }
351             }
352           \int_use:N \l__siunitx_unit_prefix_base_int
353           ^ { \int_use:N \l__siunitx_unit_prefix_int }
354         }
355       \__siunitx_print:nV { number } \l__siunitx_tmpa_tl
356     }
357   \tl_if_empty:NF \l__siunitx_unit_tl
358     {
359       \__siunitx_unit_output_number_sep:
360       \__siunitx_print:nV { unit } \l__siunitx_unit_tl
361     }
362 }
363 \LWR@formatted{__siunitx_unit_output_print:}
```

\LWR@origensuredmath is added here to avoid using an image for the exponent product.

```
364 \cs_new_protected:Npn \LWR@HTML@__siunitx_range_exponent:
365   {
366     \bool_if:NT \l__siunitx_process_fixed_bool
367       {
368         \tl_set_eq:NN \l__siunitx_tmpa_tl \l__siunitx_exponent_product_tl
369         \bool_if:NT \l__siunitx_tight_bool
370           {
371             \tl_set:Nx \l__siunitx_tmpa_tl
372               { \exp_not:N \mathord \exp_not:o \l__siunitx_tmpa_tl }
373           }
374         \tl_set:Nx \l__siunitx_tmpa_tl
375           {
376             \exp_not:N \LWR@origensuredmath {%       lwarp
377                 { } \exp_not:o \l__siunitx_tmpa_tl  { }
378             }
379             10 \exp_not:N \PrintSuperscript
380               { \int_use:N \l__siunitx_process_fixed_int }
381           }
382         \__siunitx_print:nV { number } \l__siunitx_tmpa_tl
383       }
384   }
385 \LWR@formatted{__siunitx_range_exponent:}
```

\LWR@origensuredmath is added here to avoid using an image for the exponent product.

```
386 \cs_new_protected:Npn \LWR@HTML@__siunitx_table_print_S_reserved_exponent_product:
387   {
388     \tl_set_eq:NN \l__siunitx_tmpb_tl \l__siunitx_exponent_product_tl
389     \bool_if:NT \l__siunitx_tight_bool
390       {
391         \tl_set:Nx \l__siunitx_tmpb_tl
392           { \exp_not:N \mathord \exp_not:o \l__siunitx_tmpb_tl }
393       }
394     \tl_set:Nx \l__siunitx_tmpa_tl
395       {
396       \exp_not:N \LWR@origensuredmath { { } \exp_not:o \l__siunitx_tmpb_tl { } }
397         \exp_not:o \l__siunitx_tmpa_tl
398       }
399   }
400 \LWR@formatted{__siunitx_table_print_S_reserved_exponent_product:}
```

\LWR@origensuredmath is added here to avoid using an image for the output product.

```
401 \cs_new_protected:Npn \LWR@HTML@__siunitx_number_output_product_aux: {
402   \bool_set_true:N \l__siunitx_number_compound_bool
403   \__siunitx_number_preprocess:V \l__siunitx_number_arg_tl
404   \bool_if:NF \l__siunitx_error_bool
405     {
406       \tl_if_empty:NTF \l__siunitx_number_multi_tl
407         { \__siunitx_number_output_parse_aux: }
408         { \__siunitx_number_output_quotient: }
409       \tl_if_empty:NF \l__siunitx_number_next_tl
410         {
411           \bool_if:NTF \l__siunitx_tight_bool
412             {
413               \__siunitx_print:nn { number }
414                 { \LWR@origensuredmath { \l__siunitx_output_product_tl } }
415             }
416             {
417               \__siunitx_print:nn  { number }
418             { \LWR@origensuredmath { { } \l__siunitx_output_product_tl { } } }
419             }
420           \__siunitx_number_output_parse:V \l__siunitx_number_next_tl
421         }
422     }
423 }
424 \LWR@formatted{__siunitx_number_output_product_aux:}
```

Used to detect the math font.

```
425 \cs_set_protected:Npn \__siunitx_set_math_fam:n #1 {
426   \group_begin:
427     \hbox_set:Nn \l__siunitx_tmp_box
428       {
429         \LWR@origensuredmath%         lwarp
430         {
431           \use:c { math #1 }
432             {
433               \int_const:cn { c__siunitx_math #1 _int } { \fam }
434             }
435         }
```

```
436        }
437    \group_end:
438 }
```

Force \text:

```
439 \cs_set_protected:Npn \__siunitx_range_numbers:nn #1#2
440   {
441     \__siunitx_range_numbers_aux:n {#1}
442     \text{\l__siunitx_range_phrase_tl}%      lwarp
443     \__siunitx_range_numbers_aux:n {#2}
444   }
```

Force \text:

```
445 \cs_set_protected:Npn \__siunitx_range_unit:nnnn #1#2#3#4 {
446   \__siunitx_unit_parse_options:nn {#1} {#2}
447   \bool_if:NTF \l__siunitx_range_repeat_bool
448     {
449       \__siunitx_unit_in:nn {#1} {#2}
450       \__siunitx_range_numbers_aux:n {#3}
451       \text{\l__siunitx_range_phrase_tl}%      lwarp
452       \__siunitx_range_numbers_aux:n {#4}
453     }
454     {
455       \bool_if:NT \l__siunitx_process_fixed_bool
456         { \bool_set_true:N \l__siunitx_process_drop_exponent_bool }
457       \bool_if:NT \l__siunitx_range_brackets_bool
458         { \__siunitx_print:nV { number } \l__siunitx_bracket_open_tl }
459       \__siunitx_range_numbers:nn {#3} {#4}
460       \bool_if:NT \l__siunitx_range_brackets_bool
461         { \__siunitx_print:nV { number } \l__siunitx_bracket_close_tl }
462       \__siunitx_range_exponent:
463       \__siunitx_unit_output_number_sep:
464       \__siunitx_unit_output:nn {#1} {#2}
465     }
466 }
```

```
467 \ExplSyntaxOff
```

```
468 \AtBeginDocument{
469 \sisetup{
470     detect-mode=true,
471     per-mode=symbol,      % fraction is not seen by pdftotext
472     text-celsius = {\LWR@siunitx@textcelsius},
473     text-degree = {\LWR@siunitx@textdegree},
474     text-arcminute = {\LWR@siunitx@textprime} ,
475     text-arcsecond = {\LWR@siunitx@textdblprime} ,
476 }
477 }
```

```
478 \LWR@origRequirePackage{lwarp-common-mathjax-siunitx}
```

Passing range-phrase to common-mathjax-siunitx does not seem to work with v2 using translator as it does with v3 using translations. The range-phrase therefore is set to an en-dash.

```
479 \AtBeginDocument{
```

480 \CustomizeMathJax{\def\LWRsiunitxrangephrase{\unicode{x2013}}}
481 }

<div style="text-align: right">File 458</div> **`lwarp-common-mathjax-siunitx.sty`**

<div style="text-align: right">§ 570   Package</div> # common-mathjax-siunitx

(*Emulates or patches code by* JOSEPH WRIGHT.)

common-mathjax-siunitx (*Pkg*)   common-mathjax-siunitx adds MATHJAX for siunitx and siunitx-v2.

for HTML output: For MATHJAX.
MATHJAX

The following runs much faster as separate \CusomizeMathJax calls instead of one single call.

```
1 \begin{warpMathJax}
2 \LWR@infoprocessingmathjax{siunitx}
```

```
3 \CustomizeMathJax{\newcommand{\tothe}[1]{^{#1}}}
4 \CustomizeMathJax{\newcommand{\raiseto}[2]{{#2}^{#1}}}
```

Used as an end marker when parsing values:

```
5 \CustomizeMathJax{\newcommand{\LWRsiunitxEND}{}}
```

\ang   [⟨*options*⟩] {⟨*value*⟩}

```
6 \CustomizeMathJax{\def\LWRsiunitxang#1;#2;#3;#4\LWRsiunitxEND{%
7     \ifblank{#1}{}{\num{#1}\degree}%
8     \ifblank{#2}{}{\num{#2}^{\unicode{x2032}}}% \prime
9     \ifblank{#3}{}{\num{#3}^{\unicode{x2033}}}% \dblprime
10 }}
11 \CustomizeMathJax{\newcommand{\ang}[2][]{\LWRsiunitxang#2;;;\LWRsiunitxEND}}
```

\num   [⟨*options*⟩] {⟨*value*⟩}

\num handles optional powers (e, E, d, D), multiples (x), plus and minus, and period or comma decimal output.

To split the string, \def is used with parameter delimiters. When each of the following macros is used, extra delimiters are padded to the end of the arguments of each macro when used, and the final argument of each collects any extra unused delimiters.

The number is split by dimensions (x), then by powers (E, e, D, d), then by plus / minus (+-, \pm), then by plus and minus (+, -), then into pieces before and after the decimal point or decimal comma.

Determine if the number is output with a decimal period or a decimal comma. The enclosing braces tell MATHJAX to not add extra space after the punctuation.

```
12 \ExplSyntaxOn
13 \AtBeginDocument{
14 \ifdefstring{\l__siunitx_output_decimal_tl}{{,}}
15     {\CustomizeMathJax{\def\LWRsiunitxdecimal{,}}}
16     {\CustomizeMathJax{\def\LWRsiunitxdecimal{.}}}
17 }
```

18 `\ExplSyntaxOff`

Any units which must be distributed across multiple dimensions:

19 `\CustomizeMathJax{\def\LWRsiunitxdistribunit{}}`

siunitx accepts either commas or periods as decimal points. `\LWRsiunitxprintdecimal` splits its input by periods then commas, parsing out before and after sections to print on either side of the decimal point.

`\LWRsiunitxENDTWO` is used only by `\LWRsiunitxprintdecimalsubtwo`, to avoid a parsing conflict with the more widely-used `\LWRsiunitxEND`.

The following splits by decimal commas:

20 `\CustomizeMathJax{\newcommand{\LWRsiunitxENDTWO}{}}`
21
22 `\CustomizeMathJax{\def\LWRsiunitxprintdecimalsubtwo#1,#2,#3\LWRsiunitxENDTWO{%`

If nothing is ahead of the decimal comma, add a leading zero:

23 `    \ifblank{#1}{0}{\mathrm{#1}}%`

If something is after the decimal comma, print the decimal and the fraction:

24 `    \ifblank{#2}%`
25 `        {}%`
26 `        {%`
27 `            {\LWRsiunitxdecimal}%`
28 `            \mathrm{#2}%`
29 `        }%`
30 `}}`

The following splits by decimal periods:

31 `\CustomizeMathJax{\def\LWRsiunitxprintdecimalsub#1.#2.#3\LWRsiunitxEND{%`
32 `    \LWRsiunitxprintdecimalsubtwo#1,,\LWRsiunitxENDTWO%`
33 `    \ifblank{#2}%`
34 `        {}%`
35 `        {%`
36 `            {\LWRsiunitxdecimal}%`
37 `            \LWRsiunitxprintdecimalsubtwo#2,,\LWRsiunitxENDTWO%`
38 `        }%`
39 `}}`
40
41 `\CustomizeMathJax{\newcommand{\LWRsiunitxprintdecimal}[1]{%`
42 `    \LWRsiunitxprintdecimalsub#1...\LWRsiunitxEND%`
43 `}}`

The following splits by +

44 `\CustomizeMathJax{\def\LWRsiunitxnumplus#1+#2+#3\LWRsiunitxEND{%`
45 `    \ifblank{#2}%`
46 `        {%`
47 `            \LWRsiunitxprintdecimal{#1}%`
48 `        }% no plus`
49 `        {%`
50 `            \ifblank{#1}%`
51 `                {\LWRsiunitxprintdecimal{#2}}% leading plus, ignore`
52 `                {% a+b`
53 `                    \LWRsiunitxprintdecimal{#1}%`
54 `                    \unicode{x02B}% plus sign`
55 `                    \LWRsiunitxprintdecimal{#2}%`
56 `                }%`
57 `        }%`
58 `    \LWRsiunitxdistribunit%`

59 }}

The following splits by -

```
60 \CustomizeMathJax{\def\LWRsiunitxnumminus#1-#2-#3\LWRsiunitxEND{%
61     \ifblank{#2}%
62         {\LWRsiunitxnumplus#1+++\LWRsiunitxEND}%
63         {%
64             \ifblank{#1}{}{\LWRsiunitxprintdecimal{#1}}%
65             \unicode{x02212}% mathematical minus sign
66             \LWRsiunitxprintdecimal{#2}%
67             \LWRsiunitxdistribunit%
68         }%
69 }}
```

The following splits by \pm

```
70 \CustomizeMathJax{\def\LWRsiunitxnumpmmacro#1\pm#2\pm#3\LWRsiunitxEND{%
71     \ifblank{#2}%
72         {\LWRsiunitxnumminus#1---\LWRsiunitxEND}%
73         {%
74             \LWRsiunitxprintdecimal{#1}%
75             \unicode{x0B1}% \pm
76             \LWRsiunitxprintdecimal{#2}%
77             \LWRsiunitxdistribunit%
78         }%
79 }}
```

The following splits by +-

```
80 \CustomizeMathJax{\def\LWRsiunitxnumpm#1+-#2+-#3\LWRsiunitxEND{%
81     \ifblank{#2}%
82         {\LWRsiunitxnumpmmacro#1\pm\pm\pm\LWRsiunitxEND}%
83         {%
84             \LWRsiunitxprintdecimal{#1}%
85             \unicode{x0B1}% \pm
86             \LWRsiunitxprintdecimal{#2}%
87             \LWRsiunitxdistribunit%
88         }%
89 }}
```

Processes scientific notation. Special handling for a mantissa which is either empty or only a minus sign.

```
90 \CustomizeMathJax{\newcommand{\LWRsiunitxnumscientific}[2]{%
91     \ifblank{#1}%
92         {}%
93         {%
94             \ifstrequal{#1}{-}%
95                 {-}%
96                 {\LWRsiunitxprintdecimal{#1}\times}%
97         }%
98     10^{\LWRsiunitxprintdecimal{#2}}%
99     \LWRsiunitxdistribunit%
100 }}
```

The following splits by D

```
101 \CustomizeMathJax{\def\LWRsiunitxnumD#1D#2D#3\LWRsiunitxEND{%
102     \ifblank{#2}%
103         {\LWRsiunitxnumpm#1+-+-\LWRsiunitxEND}%
104         {\mathrm{\LWRsiunitxnumscientific{#1}{#2}}}%
105 }}
```

The following splits by d

```
106 \CustomizeMathJax{\def\LWRsiunitxnumd#1d#2d#3\LWRsiunitxEND{%
107     \ifblank{#2}%
108         {\LWRsiunitxnumD#1DDD\LWRsiunitxEND}%
109         {\mathrm{\LWRsiunitxnumscientific{#1}{#2}}}%
110 }}
```

The following splits by E

```
111 \CustomizeMathJax{\def\LWRsiunitxnumE#1E#2E#3\LWRsiunitxEND{%
112     \ifblank{#2}%
113         {\LWRsiunitxnumd#1ddd\LWRsiunitxEND}%
114         {\mathrm{\LWRsiunitxnumscientific{#1}{#2}}}%
115 }}
```

The following splits by e

```
116 \CustomizeMathJax{\def\LWRsiunitxnume#1e#2e#3\LWRsiunitxEND{%
117     \ifblank{#2}%
118         {\LWRsiunitxnumE#1EEE\LWRsiunitxEND}%
119         {\mathrm{\LWRsiunitxnumscientific{#1}{#2}}}%
120 }}
```

The following splits by x

```
121 \CustomizeMathJax{\def\LWRsiunitxnumx#1x#2x#3x#4\LWRsiunitxEND{%
122     \ifblank{#2}%
123         {\LWRsiunitxnume#1eee\LWRsiunitxEND}%
124         {%
125             \ifblank{#3}%
126                 {%
127                     \LWRsiunitxnume#1eee\LWRsiunitxEND%
128                     \times%
129                     \LWRsiunitxnume#2eee\LWRsiunitxEND%
130                 }%
131                 {%
132                     \LWRsiunitxnume#1eee\LWRsiunitxEND%
133                     \times%
134                     \LWRsiunitxnume#2eee\LWRsiunitxEND%
135                     \times%
136                     \LWRsiunitxnume#3eee\LWRsiunitxEND%
137                 }%
138         }%
139 }}
```

```
140 \CustomizeMathJax{\newcommand{\num}[2][]{%
141     \LWRsiunitxnumx#2xxxxx\LWRsiunitxEND%
142 }}
```

\si   [⟨*options*⟩] {⟨*unit*⟩}

~ is converted to a thin space. Not able to convert period to thin space because the period might be in \raiseto, for example.

```
143 \CustomizeMathJax{\newcommand{\si}[2][]{%
144     \mathrm{\gsubstitute{#2}{~}{\,}}%
145 }}
```

\SI   [⟨*options*⟩] {⟨*value*⟩} [⟨*prefix*⟩] {⟨*unit*⟩}

\SI has a second optional arg, which is parsed using \ifnextchar.

```
146 \CustomizeMathJax{\def\LWRsiunitxSIopt#1[#2]#3{%
```

```
147        \def\LWRsiunitxdistribunit{\,\si{#3}}%
148        {#2}\num{#1}%
149        \def\LWRsiunitxdistribunit{}%
150 }}
151
152 \CustomizeMathJax{\newcommand{\LWRsiunitxSI}[2]{%
153        \def\LWRsiunitxdistribunit{\,\si{#2}}%
154        \num{#1}%
155        \def\LWRsiunitxdistribunit{}%
156 }}
157 \CustomizeMathJax{\newcommand{\SI}[2][]{%
158        \ifnextchar[%
159            {\LWRsiunitxSIopt{#2}}%
160            {\LWRsiunitxSI{#2}}%
161 }}
```

\numlist   [⟨*options*⟩] {⟨*list*⟩}

\numlist should only be used in text mode. If used in MathJax, it is merely printed as text, so add space around the semicolons.

```
162 \CustomizeMathJax{\newcommand{\numlist}[2][]{\text{#2}}}
```

\numrange   [⟨*options*⟩] {⟨*value1*⟩} {⟨*value2*⟩}

\numrange should only be used in text mode. If used in MathJax math, an en-dash is used instead of the range-phrase.

```
163 \CustomizeMathJax{\newcommand{\numrange}[3][]{%
164        \num{#2}\ \LWRsiunitxrangephrase\ \num{#3}%
165 }}
```

\SIlist   [⟨*options*⟩] {⟨*list*⟩}

\SIlist and \SIrange should only be used in text mode. If used in MathJax, a simple emulation is provided.

```
166 \CustomizeMathJax{\newcommand{\SIlist}[3][]{\text{#2}\,\si{#3}}}
```

\SIrange   [⟨*options*⟩] {⟨*value1*⟩} {⟨*value2*⟩} {⟨*unit*⟩}

```
167 \CustomizeMathJax{\newcommand{\SIrange}[4][]{%
168        \num{#2}\,#4\ \LWRsiunitxrangephrase\ \num{#3}\,#4%
169 }}
```

\tablenum   [⟨*options*⟩] {⟨*value*⟩}

```
170 \CustomizeMathJax{\newcommand{\tablenum}[2][]{\mathrm{#2}}}

171 \CustomizeMathJax{\newcommand{\ampere}{\mathrm{A}}}
172 \CustomizeMathJax{\newcommand{\candela}{\mathrm{cd}}}
173 \CustomizeMathJax{\newcommand{\kelvin}{\mathrm{K}}}
174 \CustomizeMathJax{\newcommand{\kilogram}{\mathrm{kg}}}
175 \CustomizeMathJax{\newcommand{\metre}{\mathrm{m}}}
176 \CustomizeMathJax{\newcommand{\mole}{\mathrm{mol}}}
177 \CustomizeMathJax{\newcommand{\second}{\mathrm{s}}}
178 %
179 \CustomizeMathJax{\newcommand{\becquerel}{\mathrm{Bq}}}
180 \CustomizeMathJax{\newcommand{\degreeCelsius}{\unicode{x2103}}}
```

```
181 \CustomizeMathJax{\newcommand{\coulomb}{\mathrm{C}}}
182 \CustomizeMathJax{\newcommand{\farad}{\mathrm{F}}}
183 \CustomizeMathJax{\newcommand{\gray}{\mathrm{Gy}}}
184 \CustomizeMathJax{\newcommand{\hertz}{\mathrm{Hz}}}
185 \CustomizeMathJax{\newcommand{\henry}{\mathrm{H}}}
186 \CustomizeMathJax{\newcommand{\joule}{\mathrm{J}}}
187 \CustomizeMathJax{\newcommand{\katal}{\mathrm{kat}}}
188 \CustomizeMathJax{\newcommand{\lumen}{\mathrm{lm}}}
189 \CustomizeMathJax{\newcommand{\lux}{\mathrm{lx}}}
190 \CustomizeMathJax{\newcommand{\newton}{\mathrm{N}}}
191 \CustomizeMathJax{\newcommand{\ohm}{\mathrm{\Omega}}}
192 \CustomizeMathJax{\newcommand{\pascal}{\mathrm{Pa}}}
193 \CustomizeMathJax{\newcommand{\radian}{\mathrm{rad}}}
194 \CustomizeMathJax{\newcommand{\siemens}{\mathrm{S}}}
195 \CustomizeMathJax{\newcommand{\sievert}{\mathrm{Sv}}}
196 \CustomizeMathJax{\newcommand{\steradian}{\mathrm{sr}}}
197 \CustomizeMathJax{\newcommand{\tesla}{\mathrm{T}}}
198 \CustomizeMathJax{\newcommand{\volt}{\mathrm{V}}}
199 \CustomizeMathJax{\newcommand{\watt}{\mathrm{W}}}
200 \CustomizeMathJax{\newcommand{\weber}{\mathrm{Wb}}}
201 \CustomizeMathJax{\newcommand{\day}{\mathrm{d}}}
202 \CustomizeMathJax{\newcommand{\degree}{\mathrm{^\circ}}}
203 \CustomizeMathJax{\newcommand{\hectare}{\mathrm{ha}}}
204 \CustomizeMathJax{\newcommand{\hour}{\mathrm{h}}}
205 \CustomizeMathJax{\newcommand{\litre}{\mathrm{l}}}
206 \CustomizeMathJax{\newcommand{\liter}{\mathrm{L}}}
207 \CustomizeMathJax{\newcommand{\arcminute}{^\prime}}
208 \CustomizeMathJax{\newcommand{\minute}{\mathrm{min}}}
209 \CustomizeMathJax{\newcommand{\arcsecond}{^{\prime\prime}}}
210 \CustomizeMathJax{\newcommand{\tonne}{\mathrm{t}}}
211 \CustomizeMathJax{\newcommand{\astronomicalunit}{au}}
212 \CustomizeMathJax{\newcommand{\atomicmassunit}{u}}
213 \CustomizeMathJax{\newcommand{\bohr}{\mathit{a}_0}}
214 \CustomizeMathJax{\newcommand{\clight}{\mathit{c}_0}}
215 \CustomizeMathJax{\newcommand{\dalton}{\mathrm{D}_\mathrm{a}}}
216 \CustomizeMathJax{\newcommand{\electronmass}{\mathit{m}_{\mathrm{e}}}}
217 \CustomizeMathJax{\newcommand{\electronvolt}{\mathrm{eV}}}
218 \CustomizeMathJax{\newcommand{\elementarycharge}{\mathit{e}}}
219 \CustomizeMathJax{\newcommand{\hartree}{\mathit{E}_{\mathrm{h}}}}
220 \CustomizeMathJax{\newcommand{\planckbar}{\mathit{\unicode{x210F}}}}
221 \CustomizeMathJax{\newcommand{\angstrom}{\mathrm{\unicode{x212B}}}}
222 \CustomizeMathJax{\let\LWRorigbar\bar}
223 \CustomizeMathJax{\newcommand{\bar}{\mathrm{bar}}}
224 \CustomizeMathJax{\newcommand{\barn}{\mathrm{b}}}
225 \CustomizeMathJax{\newcommand{\bel}{\mathrm{B}}}
226 \CustomizeMathJax{\newcommand{\decibel}{\mathrm{dB}}}
227 \CustomizeMathJax{\newcommand{\knot}{\mathrm{kn}}}
228 \CustomizeMathJax{\newcommand{\mmHg}{\mathrm{mmHg}}}
229 \CustomizeMathJax{\newcommand{\nauticalmile}{\mathrm{M}}}
230 \CustomizeMathJax{\newcommand{\neper}{\mathrm{Np}}}
231 %
232 \CustomizeMathJax{\newcommand{\yocto}{\mathrm{y}}}
233 \CustomizeMathJax{\newcommand{\zepto}{\mathrm{z}}}
234 \CustomizeMathJax{\newcommand{\atto}{\mathrm{a}}}
235 \CustomizeMathJax{\newcommand{\femto}{\mathrm{f}}}
236 \CustomizeMathJax{\newcommand{\pico}{\mathrm{p}}}
237 \CustomizeMathJax{\newcommand{\nano}{\mathrm{n}}}
238 \CustomizeMathJax{\newcommand{\micro}{\mathrm{\unicode{x00B5}}}}
239 \CustomizeMathJax{\newcommand{\milli}{\mathrm{m}}}
240 \CustomizeMathJax{\newcommand{\centi}{\mathrm{c}}}
```

```
241 \CustomizeMathJax{\newcommand{\deci}{\mathrm{d}}}
242 \CustomizeMathJax{\newcommand{\deca}{\mathrm{da}}}
243 \CustomizeMathJax{\newcommand{\hecto}{\mathrm{h}}}
244 \CustomizeMathJax{\newcommand{\kilo}{\mathrm{k}}}
245 \CustomizeMathJax{\newcommand{\mega}{\mathrm{M}}}
246 \CustomizeMathJax{\newcommand{\giga}{\mathrm{G}}}
247 \CustomizeMathJax{\newcommand{\tera}{\mathrm{T}}}
248 \CustomizeMathJax{\newcommand{\peta}{\mathrm{P}}}
249 \CustomizeMathJax{\newcommand{\exa}{\mathrm{E}}}
250 \CustomizeMathJax{\newcommand{\zetta}{\mathrm{Z}}}
251 \CustomizeMathJax{\newcommand{\yotta}{\mathrm{Y}}}
252 %
253 \CustomizeMathJax{\newcommand{\percent}{\mathrm{\%}}}
254 %
255 \CustomizeMathJax{\newcommand{\meter}{\mathrm{m}}}
256 \CustomizeMathJax{\newcommand{\metre}{\mathrm{m}}}
257 %
258 \CustomizeMathJax{\newcommand{\gram}{\mathrm{g}}}
259 \CustomizeMathJax{\newcommand{\kg}{\kilo\gram}}
260 \CustomizeMathJax{\newcommand{\of}[1]{_{\mathrm{#1}}}}
261 \CustomizeMathJax{\newcommand{\squared}{^2}}
262 \CustomizeMathJax{\newcommand{\square}[1]{\mathrm{#1}^2}}
263 \CustomizeMathJax{\newcommand{\cubed}{^3}}
264 \CustomizeMathJax{\newcommand{\cubic}[1]{\mathrm{#1}^3}}


265 \CustomizeMathJax{\newcommand{\per}{\,\mathrm{/}}}


266 \CustomizeMathJax{\newcommand{\celsius}{\unicode{x2103}}}
267 %
268 \CustomizeMathJax{\newcommand{\fg}{\femto\gram}}
269 \CustomizeMathJax{\newcommand{\pg}{\pico\gram}}
270 \CustomizeMathJax{\newcommand{\ng}{\nano\gram}}
271 \CustomizeMathJax{\newcommand{\ug}{\micro\gram}}
272 \CustomizeMathJax{\newcommand{\mg}{\milli\gram}}
273 \CustomizeMathJax{\newcommand{\g}{\gram}}
274 \CustomizeMathJax{\newcommand{\kg}{\kilo\gram}}
275 %
276 \CustomizeMathJax{\newcommand{\amu}{\mathrm{u}}}
277 %
278 \CustomizeMathJax{\newcommand{\pm}{\pico\metre}}
279 \CustomizeMathJax{\newcommand{\nm}{\nano\metre}}
280 \CustomizeMathJax{\newcommand{\um}{\micro\metre}}
281 \CustomizeMathJax{\newcommand{\mm}{\milli\metre}}
282 \CustomizeMathJax{\newcommand{\cm}{\centi\metre}}
283 \CustomizeMathJax{\newcommand{\dm}{\deci\metre}}
284 \CustomizeMathJax{\newcommand{\m}{\metre}}
285 \CustomizeMathJax{\newcommand{\km}{\kilo\metre}}
286 %
287 \CustomizeMathJax{\newcommand{\as}{\atto\second}}
288 \CustomizeMathJax{\newcommand{\fs}{\femto\second}}
289 \CustomizeMathJax{\newcommand{\ps}{\pico\second}}
290 \CustomizeMathJax{\newcommand{\ns}{\nano\second}}
291 \CustomizeMathJax{\newcommand{\us}{\micro\second}}
292 \CustomizeMathJax{\newcommand{\ms}{\milli\second}}
293 \CustomizeMathJax{\newcommand{\s}{\second}}
294 %
295 \CustomizeMathJax{\newcommand{\fmol}{\femto\mol}}
296 \CustomizeMathJax{\newcommand{\pmol}{\pico\mol}}
297 \CustomizeMathJax{\newcommand{\nmol}{\nano\mol}}
```

```
298 \CustomizeMathJax{\newcommand{\umol}{\micro\mol}}
299 \CustomizeMathJax{\newcommand{\mmol}{\milli\mol}}
300 \CustomizeMathJax{\newcommand{\mol}{\mol}}
301 \CustomizeMathJax{\newcommand{\kmol}{\kilo\mol}}
302 %
303 \CustomizeMathJax{\newcommand{\pA}{\pico\ampere}}
304 \CustomizeMathJax{\newcommand{\nA}{\nano\ampere}}
305 \CustomizeMathJax{\newcommand{\uA}{\micro\ampere}}
306 \CustomizeMathJax{\newcommand{\mA}{\milli\ampere}}
307 \CustomizeMathJax{\newcommand{\A}{\ampere}}
308 \CustomizeMathJax{\newcommand{\kA}{\kilo\ampere}}
309 %
310 \CustomizeMathJax{\newcommand{\ul}{\micro\litre}}
311 \CustomizeMathJax{\newcommand{\ml}{\milli\litre}}
312 \CustomizeMathJax{\newcommand{\l}{\litre}}
313 \CustomizeMathJax{\newcommand{\hl}{\hecto\litre}}
314 \CustomizeMathJax{\newcommand{\uL}{\micro\liter}}
315 \CustomizeMathJax{\newcommand{\mL}{\milli\liter}}
316 \CustomizeMathJax{\newcommand{\L}{\liter}}
317 \CustomizeMathJax{\newcommand{\hL}{\hecto\liter}}
318 %
319 \CustomizeMathJax{\newcommand{\mHz}{\milli\hertz}}
320 \CustomizeMathJax{\newcommand{\Hz}{\hertz}}
321 \CustomizeMathJax{\newcommand{\kHz}{\kilo\hertz}}
322 \CustomizeMathJax{\newcommand{\MHz}{\mega\hertz}}
323 \CustomizeMathJax{\newcommand{\GHz}{\giga\hertz}}
324 \CustomizeMathJax{\newcommand{\THz}{\tera\hertz}}
325 %
326 \CustomizeMathJax{\newcommand{\mN}{\milli\newton}}
327 \CustomizeMathJax{\newcommand{\N}{\newton}}
328 \CustomizeMathJax{\newcommand{\kN}{\kilo\newton}}
329 \CustomizeMathJax{\newcommand{\MN}{\mega\newton}}
330 %
331 \CustomizeMathJax{\newcommand{\Pa}{\pascal}}
332 \CustomizeMathJax{\newcommand{\kPa}{\kilo\pascal}}
333 \CustomizeMathJax{\newcommand{\MPa}{\mega\pascal}}
334 \CustomizeMathJax{\newcommand{\GPa}{\giga\pascal}}
335 %
336 \CustomizeMathJax{\newcommand{\mohm}{\milli\ohm}}
337 \CustomizeMathJax{\newcommand{\kohm}{\kilo\ohm}}
338 \CustomizeMathJax{\newcommand{\Mohm}{\mega\ohm}}
339 %
340 \CustomizeMathJax{\newcommand{\pV}{\pico\volt}}
341 \CustomizeMathJax{\newcommand{\nV}{\nano\volt}}
342 \CustomizeMathJax{\newcommand{\uV}{\micro\volt}}
343 \CustomizeMathJax{\newcommand{\mV}{\milli\volt}}
344 \CustomizeMathJax{\newcommand{\V}{\volt}}
345 \CustomizeMathJax{\newcommand{\kV}{\kilo\volt}}
346 %
347 \CustomizeMathJax{\newcommand{\W}{\watt}}
348 \CustomizeMathJax{\newcommand{\uW}{\micro\watt}}
349 \CustomizeMathJax{\newcommand{\mW}{\milli\watt}}
350 \CustomizeMathJax{\newcommand{\kW}{\kilo\watt}}
351 \CustomizeMathJax{\newcommand{\MW}{\mega\watt}}
352 \CustomizeMathJax{\newcommand{\GW}{\giga\watt}}
353 %
354 \CustomizeMathJax{\newcommand{\J}{\joule}}
355 \CustomizeMathJax{\newcommand{\uJ}{\micro\joule}}
356 \CustomizeMathJax{\newcommand{\mJ}{\milli\joule}}
357 \CustomizeMathJax{\newcommand{\kJ}{\kilo\joule}}
```

```
358 %
359 \CustomizeMathJax{\newcommand{\eV}{\electronvolt}}
360 \CustomizeMathJax{\newcommand{\meV}{\milli\electronvolt}}
361 \CustomizeMathJax{\newcommand{\keV}{\kilo\electronvolt}}
362 \CustomizeMathJax{\newcommand{\MeV}{\mega\electronvolt}}
363 \CustomizeMathJax{\newcommand{\GeV}{\giga\electronvolt}}
364 \CustomizeMathJax{\newcommand{\TeV}{\tera\electronvolt}}
365 %
366 \CustomizeMathJax{\newcommand{\kWh}{\kilo\watt\hour}}
367 %
368 \CustomizeMathJax{\newcommand{\F}{\farad}}
369 \CustomizeMathJax{\newcommand{\fF}{\femto\farad}}
370 \CustomizeMathJax{\newcommand{\pF}{\pico\farad}}
371 %
372 \CustomizeMathJax{\newcommand{\K}{\mathrm{K}}}
373 %
374 \CustomizeMathJax{\newcommand{\dB}{\mathrm{dB}}}
375 %
376 \CustomizeMathJax{\newcommand{\kibi}{\mathrm{Ki}}}
377 \CustomizeMathJax{\newcommand{\mebi}{\mathrm{Mi}}}
378 \CustomizeMathJax{\newcommand{\gibi}{\mathrm{Gi}}}
379 \CustomizeMathJax{\newcommand{\tebi}{\mathrm{Ti}}}
380 \CustomizeMathJax{\newcommand{\pebi}{\mathrm{Pi}}}
381 \CustomizeMathJax{\newcommand{\exbi}{\mathrm{Ei}}}
382 \CustomizeMathJax{\newcommand{\zebi}{\mathrm{Zi}}}
383 \CustomizeMathJax{\newcommand{\yobi}{\mathrm{Yi}}}
384 \end{warpMathJax}
```

File 459   **`lwarp-skmath.sty`**

§ 571   Package   **skmath**

(*Emulates or patches code by* Simon Sigurdhsson.)

skmath (*Pkg*)   skmath is used as-is for svg math, and is emulated for MathJax.

for HTML output: 1 \LWR@ProvidesPackagePass{skmath}[2019/10/15]

Only defined if package option requested:

```
2 \begin{warpMathJax}
3 \ExplSyntaxOn
4 \bool_if:NT\g__skmath_define_common_sets_bool{
5 \CustomizeMathJax{\newcommand{\N}{\mathbb{N}}}
6 \CustomizeMathJax{\newcommand{\Z}{\mathbb{Z}}}
7 \CustomizeMathJax{\newcommand{\Q}{\mathbb{Q}}}
8 \CustomizeMathJax{\newcommand{\R}{\mathbb{R}}}
9 \CustomizeMathJax{\newcommand{\C}{\mathbb{C}}}
10 }
```

skmath is using l3keys, which does not seem to have an equivalent to \@ifpackagewith. To detect package options, comparisons with the following are made to see if various macros have been defined as follows:

```
11 \cs_gset_nopar:Npn\LWR__skmath_imaginary_unit:n#1{{#1}}
12 \cs_gset_nopar:Npn\LWR__skmath_natural_log_e:{{e}}
13 \cs_gset_nopar:Npn\LWR__skmath_integral_d:{{d}}
```

14 `\cs_gset_nopar:Npn\LWR__skmath_total_derivative_d:{{d}}`

If `notation=iso`, use upright, else italic:

```
15 \cs_if_eq:NNTF \__skmath_imaginary_unit:n \LWR__skmath_imaginary_unit:n
16     {
17         \CustomizeMathJax{\newcommand{ii}{\mathit{i}}}
18         \CustomizeMathJax{\newcommand{jj}{\mathit{j}}}
19     }
20     {
21         \CustomizeMathJax{\newcommand{ii}{\mathrm{i}}}
22         \CustomizeMathJax{\newcommand{jj}{\mathrm{j}}}
23     }
```

If `notation=iso`, use upright, else italic:

```
24 \cs_if_eq:NNTF \__skmath_natural_log_e: \LWR__skmath_natural_log_e:
25     { \CustomizeMathJax{\newcommand{\ee}{\mathit{e}}} }
26     { \CustomizeMathJax{\newcommand{\ee}{\mathrm{e}}} }
```

skmath uses `\DeclarePairedDelimiter` from mathtools for `\abs` and `\norm`, and lwarp uses this to automatically define MATHJAX definitions for each.

If `notation=english`, use slanted, else upright:

```
27 \cs_if_eq:NNTF \__skmath_integral_d: \LWR__skmath_integral_d:
28     { \CustomizeMathJax{\newcommand{\d}{\mathit{d}}} }
29     { \CustomizeMathJax{\newcommand{\d}{\mathrm{d}}} }
```

Used to parse comma and caret arguments for `\pd` and `\td`:

30 `\CustomizeMathJax{\def\LWRskmathEND{}}`

Parse the arguments with up to four commas. Argument 6 contains any leftover commas.

```
31 \CustomizeMathJax{\def\LWRskmathpdstarsub#1#2,#3,#4,#5,#6\LWRskmathEND{
32     #1_{#2#3#4#5}%
33 }}
34
35 \CustomizeMathJax{\newcommand{\LWRskmathpdstar}[2]{%
36     \LWRskmathpdstarsub{#1}#2,,,,\LWRskmathEND%
37 }}
```

Parse the arguments with up to two carets. Argument 3 contains any leftover carets. `\LWRskmathpdplus` is used to only place a plus sign starting after the first term. `\LWRskmathpdone` is used to only place a 1 digit if a second or later term does not have a power.

```
38 \CustomizeMathJax{\def\LWRskmathpdnumerator#1^#2^#3\LWRskmathEND{%
39     \ifblank{#1}{}{
40         \ifblank{#2}{\LWRskmathpdplus\LWRskmathpdone}{\LWRskmathpdplus#2}
41     }
42 }}
```

Parse the arguments with up to two carets. Argument 3 contains any leftover carets.

```
43 \CustomizeMathJax{\def\LWRskmathpddenominator#1^#2^#3\LWRskmathEND{%
44     \ifblank{#1}{}{%
45         \ifblank{#2}%
46             {\partial{#1}}%
47             {\partial{#1}^{#2}}%
48     }%
49 }}
```

Factored from \LWRskmathpdnostarsub, following:

The phrase ^{} appears to be required while parsing the carets. \LWRskmathpdplus is used to only place a plus sign starting after the first term. \LWRskmathpdone is used to only place a 1 digit if a second or later term does not have a power.

This may not be recursion-safe. (Is there really such as a thing as nested differentials?)

```
50 \CustomizeMathJax{\newcommand{\LWRskmathdonumerator}[5]{%
51     \partial^{%
52         \def\LWRskmathpdplus{}%
53         \LWRskmathpdnumerator#2^{}^{}\LWRskmathEND%
54         \def\LWRskmathpdplus{+}%
55         \def\LWRskmathpdone{1}%
56         \LWRskmathpdnumerator#3^{}^{}\LWRskmathEND%
57         \LWRskmathpdnumerator#4^{}^{}\LWRskmathEND%
58         \LWRskmathpdnumerator#5^{}^{}\LWRskmathEND%
59     }%
60     {#1}%
61 }}
62
63 \CustomizeMathJax{\newcommand{\LWRskmathdodenominator}[4]{%
64     \LWRskmathpddenominator#1^{}^{}^{}\LWRskmathEND%
65     \ifblank{#2}{}{\,}%
66     \LWRskmathpddenominator#2^{}^{}^{}\LWRskmathEND%
67     \ifblank{#3}{}{\,}%
68     \LWRskmathpddenominator#3^{}^{}^{}\LWRskmathEND%
69     \ifblank{#4}{}{\,}%
70     \LWRskmathpddenominator#4^{}^{}^{}\LWRskmathEND%
71 }}
```

Parse the arguments with up to four commas. Argument 6 contains any leftover commas.

```
72 \CustomizeMathJax{\def\LWRskmathpdnostarsub#1#2,#3,#4,#5,#6\LWRskmathEND{
73     \ifblank{#3}{\def\LWRskmathpdone{}}{\def\LWRskmathpdone{1}}
74     \frac%
75         {\LWRskmathdonumerator{#1}{#2}{#3}{#4}{#5}}%
76         {\LWRskmathdodenominator{#2}{#3}{#4}{#5}}%
77 }}
78
79 \CustomizeMathJax{\newcommand{\LWRskmathpdnostar}[2]{%
80     \LWRskmathpdnostarsub{#1}#2,,,,,,\LWRskmathEND%
81 }}
82 \CustomizeMathJax{\newcommand{\pd}{\ifstar\LWRskmathpdstar\LWRskmathpdnostar}}
```

If notation=english or legacy, use slanted, else upright:

```
83 \cs_if_eq:NNTF \__skmath_total_derivative_d: \LWR__skmath_total_derivative_d:
```

```
84      { \CustomizeMathJax{\newcommand{\LWRskmathtd}{\mathit{d}}} }
85      { \CustomizeMathJax{\newcommand{\LWRskmathtd}{\mathrm{d}}} }

86 \CustomizeMathJax{\def\LWRskmathtdsub#1#2^#3\LWRskmathEND{%
87      \frac
88          {\LWRskmathtd^{#3}{#1}}
89          {\LWRskmathtd{#2}^{#3}}
90 }}
91
92 \CustomizeMathJax{\newcommand{\td}[2]{%
93      \LWRskmathtdsub{#1}#2^{}\LWRskmathEND%
94 }}

95 \CustomizeMathJax{\newcommand{\E}[1]{%
96      \operatorname{E}\left[#1\right]%
97 }}

98 \CustomizeMathJax{\let\given\mid}
99
100 \CustomizeMathJax{\newcommand{\P}[1]{%
101      \operatorname{P}%
102      \left(#1\right)%
103 }}

104 \CustomizeMathJax{\newcommand{\var}[1]{%
105    \operatorname{Var}\left(#1\right)%
106 }}
107
108 \CustomizeMathJax{\newcommand{\cov}[2]{%
109    \operatorname{Cov}\left(#1,#2\right)%
110 }}
```

Common code for \sin etc:

```
111 \CustomizeMathJax{\newcommand{\LWRskmathtrigtwo}[2][]{%
112      \ifblank{#1}{}{^{#1}}%
113      \ifblank{#2}{}{\left(#2\right)}%
114 }}
115
116 \CustomizeMathJax{\newcommand{\LWRskmathtrig}[1]{%
117      \operatorname{#1}%
118      \LWRskmathtrigtwo%
119 }}

120 \CustomizeMathJax{\renewcommand{\sin}{\LWRskmathtrig{sin}}}
121 \CustomizeMathJax{\renewcommand{\arcsin}{\LWRskmathtrig{arcsin}}}
122
123 \CustomizeMathJax{\renewcommand{\cos}{\LWRskmathtrig{cos}}}
124 \CustomizeMathJax{\renewcommand{\arccos}{\LWRskmathtrig{arccos}}}
125
126 \CustomizeMathJax{\renewcommand{\tan}{\LWRskmathtrig{tan}}}
127 \CustomizeMathJax{\renewcommand{\arctan}{\LWRskmathtrig{arctan}}}
128
129 \CustomizeMathJax{\renewcommand{\cot}{\LWRskmathtrig{cot}}}
130
131 \CustomizeMathJax{\renewcommand{\sinh}{\LWRskmathtrig{sinh}}}
132 \CustomizeMathJax{\renewcommand{\cosh}{\LWRskmathtrig{cosh}}}
133 \CustomizeMathJax{\renewcommand{\tanh}{\LWRskmathtrig{tanh}}}
```

Common code for \ln and \log:

```
134 \CustomizeMathJax{\newcommand{\LWRskmathlogtwo}[2][]{%
135     \ifblank{#1}{}{_{#1}}%
136     \ifblank{#2}{}{\left(#2\right)}%
137 }}
138
139 \CustomizeMathJax{\newcommand{\LWRskmathlog}[1]{%
140     \operatorname{#1}%
141     \LWRskmathlogtwo%
142 }}
143
144 \CustomizeMathJax{\renewcommand{\ln}{\LWRskmathlog{ln}}}
145 \CustomizeMathJax{\renewcommand{\log}{\LWRskmathlog{log}}}

145 \CustomizeMathJax{\newcommand{\LWRskmathexpparens}[1]{%
146     \operatorname{exp}%
147     \ifblank{#1}{}{\left(#1\right)}%
148 }}
```

See the skmath source for the original of the following:

```
149 \CustomizeMathJax{\newcommand{\LWRskmathexpnostar}[1]{%
150     \mathchoice
151         {\ee^{#1}}
152         {\LWRskmathexpparens{#1}}
153         {\LWRskmathexpparens{#1}}
154         {\LWRskmathexpparens{#1}}
155 }}
156
157 \CustomizeMathJax{\renewcommand{\exp}{\ifstar\LWRskmathexpparens\LWRskmathexpnostar}}
```

Common code for \min etc:

```
158 \CustomizeMathJax{\newcommand{\LWRskmathminstar}[2][]{%
159     \operatorname{\LWRskmathminname}%
160     \ifblank{#1}{}{%
161         _{\mathchoice{\mathclap{#1}}{#1}{#1}{#1}}
162     }%
163     \ifblank{#2}{}{#2}%
164 }}
165
166 \CustomizeMathJax{\newcommand{\LWRskmathminnostar}[2][]{%
167     \ifblank{#1}%
168         {\operatorname{\LWRskmathminname}}%
169         {%
170             \underset%
171                 {\mathchoice{\mathclap{#1}}{#1}{#1}{#1}}%
172                 {\operatorname{\LWRskmathminname}}%
173         }%
174     \ifblank{#2}{}{\left\{#2\right\}}%
175 }}
```

\LWRskmathminname seems to be recursion-safe since it is used immediately.

```
175 \CustomizeMathJax{\newcommand{\LWRskmathmin}[1]{%
176     \def\LWRskmathminname{#1}%
177     \ifstar\LWRskmathminstar\LWRskmathminnostar%
178 }}
```

```
179 \CustomizeMathJax{\renewcommand{\min}{\LWRskmathmin{min}}}
180 \CustomizeMathJax{\renewcommand{\argmin}{\arg\LWRskmathmin{min}}}
181
182 \CustomizeMathJax{\renewcommand{\max}{\LWRskmathmin{max}}}
183 \CustomizeMathJax{\renewcommand{\argmax}{\arg\LWRskmathmin{max}}}
184 \CustomizeMathJax{\renewcommand{\sup}{\LWRskmathmin{sup}}}
185 \CustomizeMathJax{\renewcommand{\inf}{\LWRskmathmin{inf}}}

186 \CustomizeMathJax{\let\bar\overline}
187
188 \CustomizeMathJax{\let\vec\boldsymbol}
```

Remember the original definitions:

```
189 \CustomizeMathJax{\let\LWRskmathRe\Re}
190 \CustomizeMathJax{\let\LWRskmathIm\Im}
```

Redefine depending on notation=iso:

```
191 \bool_if:NTF\g__skmath_iso_complex_parts_bool{
192     \CustomizeMathJax{\renewcommand{\Re}[1]{%
193         \LWRskmathRe%
194         \ifblank{#1}{}{\left(#1\right)}%
195     }}
196     \CustomizeMathJax{\renewcommand{\Im}[1]{%
197         \LWRskmathIm%
198         \ifblank{#1}{}{\left(#1\right)}%
199     }}
200 }{
201     \CustomizeMathJax{\renewcommand{\Re}[1]{%
202         \operatorname{Re}%
203         \ifblank{#1}{}{#1}%
204     }}
205     \CustomizeMathJax{\renewcommand{\Im}[1]{%
206         \operatorname{Im}%
207         \ifblank{#1}{}{#1}%
208     }}
209 }
210
211 \ExplSyntaxOff
212 \end{warpMathJax}
```

File 460   **`lwarp-slantsc.sty`**

§ 572   Package   **slantsc**

(*Emulates or patches code by* HARALD HARDERS.)

slantsc (*Pkg*)   slantsc is emulated for HTML, and used as-is for print output.

for HTML output: 1 \LWR@ProvidesPackagePass{slantsc}[2012/01/01]

```
2 \newcommand*{\LWR@HTML@noscshape}{}
3 \LWR@formatted{noscshape}
4
5 \FilenameNullify{%
```

```
6     \LetLtxMacro\noscshape\@empty%
7 }
```

File 461   **lwarp-slashed.sty**

§ 573   Package   # slashed

(*Emulates or patches code by* DAVID CARLISLE.)

slashed (*Pkg*)   slashed works as-s for HTML SVG math. For MATHJAX, emulation is provided.

for HTML output: 1 \LWR@ProvidesPackagePass{slashed}[1997/01/16]

```
2 \begin{warpMathJax}
3 \CustomizeMathJax{\newcommand{\slashed}[1]{\cancel{#1}}}
4 \end{warpMathJax}
```

File 462   **lwarp-soul.sty**

§ 574   Package   # soul

(*Emulates or patches code by* MELCHIOR FRANZ.)

soul (*Pkg*)   soul is emulated.

for HTML output: 1 \LWR@ProvidesPackageDrop{soul}[2003/11/17]
2 \RequirePackage{xcolor}% for \convertcolorspec

Storage for the colors to use:

```
3 \newcommand*{\LWR@soululcolor}{}
4
5 \newcommand*{\LWR@soulstcolor}{}
6
7 % \definecolor{LWR@soulhlcolordefault}{HTML}{F8E800}
8 % \newcommand*{\LWR@soulhlcolor}{LWR@soulhlcolordefault}
9 \newcommand*{\LWR@soulhlcolor}{}
```

\so   {⟨*text*⟩}

Basic markup with CSS:

```
10 \newcommand{\so}[1]{%
11 \InlineClass(letter-spacing:.2ex){letterspacing}{#1}%
12 }
```

\caps   {⟨*text*⟩}

```
13 \newcommand{\caps}[1]{%
14     \InlineClass%
15         (font-variant:small-caps;letter-spacing:.1ex)%
16         {capsspacing}{#1}%
17 }
```

\LWR@soulcolor   {⟨*text*⟩} {⟨*color*⟩} {⟨*class*⟩} {⟨*colorstyle*⟩} {⟨*FormatWPstyle*⟩}

Add colors if not empty:

```
18 \newcommand{\LWR@soulcolor}[5]{%
19 \ifcsempty{#2}%
20 {%
21     \InlineClass(#5){#3}{#1}%
22 }%
23 {%
24     \convertcolorspec{named}{\@nameuse{#2}}{HTML}\LWR@tempcolor%
25     \LWR@htmlspanclass[#5;#4:\LWR@origpound\LWR@tempcolor]{#3}{#1}%
26 }%
27 }
```

```
28 \newcommand{\ul}[1]{%
29 \LWR@soulcolor{#1}{LWR@soululcolor}{uline}{text-decoration-color}%
30     {text-decoration:underline; text-decoration-skip: auto;}%
31 }
32
33 \newcommand{\st}[1]{
34 \LWR@soulcolor{#1}{LWR@soulstcolor}{sout}{text-decoration-color}%
35     {text-decoration:line-through}%
36 }
37
38 \newcommand{\hl}[1]{
39 \LWR@soulcolor{#1}{LWR@soulhlcolor}{highlight}{background-color}%
40     {background:\LWR@origpound{}F8E800}
41 }
```

Nullified:

```
42 \newcommand*{\soulaccent}[1]{}
43 \newcommand*{\soulregister}[2]{}
44 \newcommand{\sloppyword}[1]{#1}
45 \newcommand*{\sodef}[5]{\DeclareRobustCommand*#1[1]{\so{##1}}}
46 \newcommand*{\resetso}{}
47 \newcommand*{\capsdef}[5]{}
48 \newcommand*{\capsreset}{}
49 \newcommand*{\capssave}[1]{}
50 \newcommand*{\capsselect}[1]{}
51 \newcommand*{\setul}[2]{}
52 \newcommand*{\resetul}{}
53 \newcommand*{\setuldepth}[1]{}
54 \newcommand*{\setuloverlap}[1]{}
55 \newcommand*{\<}{}
```

Set colors:

```
56 \newcommand*{\setulcolor}[1]{\renewcommand{\LWR@soululcolor}{#1}}
57 \newcommand*{\setstcolor}[1]{\renewcommand{\LWR@soulstcolor}{#1}}
58 \newcommand*{\sethlcolor}[1]{\renewcommand{\LWR@soulhlcolor}{#1}}
```

Long versions of the user-level macros:

```
59 \let\textso\so
60 \let\textul\ul
61 \let\texthl\hl
62 \let\textcaps\caps
```

File 463 **`lwarp-soulpos.sty`**

§ 575 Package **soulpos**

(*Emulates or patches code by* JAVIER BEZOS.)

soulpos (*Pkg*) soulpos is emulated.

for HTML output:
```
1 \RequirePackage{soul}
2 \RequirePackage{soulutf8}
3 \LWR@ProvidesPackageDrop{soulpos}[2012/02/25]

4 \NewDocumentCommand{\ulposdef}{m o m}{}
5
6 \newdimen\ulwidth
7
8 \newcommand\ifulstarttype[1]{%
9 \expandafter\@secondoftwo%
10 }
11
12 \newcommand\ifulendtype[1]{%
13 \expandafter\@secondoftwo%
14 }
15
16 \newcommand{\ulstarttype}{0}
17 \newcommand{\ulendtype}{0}
18 \newcommand\ulpostolerance{0}%
```

File 464 **`lwarp-soulutf8.sty`**

§ 576 Package **soulutf8**

soulutf8 (*Pkg*) soulutf8 is emulated.

lwarp's HTML output naturally supports UTF-8 encoding.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{soulutf8}[2016/05/16]
2 \RequirePackage{soul}
```

File 465 **`lwarp-splitbib.sty`**

§ 577 Package **splitbib**

(*Emulates or patches code by* NICOLAS MARKEY.)

splitbib (*Pkg*) splitbib is patched for use by lwarp.

for HTML output:
```
1 \LWR@ProvidesPackagePass{splitbib}[2005/12/22]
```

```
2 \def\NMSB@stylebox#1#2{%
3 \begin{BlockClass}[text-align:center ; border: 1px solid black]{splitbibbox}
4    \csname SB\NMSB@level font\endcsname{\LWR@textcurrentfont{#1#2}}
5 \end{BlockClass}
6 }
7
8 \def\NMSB@stylebar#1#2{%
9 \begin{BlockClass}[%
10    text-align:center ;
11    border-top: 1px solid black ;
12    border-bottom: 1px solid black ;
13 ]{splitbibbar}
14    \csname SB\NMSB@level font\endcsname{\LWR@textcurrentfont{#1#2}}
15 \end{BlockClass}
16 }
17
18 \def\NMSB@styledash#1#2{%
19 \begin{BlockClass}[%
20    text-align:center ;
21 ]{splitbibdash}
22    \csname SB\NMSB@level font\endcsname{\LWR@textcurrentfont{---~#1#2~---}}
23 \end{BlockClass}
24 }
25
26 \def\NMSB@stylenone#1#2{%
27    \par
28 }
29
30 \def\NMSB@stylesimple#1#2{%
31 \par
32 \csname SB\NMSB@level font\endcsname{\LWR@textcurrentfont{#1#2}}
33 \par
34 }
```

File 466   **lwarp-splitidx.sty**

§ 578   Package   **splitidx**

(*Emulates or patches code by* MARKUS KOHM.)

splitidx (*Pkg*)   splitidx is patched for use by lwarp.

If the latexmk option is selected for lwarp, *latexmk* will compile the document but will *not* compile the indexes. **lwarpmk printindex** and **lwarpmk htmlindex** will still be required.

⚠ \thepage   When using \AtWriteToIndex or \AtNextWriteToIndex, the user must not refer to \thepage during HTML output, as the concept of a page number is meaningless. Instead, do

```
\addtocounter{LWR@autoindex}{1}
\LWR@new@label{LWRindex-\arabic{LWR@autoindex}}
```

where the \index-like action occurs, and then refer to \arabic{LWR@autoindex} instead of \thepage where the reference should occur.

See section 702.17 in the lwarp-patch-memoir package for the \@@wrspindexhyp
macro as an example.

1 \LWR@ProvidesPackagePass{splitidx}[2016/02/18]

2 \VerifyCommand[lwarp][splitidx]{\newindex}{84695DF9965D5007036BA0B4023C59B5}
3
4 \catcode`_=12%
5 \xpatchcmd{\newindex}
6   {\jobname-#2.idx}
7   {\jobname-#2_html.idx}
8   {}
9   {\LWR@patcherror{splitidx}{@newindex}}
10 \catcode`_=8%

Patched to use lwarp's automatic indexing counter instead of \thepage:

11 \VerifyCommand[lwarp][splitidx]{\@wrsindex}{6E1A6193E20ABD0DFD6A1FC3F35113A6}
12
13 \renewcommand*{\@wrsindex}[2][]{%
14   \ifx\relax#1\relax
15   \if@splitidx
16   \@wrsindex[idx]{#2}%
17   \else
18   \def\@tempa{#2}%
19   \if@verbindex\@onelevel@sanitize\@tempa\fi
20   \@wrindex{\@tempa}%
21   \fi
22   \else
23   \def\@tempa{#2}%
24   \csname index@#1@hook\endcsname
25 %   \expandafter\ifx\csname @@wrsindex\endcsname\relax
26   \addtocounter{LWR@autoindex}{1}%   lwarp
27 %   \@@@wrsindex{#1}{{\@tempa}{\thepage}}%
28   \@@@wrsindex{#1}{{\@tempa}{\arabic{LWR@autoindex}}}%
29 %   \else
30 %   \def\@tempb{\@@wrsindex{#1}}%
31 %   \expandafter\@tempb\@tempa||\\%
32 %   \fi

The label is assigned after the file write to avoid conflict with cleveref.

33   \label{LWRindex-\arabic{LWR@autoindex}}%   lwarp
34   \endgroup
35   \@esphack
36   \fi
37 }

lwarp defines sectioning commands with xparse, so the below patches are done
as temporary redefinitions instead of being \let.

Not using \VerifyCommand here since the patches are not likely to be affected by
changes in the original.

38 \xpatchcmd{\printsubindex}
39   {\let\section\subsection}
40   {\renewcommand*{\section}{\subsection}}
41   {}
42   {\LWR@patcherror{splitidx}{printsubindex-section}}

```
43
44 \xpatchcmd{\printsubindex}
45     {\let\chapter\section}
46     {\renewcommand*{\chapter}{\section}}
47     {}
48     {\LWR@patcherror{splitidx}{printsubindex-chapter}}
49
50 \xpatchcmd{\printsubindex}
51     {\let\@makechapterhead\section}
52     {\def\@makechapterhead{\section}}
53     {}
54     {\LWR@patcherror{splitidx}{printsubindex-chapter}}
```

File 467   **`lwarp-srcltx.sty`**

§ 579   Package   # srcltx

srcltx (*Pkg*)   srcltx is ignored.

for HTML output: 1 `\LWR@ProvidesPackageDrop{srcltx}[2006/11/12]`

2 `\newif\ifSRCOK \SRCOKfalse`
3 `\newcommand*\srcIncludeHook[1]{}`
4 `\newcommand*\srcInputHook[1]{}`
5 `\newcommand*\MainFile{}`
6 `\def\MainFile{\jobname.tex}`
7 `\newcommand*\CurrentInput{}`
8 `\gdef\CurrentInput{\MainFile}`
9 `\newcommand\Input{}`
10 `\let\Input\input`

File 468   **`lwarp-srctex.sty`**

§ 580   Package   # srctex

srctex (*Pkg*)   srctex is ignored.

for HTML output: 1 `\LWR@ProvidesPackageDrop{srctex}[2006/11/12]`
2 `\LWR@origRequirePackage{lwarp-srcltx}`

File 469   **`lwarp-stabular.sty`**

§ 581   Package   # stabular

(*Emulates or patches code by* Sigitas Tolušis.)

stabular (*Pkg*)   stabular is emulated.

for HTML output: 1 `\LWR@ProvidesPackageDrop{stabular}[2014/03/20]`

Env stabular   [⟨*vpos*⟩] {⟨*colspec*⟩}

```
2 \newenvironment{stabular}[2][c]
3 {
4 \begin{tabular}[#1]{#2}
5 \renewcommand{\noalign}[1]{}
6 }
7 {\end{tabular}}
```

Env stabular   {⟨*width*⟩} [⟨*vpos*⟩] {⟨*colspec*⟩}

```
8 \NewDocumentEnvironment{stabular*}{m o m}
9 {
10 \begin{tabular}[#2]{#3}
11 \renewcommand{\noalign}[1]{}
12 }
13 {\end{tabular}}
```

File 470   **lwarp-stackengine.sty**

§ 582   Package   # stackengine

(*Emulates or patches code by* STEVEN B. SEGLETES.)

stackengine (*Pkg*)   stackengine is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{stackengine}[2017/02/13]

Not using \VerifyCommand here because these patches appear to be fairly transparent to changes in the original

The original version is neccessary for the patched \@stack and \stackanchor, where nesting lateximages does not work:

```
2 \LetLtxMacro\LWR@orig@stackengine\stackengine
```

```
3 \renewcommand*{\stackengine}[8]{%
4     \ifstrequal{#4}{O}%
5         {\begin{lateximage}[\ImageAltText]?}%
6         {\begin{lateximage}[\ImageAltText]?[][vertical-align:top]}%
7     \LWR@orig@stackengine{#1}{#2}{#3}{#4}{#5}{#6}{#7}{#8}%
8     \end{lateximage}%
9 }
```

\@stack uses a lateximage with a vertical alignment:

```
10 \LetLtxMacro\LWR@orig@@stack\@stack
11
12 \xpatchcmd{\LWR@orig@@stack}{\stackengine}{\LWR@orig@stackengine}
13     {}
14     {\LWR@patcherror{stackengine}{LWR@orig@@stack}}
15
16 \renewcommand*{\@stack}[4]{%
17     \ifstrequal{#3}{O}%
18         {\begin{lateximage}[\ImageAltText]?}%
19         {\begin{lateximage}[\ImageAltText]?[][vertical-align:top]}%
```

```
20      \LWR@orig@@stack{#1}{#2}{#3}{#4}%
21      \end{lateximage}%
22 }
```

The lapping macros are disabled for HTML:

```
23 \newcommand*\LWR@HTML@@stacklap[4]{#3}
24 \LWR@formatted{@stacklap}
```

\stackanchor is patched for two instances of \stackengine. A lateximage with vertical alignment is used.

```
25 \xpatchcmd{\stackanchor}{\stackengine}{\LWR@orig@stackengine}
26      {}
27      {\LWR@patcherror{stackengine}{stackanchor patch 1}}
28
29 \xpatchcmd{\stackanchor}{\stackengine}{\LWR@orig@stackengine}
30      {}
31      {\LWR@patcherror{stackengine}{stackanchor patch 2}}
32
33 \xpretocmd{\stackanchor}
34      {\begin{lateximage}[\ImageAltText]?[][vertical-align:middle]}%
35      {}
36      {\LWR@patcherror{stackengine}{stackanchor pre}}
37
38 \xapptocmd{\stackanchor}{\end{lateximage}}
39      {}
40      {\LWR@patcherror{stackengine}{stackanchor app}}
```

\Centerstack is simply placed inside a lateximage with a vertical alignment:

```
41 \xpretocmd{\Centerstack}
42      {\begin{lateximage}[\ImageAltText]?[][vertical-align:middle]}%
43      {}
44      {\LWR@patcherror{stackengine}{Centerstack pre}}
45
46 \xapptocmd{\Centerstack}{\end{lateximage}}
47      {}
48      {\LWR@patcherror{stackengine}{Centerstack app}}
```

\savestack reverts to print mode while saving the box, then places it inside a lateximage when used:

```
49 \VerifyCommand[lwarp][stackengine]{\savestack}{4B06A7F9D3F0B829FE293FB452D43430}
50
51 \renewcommand*\savestack[2]{%
52   \xdef\sv@name{\stack@macro@name{#1}}%
53   \@ifundefined{\sv@name content}{%
54     \expandafter\newsavebox\expandafter{\csname\sv@name content\endcsname}%
55   }{}%
56     \begingroup%       lwarp
57     \LWR@restoreorigformatting%       lwarp
58   \RenewDocumentEnvironment{lateximage}{s o s t? o o d()}{}{}% lwarp: inside group
59 \expandafter\LWR@gsavebox\csname\sv@name content\endcsname{#2}%
60 \expandafter\gdef\expandafter#1\expandafter{%
61       \expandafter\begin\expandafter{lateximage\expandafter}%       lwarp
62       \expandafter\usebox\expandafter%
63       {\csname\sv@name content\endcsname}%
64       \expandafter\end\expandafter{lateximage\expandafter}%         lwarp
```

```
65     }%
66     \endgroup%        lwarp
67 }
```

File 471 **`lwarp-stackrel.sty`**

§ 583 Package # stackrel

(*Emulates or patches code by* HEIKO OBERDIEK.)

stackrel (*Pkg*) stackrel is used as-is for SVG math, and is emulated for MATHJAX.

for HTML output:
```
1 \LWR@ProvidesPackagePass{stackrel}[2016/05/16]
```

```
2 \begin{warpMathJax}
3 \CustomizeMathJax{\renewcommand{\stackrel}[3][]{%
4     \mathrel{\mathop{#3}\limits_{#1}^{#2}}%
5 }}
6
7 \CustomizeMathJax{\newcommand{\stackbin}[3][]{%
8     \mathbin{\mathop{#3}\limits_{#1}^{#2}}%
9 }}
10 \end{warpMathJax}
```

File 472 **`lwarp-statex2.sty`**

§ 584 Package # statex2

(*Emulates or patches code by* RODNEY A SPARAPANI.)

statex2 (*Pkg*) statex2 is patched for use by lwarp, and emulated for MATHJAX.

⚠ As of this version, option autobold does not appear to work for PDF output.

⚠ For MATHJAX, the tilde character ~ does not create \sim. Use \sim directly.

⚠ Because MATHJAX has limited conditional processing:

- \wrap only creates square braces, no matter what its optional arguments.

- \P, \pCau, \pN, and \pU do not handle special cases.

⚠ \and To have \and work if using \maketitle, place the following after the start of the document:

```
\newcommand*{\and}{%
    \relax\ifmmode%
        \expandafter\;\mb{\mathrm{and}}\;%
    \else%
        \expandafter\STATEXand%
    \fi%
}
```

1 `\LWR@ProvidesPackagePass{statex2}[2011/09/14]`

2 `\newcommand*{\LWR@HTML@Alpha}[1][]{%`
3 `    \fcolorbox{black}{ForestGreen}{\textcolor{white}{\textsf{ALPHA}}}%`
4 `    \textbf{\textcolor{ForestGreen}{\textsf{#1}}}%`
5 `}`
6 `\LWR@formatted{Alpha}`
7
8 `\newcommand*{\LWR@HTML@List}[1]{%`
9 `    \textbf{\textcolor{Dandelion}{\textsf{L}\textsubscript{\textit{#1}}}}%`
10 `}`
11 `\LWR@formatted{List}`
12
13 `\newcommand*{\LWR@HTML@Snd}[1][]{%`
14 `    \fcolorbox{black}{Dandelion}{\textcolor{white}{\textsf{2nd}}}%`
15 `    \textbf{\textcolor{Dandelion}{\textsf{#1}}}%`
16 `}`
17 `\LWR@formatted{Snd}`
18
19 `\begin{warpMathJax}`
20 `\LWR@infoprocessingmathjax{statex2}`
21
22 `\CustomizeMathJax{\newcommand{\cpi}{\boldsymbol{\pi}}}`
23 `\CustomizeMathJax{\newcommand{\c}[1]{\boldsymbol{\mathrm{#1}}}}`
24 `\CustomizeMathJax{\newcommand{\sfsl}[1]{\mathsf#1}}%      not slanted`
25
26 `\if@manualbold`
27 `\CustomizeMathJax{\newcommand{\mb}[1]{#1}}`
28 `\else`
29 `\CustomizeMathJax{\newcommand{\mb}[1]{\boldsymbol{#1}}}`
30 `\fi`
31
32 `\CustomizeMathJax{\newcommand{\diag}{\mb{\mathrm{diag}}}}`
33 `\CustomizeMathJax{\newcommand{\blockdiag}{\mb{\mathrm{blockdiag}}}}`
34 `\CustomizeMathJax{\newcommand{\erf}{\mb{\mathrm{erf}}}}`
35 `\CustomizeMathJax{\newcommand{\logit}{\mb{\mathrm{logit}}}}`
36 `\CustomizeMathJax{\newcommand{\trace}{\mb{\mathrm{trace}}}}`
37
38 `\CustomizeMathJax{\newcommand{\chisq}{{\mb{\chi^2}}}}`
39 `\CustomizeMathJax{\newcommand{\deriv}[2]{\mb{\frac{\d{}}{\d{#1}}}\wrap{\mb{#2}}}}`
40 `\CustomizeMathJax{\newcommand{\derivf}[2]{\mb{\frac{\d{}}{\d{#2}}}\wrap{\mb{#1}}}}`
41 `\CustomizeMathJax{\newcommand{\e}[1]{\mb{\mathrm{e}^{#1}}}}`
42 `\CustomizeMathJax{\newcommand{\E}[2][]{\mb{\mathrm{E}}_{\mb{#1}} \wrap{\mb{#2}}}}`
43 `\CustomizeMathJax{\newcommand{\ha}{{\mb{\frac{\alpha}{2}}}}}`
44 `\CustomizeMathJax{\newcommand{\I}[2][]{%`
45 `    \mb{\mathrm{I}}_{\mb{#1}} \LWRwrapparen{\mb{#2}}%`
46 `}}`
47 `\CustomizeMathJax{\newcommand{\IBeta}[2]{%`
48 `    \mb{\frac{\Gamma[#1+#2]}{\Gamma[#1]\Gamma[#2]}}%`
49 `}}`
50 `\CustomizeMathJax{\newcommand{\If}{\;\mb{\mathrm{if}}\;}}`
51 `\CustomizeMathJax{\newcommand{\im}{\mb{\mathrm{i}}}}`
52 `\CustomizeMathJax{\newcommand{\ol}{\overline}}`
53 `\CustomizeMathJax{\newcommand{\ow}{\;\mb{\mathrm{otherwise}}\;}}`
54 `\CustomizeMathJax{\newcommand{\pderiv}[2]{%`
55 `    \mb{\frac{\partial}{\partial #1}}\wrap{\mb{#2}}%`
56 `}}`
57 `\CustomizeMathJax{\newcommand{\pderivf}[2]{%`
58 `    \mb{\frac{\partial}{\partial #2}}\wrap{\mb{#1}}%`
59 `}}`

```
60 \CustomizeMathJax{\newcommand{\sd}{\mb{\sigma}}}
61 \CustomizeMathJax{\newcommand{\ul}{\underline}}
62 \CustomizeMathJax{\newcommand{\V}[2][]{\mb{\mathrm{V}}_{\mb{#1}} \wrap{\mb{#2}}}}
63 \CustomizeMathJax{\newcommand{\vs}{\;\mb{\mathrm{vs.}}\;}}
64 \CustomizeMathJax{\newcommand{\where}{\;\mb{\mathrm{where}}\;}}
65 \CustomizeMathJax{\newcommand{\wrap}[2][]{\left[ #2 \right]}}%        only []
66 \CustomizeMathJax{\newcommand{\LWRwrapparen}[1]{\left( #1 \right)}}%   lwarp
67
68 % \CustomizeMathJax{\renewcommand{~}{\mb{\sim}}}% doesn't work,
69 % replace <space>~<space> with <space>\sim<space>
70
71 \CustomizeMathJax{\newcommand{\iid}{\;\stackrel{\mb{\mathrm{iid}}}{\sim}\;}}
72 \CustomizeMathJax{\newcommand{\ind}{\;\stackrel{\mb{\mathrm{ind}}}{\sim}\;}}
73 \CustomizeMathJax{\newcommand{\indpr}{%
74     \;\stackrel{\mb{\mathrm{ind}}}{\stackrel{\mb{\mathrm{prior}}}{\sim}}\;
75 }}
76 \CustomizeMathJax{\newcommand{\post}{\;\stackrel{\mb{\mathrm{post}}}{\sim}\;}}
77 \CustomizeMathJax{\newcommand{\prior}{\;\stackrel{\mb{\mathrm{prior}}}{\sim}\;}}
78
79 \CustomizeMathJax{\let\STATEXGamma=\Gamma}
80 \CustomizeMathJax{\renewcommand{\Gamma}[1][]{\mb{\STATEXGamma}\LWRwrapparen{\mb{#1}}}}
81 %
82 \CustomizeMathJax{\renewcommand{\and}{\;\mb{\mathrm{and}}\;}}
83 %
84 \CustomizeMathJax{\newcommand{\H}{\mb{\mathrm{H}}}}
85 %
86 \CustomizeMathJax{\newcommand{\P}[2][]{\mb{\mathrm{P}}_{\mb{#1}}\wrap{\mb{#2}}}}
87 %
88 \CustomizeMathJax{\newcommand{\|}{\mb{\mid}}}
89
90 \CustomizeMathJax{\newcommand{\B}[1]{\mb{\mathrm{B}}\LWRwrapparen{\mb{#1}}}}
91 \CustomizeMathJax{\newcommand{\BB}[1]{\mb{\mathrm{BetaBin}}\LWRwrapparen{\mb{#1}}}}
92 \CustomizeMathJax{\newcommand{\Bin}[2]{\mb{\mathrm{Bin}}\LWRwrapparen{\mb{#1,\ #2}}}}
93 \CustomizeMathJax{\newcommand{\Dir}[1]{\mb{\mathrm{Dirichlet}}\LWRwrapparen{\mb{#1}}}}
94 \CustomizeMathJax{\newcommand{\HG}[3]{%
95     \mb{\mathrm{Hypergeometric}}\LWRwrapparen{\mb{#1,\ #2,\ #3}}%
96 }}
97 \CustomizeMathJax{\newcommand{\M}[2]{%
98     \mb{\mathrm{Multinomial}}\LWRwrapparen{\mb{#1,\ #2}}%
99 }}
100 \CustomizeMathJax{\newcommand{\NB}[2]{\mb{\mathrm{NegBin}}\LWRwrapparen{\mb{#1,\ #2}}}}
101 \CustomizeMathJax{\newcommand{\Poi}[1]{\mb{\mathrm{Poisson}}\LWRwrapparen{\mb{#1}}}}
102 \CustomizeMathJax{\let\Poisson=\Poi}
103
104 \CustomizeMathJax{\newcommand{\pBB}[4][x]{%
105     \mb{\frac{\Gamma[#2+1]\Gamma[#3+#1]\Gamma[#2+#4-#1]\Gamma[#3+#4]}%
106     {\Gamma[#1+1]\Gamma[#2-#1+1]\Gamma[#2+#3+#4]\Gamma[#3]\Gamma[#4]}%
107     \I[#1]{\{0, 1,\., #2\}}, \where #3>0,\; #4>0 \and n=1, 2,\.}%
108 }}
109 \CustomizeMathJax{\newcommand{\pBin}[3][x]{%
110     \mb{\binom{#2}{#1}#3^{#1}} \LWRwrapparen{\mb{{1-#3}^{#2-#1}}}%
111     \mb{\I[#1]{\{0,1,\.,#2\}}, \where p \in (0, 1) \and n=1, 2,\.}%
112 }}
113 \CustomizeMathJax{\newcommand{\pPoi}[2][x]{%
114     \mb{\frac{1}{#1!}#2^{#1}\e{-#2}\I[#1]{\{0, 1,\.\}}, \where #2>0}%
115 }}
116
117 \CustomizeMathJax{\newcommand{\Cau}[2]{\mb{\mathrm{Cauchy}}\LWRwrapparen{\mb{#1,\ #2}}}}
118 \CustomizeMathJax{\let\Cauchy=\Cau}
119 \CustomizeMathJax{\newcommand{\Chi}[2][]{%
```

```
120      \chisq_{\mb{#1}}\LWRwrapparen{\mb{#2}}%
121 }}
122 \CustomizeMathJax{\let\Chisq=\Chi}
123 \CustomizeMathJax{\newcommand{\Bet}[2]{\mb{\mathrm{Beta}}\LWRwrapparen{\mb{#1,\ #2}}}}
124 \CustomizeMathJax{\let\Beta=\Bet}
125 \CustomizeMathJax{\newcommand{\Exp}[1]{\mb{\mathrm{Exp}}\LWRwrapparen{\mb{#1}}}}
126 \CustomizeMathJax{\newcommand{\F}[2]{\mb{\mathrm{F}}\LWRwrapparen{\mb{#1,\ #2}}}}
127 \CustomizeMathJax{\newcommand{\Gam}[2]{\mb{\mathrm{Gamma}}\LWRwrapparen{\mb{#1,\ #2}}}}
128 \CustomizeMathJax{\newcommand{\IC}[1]{\mb{\mathrm{\chi^{-2}}}\LWRwrapparen{\mb{#1}}}}
129 \CustomizeMathJax{\newcommand{\IG}[2]{%
130      \mb{\mathrm{Gamma^{-1}}}\LWRwrapparen{\mb{#1,\ #2}}%
131 }}
132 \CustomizeMathJax{\newcommand{\IW}[2]{%
133      \mb{\mathrm{Wishart^{-1}}}\LWRwrapparen{\mb{#1,\ #2}}%
134 }}
135 \CustomizeMathJax{\newcommand{\Log}[2]{%
136 \mb{\mathrm{Logistic}}\LWRwrapparen{\mb{#1,\ #2}}%
137 }}
138 \CustomizeMathJax{\newcommand{\LogN}[2]{%
139      \mb{\mathrm{Log\!-\!N}}\LWRwrapparen{\mb{#1,\ #2}}%
140 }}
141 \CustomizeMathJax{\newcommand{\N}[3][]{%
142      \mb{\mathrm{N}}_{\mb{#1}}\LWRwrapparen{\mb{#2,\ #3}}%
143 }}
144 \CustomizeMathJax{\newcommand{\Par}[2]{\mb{\mathrm{Pareto}}\LWRwrapparen{\mb{#1,\ #2}}}}
145 \CustomizeMathJax{\let\Pareto=\Par}
146 \CustomizeMathJax{\newcommand{\Tsq}[2]{\mb{\mathrm{T^2}}\LWRwrapparen{\mb{#1,\ #2}}}}
147 \CustomizeMathJax{\newcommand{\U}[1]{\mb{\mathrm{U}}\LWRwrapparen{\mb{#1}}}}
148 \CustomizeMathJax{\newcommand{\W}[2]{\mb{\mathrm{Wishart}}\LWRwrapparen{\mb{#1,\ #2}}}}
149
150 \CustomizeMathJax{\renewcommand{\t}[1]{\mb{\mathrm{t}}\LWRwrapparen{\mb{#1}}}}
151
152 \CustomizeMathJax{\newcommand{\pBet}[3][x]{%
153      \IBeta{#2}{#3}%
154      #1^{#2-1}\LWRwrapparen{1-#1}^{#3-1}\I[#1]{0,\ 1}, \where #2>0 \and #3>0%
155 }}
156 \CustomizeMathJax{\newcommand{\pCau}[3][x]{%
157 %    \ifthenelse{\equal{#2, #3}{0, 1}}{\frac{1}{\cpi\LWRwrapparen{1+#1}^2}}%
158     {\frac{1}{#3\cpi\left\{1+\wrap{\LWRwrapparen{x-#2}/#3}^2\right\}}}, \where #3>0%
159 }}% no special case for 0,1
160 \CustomizeMathJax{\newcommand{\pChi}[2][x]{%
161      \frac{2^{-#2/2}}{\Gamma[#2/2]}#1^{#2/2-1}\e{-#1/2}%
162      \I[#1]{0,\infty}, \where #2>0%
163 }}
164 \CustomizeMathJax{\newcommand{\pExp}[2][x]{%
165      \frac{1}{#2}\e{-#1/#2}\I[#1]{0,\infty},%
166      \where #2>0%
167 }}
168 \CustomizeMathJax{\newcommand{\pGam}[3][x]{%
169      \frac{#3^{#2}}{\Gamma[#2]}#1^{#2-1}\e{-#3#1}%
170      \I[#1]{0,\infty}, \where #2>0 \and #3>0%
171 }}
172 \CustomizeMathJax{\newcommand{\pN}[3][x]{%
173 %    \ifthenelse{\equal{#2, #3}{0, 1}}%
174 %    {\frac{1}{\sqrt{2\cpi}}\e{-#1^2/2}}%
175     {\frac{1}{\sqrt{2\cpi \cdot #3}}\e{-\LWRwrapparen{#1-#2}^2/2 \cdot #3}}%
176 }}% no test for 0,1, must add \cdot
177 \CustomizeMathJax{\newcommand{\pPar}[3][x]{%
178      \frac{#3}{#2\LWRwrapparen{1+#1/#2}^{#3+1}}\I[#1]{0,\infty},%
179      \where #2>0 \and #3>0%
```

```
180 }}
181 \CustomizeMathJax{\newcommand{\pU}[3][x]{%
182 %      \ifthenelse{\equal{#2, #3}{0, 1}}{\I[#1]{0,\ 1}}%
183     {\frac{1}{#3-#2}\I[#1]{#2,\ #3}, \where #2<#3}%
184 }}% no special case for 0,1
185
186 \CustomizeMathJax{\newcommand{\=}[1]{\bar{#1}}}
187 \CustomizeMathJax{\let\^\widehat}
188 \CustomizeMathJax{\let\~\widetilde}
189 \CustomizeMathJax{\newcommand{\'}[1]{\LWRwrapparen{\mb{#1}}}}
190 \CustomizeMathJax{\newcommand{\b}[1]{\bar{#1}}}
191 \CustomizeMathJax{\newcommand{\c}[1]{\mb{\mathrm{#1}}}}
192 \CustomizeMathJax{\newcommand{\d}[1]{\,\mb{\mathrm{d}{#1}}}}
193 \CustomizeMathJax{\newcommand{\.}{\mb{\ldots}}}
194 \end{warpMathJax}
```

File 473   **`lwarp-statistics.sty`**

§ 585   Package   # statistics

(*Emulates or patches code by* JULIEN RIVAUD.)

statistics (*Pkg*)   statistics is patched for use by lwarp.

⚠ \color   The statistics documentation examples include the use of the \color macro. Use
\textcolor instead.

⚠ math   The statistics package uses math arrays, but the HTML version uses text tabulars
to allow text copy/paste. If math is required, use \ensuremath or \(and \) as
needed.

Pre/postline is ignored, and \hline is used instead. Each table will have an \hline
above and below as a frame.

for HTML output: 1 \LWR@ProvidesPackagePass{statistics}[2019/09/29]

2 \ExplSyntaxOn

To use text tabular instead of math array. This allows text copy/paste of the
results.

In the following, all changes for the Lwarp package are labelled "lwarp".

Redefined using the lwarp version of &:

```
3 \VerifyCommand[lwarp][statistics]{\__statistics_table_make:nn}
4    {DC8BA2460EA83AE75FA0C0F00E775B5E}
5
6 \StartDefiningTabulars%      lwarp, no other changes below
7 \cs_set_protected_nopar:Nn \__statistics_table_make:nn {
8    \int_compare:nT
9        { 0 < \l__statistics_table_maxcols_int
10            = \l__statistics_nbvals_int } {
11      \__statistics_table_end:
12      \tl_use:N \l__statistics_table_sep_tl
13      \__statistics_table_start:
14    }
```

```
15      \int_incr:N \l__statistics_nbvals_int
16      \int_incr:N \l__statistics_currange_int
17      \fp_add:Nn \l__statistics_curtotal_fp { #2 }
18      \__statistics_set_if_shown:N \l_tmpa_bool
19      \tl_set:Nx \l_tmpa_tl {
20          \exp_not:n { & \tl_set:Nn \currentcolumn } {
21              \int_use:N \l__statistics_currange_int
22          }
23      }
24      \bool_if:NTF \l_tmpa_bool {
25          \tl_put_right:Nn \l_tmpa_tl
26              {\__statistics_table_shown_format:n}
27      }{
28          \tl_put_right:Nn \l_tmpa_tl
29              {\__statistics_table_hidden_format:n}
30      }
31      \seq_put_right:Nn \l__statistics_store_values_seq { #1 }
32      \bool_if:NT \l_statistics_table_values_bool {
33          \tl_put_right:Nx \l__statistics_table_values_tl {
34              \exp_not:V \l_tmpa_tl {
35                  \exp_not:n {
36                      \__statistics_table_values_format:n { #1 }
37                  }
38              }
39          }
40      }
41      \seq_put_right:Nx \l__statistics_store_counts_seq { \fp_eval:n {#2} }
42      \bool_if:NT \l_statistics_table_counts_bool {
43          \tl_put_right:Nx \l__statistics_table_counts_tl {
44              \exp_not:V \l_tmpa_tl {
45                  \exp_not:n {
46                      \__statistics_table_counts_format:n {
47                          { \__statistics_table_allcounts_format:n { #2 } }
48                      }
49                  }
50              }
51          }
52      }
53      \bool_if:NT \l_statistics_table_icc_bool {
54          \tl_put_right:Nx \l__statistics_table_icc_tl {
55              \exp_not:V \l_tmpa_tl {
56                  \exp_not:n { \__statistics_table_icc_format:n }
57                  {
58                      \exp_not:n{ \__statistics_table_allcounts_format:n }
59                          { \fp_use:N \l__statistics_curtotal_fp }
60                  }
61              }
62          }
63      }
64      \bool_if:NT \l_statistics_table_dcc_bool {
65          \tl_put_right:Nx \l__statistics_table_dcc_tl {
66              \exp_not:V \l_tmpa_tl {
67                  \exp_not:n { \__statistics_table_dcc_format:n }
68                  {
69                      \exp_not:n{ \__statistics_table_allcounts_format:n }
70                      {
71                          \fp_eval:n {
72                              \l__statistics_total_fp
73                                  - \l__statistics_curtotal_fp
74                                  + #2
```

```
75                                    }
76                                }
77                            }
78                        }
79                    }
80                }
81          \fp_set:Nn \l__statistics_table_curICF_fp {
82                round(\l__statistics_curtotal_fp
83                        / \l__statistics_total_fp,
84                    \l__statistics_table_round_int)
85          }
86          \bool_if:NT \l__statistics_table_frequencies_bool {
87                \tl_put_right:Nx \l__statistics_table_frequencies_tl {
88                    \exp_not:V \l_tmpa_tl {
89                        \exp_not:n { \__statistics_table_frequencies_format:n }
90                        {
91                            \exp_not:n{ \__statistics_table_allfreqs_format:n }
92                            {
93                                \fp_eval:n {
94                                    \l__statistics_table_curICF_fp
95                                        - \l__statistics_table_prevICF_fp
96                                }
97                            }
98                        }
99                    }
100               }
101         }
102         \bool_if:NT \l__statistics_table_icf_bool {
103               \tl_put_right:Nx \l__statistics_table_icf_tl {
104                   \exp_not:V \l_tmpa_tl {
105                       \exp_not:n { \__statistics_table_icf_format:n }
106                       {
107                           \exp_not:n{ \__statistics_table_allfreqs_format:n }
108                               { \fp_to_decimal:N \l__statistics_table_curICF_fp }
109                       }
110                   }
111               }
112         }
113         \bool_if:NT \l__statistics_table_dcf_bool {
114               \tl_put_right:Nx \l__statistics_table_dcf_tl {
115                   \exp_not:V \l_tmpa_tl {
116                       \exp_not:n { \__statistics_table_dcf_format:n }
117                       {
118                           \exp_not:n{ \__statistics_table_allfreqs_format:n }
119                           {
120                               \fp_eval:n {
121                                   1 - \l__statistics_table_prevICF_fp
122                               }
123                           }
124                       }
125                   }
126               }
127         }
128         \fp_set_eq:NN
129             \l__statistics_table_prevICF_fp
130             \l__statistics_table_curICF_fp
131 }
132 \StopDefiningTabulars%  lwarp
```

Redefined using `tabular`. Also, preline and postline do not work correctly with lwarp, which looks for certain tokens to detect \hlines, so \hline is used instead.

```
133 \VerifyCommand[lwarp][statistics]{\__statistics_table_end:}
134     {B2F9FC5A36B44E6E06A8D9807FCBAA6D}
135
136 \cs_set_protected_nopar:Nn \__statistics_table_end: {
137     \tl_set:Nx \l__statistics_table_preamble_tl {
138 %          \exp_not:n { \begin{array}[ }
139        \exp_not:n {\begin{tabular}[ }%       lwarp
140            \exp_not:V \l__statistics_table_valign_tl
141        \exp_not:n { ] }
142                { \exp_not:V \l__statistics_table_headcoltype_tl
143                    \prg_replicate:nn { \l__statistics_nbvals_int }
144                        { \exp_not:V \l__statistics_table_coltype_tl } }
145     }
146     \seq_clear:N \l__statistics_table_contents_seq
147     \clist_map_inline:nn { values, counts, icc, dcc, frequencies, icf, dcf } {
148        \bool_if:cT { l__statistics_table_##1_bool } {
149            \seq_put_right:Nv
150                    \l__statistics_table_contents_seq
151                    { l__statistics_table_##1_tl }
152        }
153     }
154 %     $
155     \tl_use:N \l__statistics_table_preamble_tl
156        \hline%        lwarp
157 %          \l__statistics_table_preline_tl
158        \seq_use:Nn
159                \l__statistics_table_contents_seq
160                { \l__statistics_table_newline_tl }
161        \\
162 %          \l__statistics_table_postline_tl
163        \hline%        lwarp
164 %     \end{array}$
165     \end{tabular}%        lwarp
166 }
```

With lwarp, \ensuremath creates an SVG image, but its alt tag does not contain the text of the contents for copy/paste, since these expressions are usually not simple text. For the statistics package, copy/paste is restored by using text instead of math output.

For the leftmost column. Redefined to use text output:

```
167 \VerifyCommand[lwarp][statistics]{\__statistics_table_start:}
168     {624FAC0783057B481861D9F02764F6C5}
169
170 \cs_set_protected_nopar:Nn \__statistics_table_start: {
171     \int_zero:N \l__statistics_nbvals_int
172     \clist_pop:NNT \l__statistics_table_maxcols_clist \l_tmpa_tl {
173        \int_set:Nn \l__statistics_table_maxcols_int { \l_tmpa_tl }
174     }
175     \clist_map_inline:nn { values, counts, frequencies, icc, icf, dcc, dcf } {
176        \tl_set:cx { l__statistics_table_##1_tl } {
177 %            \exp_not:N \ensuremath {
178                \exp_not:N \hbox {
179                    \exp_not:c { l__statistics_table_##1_name_tl }
180                }
181 %            }
```

```
182          }
183      }
184 }
```

For the first row. Redefined to use text output:

```
185 \VerifyCommand[lwarp][statistics]{\__statistics_IN:w}
186     {DD1B22587CFB4DEDBEE4D8E9A1E0CCAF}
187
188 \RenewDocumentCommand \__statistics_IN:w { m u{;} u{;} m } {
189 %     \ensuremath{ \left#1 \num{#2} \mathbin{;} \num{#3} \right#4 }
190     #1 #2 ; #3 #4%        lwarp
191 }
192
193 \__statistics_setup:nn { table } {
194 %     values/format = \ensuremath{#1},
195     values/format = {#1},%        lwarp
196 }
```

Added \ExplSyntaxOn/Off to avoid errors. (In once instance, a double subscript error appeared.)

```
197 \VerifyCommand[lwarp][statistics]{\StatsGraph}
198     {998267D2E90514DBDFD5544FB69AD6C8}
199
200 \RenewDocumentCommand \StatsGraph { +O{} +m +O{} } {
201     \group_begin:
202     \int_gincr:N \g__statistics_graph_last_int
203     \tl_set:Nx \l_tmpa_tl {
204         \exp_not:n { g__statistics_graph_xstep_ }
205         \int_use:N \g__statistics_graph_last_int
206         \exp_not:n { _tl }
207     }
208     \tl_if_exist:cTF { \l_tmpa_tl } {
209         \fp_gset:Nn \g__statistics_graph_xstep_fp
210             { \tl_use:c {\l_tmpa_tl} }
211     }{
212         \fp_gset:Nn \g__statistics_graph_xstep_fp { \c_one_int }
213     }
214     \__statistics_setup:nn { graph } { #1, #3 }
215     \tl_if_single:nTF { #2 } {
216         \cs_if_exist:NF #2 { #2 }
217         \tl_set_eq:NN \l__statistics_data_tl #2
218     }{
219         \tl_set:Nn \l__statistics_data_tl { #2 }
220     }
221     \fp_zero:N \l__statistics_graph_maxheight_fp
222     \fp_set:Nn \l__statistics_graph_minvalue_fp {inf}
223     \fp_set:Nn \l__statistics_graph_maxvalue_fp {-inf}
224     \fp_zero:N \l__statistics_total_fp
225     \int_zero:N \l__statistics_nbvals_int
226     \bool_set_true:N \l__statistics_graph_allranges_bool
227     \keyval_parse:NNV
228             \__statistics_graph_prepare:n
229             \__statistics_graph_prepare:nn
230             \l__statistics_data_tl
231     \tl_clear:N \l__statistics_graph_tikzdata_tl
232     \tl_clear:N \l__statistics_graph_tikzinfo_tl
233     \int_zero:N \l__statistics_currange_int
234     \bool_if:NTF \l__statistics_graph_allranges_bool {
```

```
235         \bool_if:NTF \l__statistics_graph_cumulative_bool {
236 \ExplSyntaxOn%     lwarp
237             \__statistics_graph_dopicture_cumulative:
238 \ExplSyntaxOff%    lwarp
239         }{
240 \ExplSyntaxOn%     lwarp
241             \__statistics_graph_dopicture_hist:
242 \ExplSyntaxOff%    lwarp
243         }
244     }{
245 \ExplSyntaxOn%    lwarp
246         \__statistics_graph_dopicture_comb:
247 \ExplSyntaxOff%    lwarp
248     }
249     \iow_now:Nx \@auxout {
250         \exp_not:n {
251             \ExplSyntaxOn
252             \tl_gset:cn
253         }
254         {
255             \exp_not:n {g__statistics_graph_xstep_}
256             \int_use:N \g__statistics_graph_last_int
257             \exp_not:n {_tl}
258         }
259         {
260             \fp_to_decimal:N \g__statistics_graph_xstep_fp
261         }
262         \exp_not:n {
263             \ExplSyntaxOff
264         }
265     }
266     \group_end:
267 }
268
269 \ExplSyntaxOff
```

File 474   **lwarp-statmath.sty**

§ 586   Package   # statmath

(*Emulates or patches code by* Sebastian Ankargren.)

statmath (*Pkg*)   statmath is used as-is for svg math, and is emulated for MathJax.

for HTML output:

```
1 \LWR@ProvidesPackagePass{statmath}[2018/03/08]

2 \LWR@origRequirePackage{lwarp-common-mathjax-letters}
3
4 \begin{warpMathJax}
5 \LWR@infoprocessingmathjax{statmath}
6
7 \ifdefequal{\abcbf}{\mathbf}
8     {\CustomizeMathJax{\newcommand{\abcbf}[1]{\mathbf{#1}}}}
9     {\CustomizeMathJax{\newcommand{\abcbf}[1]{\boldsymbol{#1}}}}
10
11 \CustomizeMathJax{\newcommand{\greekbf}[1]{\boldsymbol{#1}}}
12
```

```
13 \CustomizeMathJax{\newcommand{\bfA}{\abcbf A}}
14 \CustomizeMathJax{\newcommand{\bfB}{\abcbf B}}
15 \CustomizeMathJax{\newcommand{\bfC}{\abcbf C}}
16 \CustomizeMathJax{\newcommand{\bfD}{\abcbf D}}
17 \CustomizeMathJax{\newcommand{\bfE}{\abcbf E}}
18 \CustomizeMathJax{\newcommand{\bfF}{\abcbf F}}
19 \CustomizeMathJax{\newcommand{\bfG}{\abcbf G}}
20 \CustomizeMathJax{\newcommand{\bfH}{\abcbf H}}
21 \CustomizeMathJax{\newcommand{\bfI}{\abcbf I}}
22 \CustomizeMathJax{\newcommand{\bfJ}{\abcbf J}}
23 \CustomizeMathJax{\newcommand{\bfK}{\abcbf K}}
24 \CustomizeMathJax{\newcommand{\bfL}{\abcbf L}}
25 \CustomizeMathJax{\newcommand{\bfM}{\abcbf M}}
26 \CustomizeMathJax{\newcommand{\bfN}{\abcbf N}}
27 \CustomizeMathJax{\newcommand{\bfO}{\abcbf O}}
28 \CustomizeMathJax{\newcommand{\bfP}{\abcbf P}}
29 \CustomizeMathJax{\newcommand{\bfQ}{\abcbf Q}}
30 \CustomizeMathJax{\newcommand{\bfR}{\abcbf R}}
31 \CustomizeMathJax{\newcommand{\bfS}{\abcbf S}}
32 \CustomizeMathJax{\newcommand{\bfT}{\abcbf T}}
33 \CustomizeMathJax{\newcommand{\bfU}{\abcbf U}}
34 \CustomizeMathJax{\newcommand{\bfV}{\abcbf V}}
35 \CustomizeMathJax{\newcommand{\bfW}{\abcbf W}}
36 \CustomizeMathJax{\newcommand{\bfX}{\abcbf X}}
37 \CustomizeMathJax{\newcommand{\bfY}{\abcbf Y}}
38 \CustomizeMathJax{\newcommand{\bfZ}{\abcbf Z}}
39 \CustomizeMathJax{\newcommand{\bfa}{\abcbf a}}
40 \CustomizeMathJax{\newcommand{\bfb}{\abcbf b}}
41 \CustomizeMathJax{\newcommand{\bfc}{\abcbf c}}
42 \CustomizeMathJax{\newcommand{\bfd}{\abcbf d}}
43 \CustomizeMathJax{\newcommand{\bfe}{\abcbf e}}
44 \CustomizeMathJax{\newcommand{\bff}{\abcbf f}}
45 \CustomizeMathJax{\newcommand{\bfg}{\abcbf g}}
46 \CustomizeMathJax{\newcommand{\bfh}{\abcbf h}}
47 \CustomizeMathJax{\newcommand{\bfi}{\abcbf i}}
48 \CustomizeMathJax{\newcommand{\bfj}{\abcbf j}}
49 \CustomizeMathJax{\newcommand{\bfk}{\abcbf k}}
50 \CustomizeMathJax{\newcommand{\bfl}{\abcbf l}}
51 \CustomizeMathJax{\newcommand{\bfm}{\abcbf m}}
52 \CustomizeMathJax{\newcommand{\bfn}{\abcbf n}}
53 \CustomizeMathJax{\newcommand{\bfo}{\abcbf o}}
54 \CustomizeMathJax{\newcommand{\bfp}{\abcbf p}}
55 \CustomizeMathJax{\newcommand{\bfq}{\abcbf q}}
56 \CustomizeMathJax{\newcommand{\bfr}{\abcbf r}}
57 \CustomizeMathJax{\newcommand{\bfs}{\abcbf s}}
58 \CustomizeMathJax{\newcommand{\bft}{\abcbf t}}
59 \CustomizeMathJax{\newcommand{\bfu}{\abcbf u}}
60 \CustomizeMathJax{\newcommand{\bfv}{\abcbf v}}
61 \CustomizeMathJax{\newcommand{\bfw}{\abcbf w}}
62 \CustomizeMathJax{\newcommand{\bfx}{\abcbf x}}
63 \CustomizeMathJax{\newcommand{\bfy}{\abcbf y}}
64 \CustomizeMathJax{\newcommand{\bfz}{\abcbf z}}
65
66 \LWR@mathjax@addgreek@l@bfit{bf}{}% Greek lowercase bold face italic
67 \LWR@mathjax@addgreek@u@bfup*{bf}{}% Greek uppercase bold face upright, cap macros.
68
69 \CustomizeMathJax{\newcommand{\bfzero}{\greekbf 0}}
70
71 \CustomizeMathJax{\DeclareMathOperator{\cov}{Cov}}
72 \CustomizeMathJax{\DeclareMathOperator{\E}{E}}
```

```
73 \CustomizeMathJax{\DeclareMathOperator{\V}{V}}
74 \CustomizeMathJax{\newcommand{\inas}{\overset{a.s.}{\to}}}
75 \CustomizeMathJax{\newcommand{\indist}{\overset{d}{\to}}}
76 \CustomizeMathJax{\newcommand{\inprob}{\overset{p}{\to}}}
77 \CustomizeMathJax{\DeclareMathOperator{\plim}{plim}}
78 \CustomizeMathJax{\DeclareMathOperator{\tr}{tr}}
79 \CustomizeMathJax{\DeclareMathOperator{\vc}{vec}}
80 \CustomizeMathJax{\DeclareMathOperator{\vcs}{vecs}}
81 \CustomizeMathJax{\DeclareMathOperator{\vch}{vech}}
82 \CustomizeMathJax{\DeclareMathOperator{\diag}{diag}}
83 \CustomizeMathJax{\DeclareMathOperator{\argmin}{arg\,min}}
84 \CustomizeMathJax{\DeclareMathOperator{\argmax}{arg\,max}}
85 \end{warpMathJax}
```

File 475   **`lwarp-steinmetz.sty`**

§ 587   Package   # steinmetz

(*Emulates or patches code by* ENRICO GREGORIO.)

steinmetz (*Pkg*)   steinmetz is patched for use by lwarp. Emulation is provided for MATHJAX

for HTML output: 1 `\LWR@ProvidesPackagePass{steinmetz}[2009/06/14]`

```
2 \renewcommand{\phase}[2][]{%
3     \begin{lateximage}*[steinmetz\{\detokenize{#2}\}]?%
4     \ensuremath{\underline{/#2}}
5     \end{lateximage}
6 }
7
8 \begin{warpMathJax}
9 \CustomizeMathJax{\newcommand{\phase}[2][]{\underline{/#2}}}
10 \end{warpMathJax}
```

File 476   **`lwarp-stfloats.sty`**

§ 588   Package   # stfloats

stfloats (*Pkg*)   stfloats is ignored.

for HTML output: 1 `\LWR@ProvidesPackageDrop{stfloats}[2017/03/27]`

stfloats may have been preloaded by a ltj* class.

The following are provided in case they have not yet been defined:

```
2 \providecommand*{\fnbelowfloat}{}
3 \providecommand*{\fnunderfloat}{}
4 \providecommand*{\setbaselinefloat}{}
5 \providecommand*{\setbaselinefixed}{}
```

Nullified for HTML:

```
6 \renewcommand*{\fnbelowfloat}{}
7 \renewcommand*{\fnunderfloat}{}
8 \renewcommand*{\setbaselinefloat}{}
9 \renewcommand*{\setbaselinefixed}{}
```

File 477 **`lwarp-struktex.sty`**

§ 589 Package # struktex

(*Emulates or patches code by* JOBST HOFFMANN.)

struktex (*Pkg*) struktex is patched for use by lwarp.

for HTML output: `1 \LWR@ProvidesPackagePass{struktex}`

```
2 \BeforeBeginEnvironment{struktogramm}{%
3     \begin{lateximage}[-struktex-~\PackageDiagramAltText]?%
4 }
5 \AfterEndEnvironment{struktogramm}{\end{lateximage}}
6
7 \newenvironment{LWR@HTML@centernss}{\begin{center}}{\end{center}}
8 \LWR@formattedenv{centernss}
9
10 \newcommand{\LWR@HTML@CenterNssFile}[1]{%
11     \begin{center}
12     \input{#1.nss}
13     \end{center}
14 }
15 \LWR@formatted{CenterNssFile}
16
17 \newcommand{\LWR@HTML@centernssfile}{\LWR@HTML@CenterNssFile}
18 \LWR@formatted{centernssfile}
```

File 478 **`lwarp-subcaption.sty`**

§ 590 Package # subcaption

(*Emulates or patches code by* AXEL SOMMERFELDT.)

subcaption (*Pkg*) subcaption is patched for use by lwarp.

for HTML output: `1 \LWR@ProvidesPackagePass{subcaption}[2018/05/01]`

Tells lwarp to ignore minipage widths inside a subfigure or subtable. In print mode the minipages are used to place the items next to each other. In HTML they are placed side-by-side automatically.

```
2 \xpretocmd{\subcaption@iiminipage}
3     {\minipagefullwidth}
4     {}
5     {\LWR@patcherror{subcaption}{subcaption@iiminipage}}
```

Likewise for a \subcaptionbox:

```
6 \xpretocmd{\subcaptionbox}
7     {\minipagefullwidth}
8     {}
9     {\LWR@patcherror{subcaption}{subcaptionbox}}
```

File 479   **`lwarp-subfig.sty`**

§ 591   Package   # subfig

(*Emulates or patches code by* STEVEN DOUGLAS COCHRAN.)

subfig (*Pkg*)   subfig is supported and patched by lwarp.

⚠ table numbering   To have correct sub table numbers:

```
\usepackage{caption}
\captionsetup[table]{position=top}
```

⚠ lof/lotdepth   At present, the package options for lofdepth and lotdepth are not working. These
counters must be set separately after the package has been loaded.

⚠ horizontal spacing   In the document source, use \hfill and \hspace* between subfigures to spread
them apart horizontally. The use of other forms of whitespace may cause para-
graph tags to be generated, resulting in subfigures appearing on the following lines
instead of all on a single line.

for HTML output: Accept all options for lwarp-subfig:

```
1 \LWR@ProvidesPackagePass{subfig}[2005/06/28]
```

\sf@@@subfloat   {⟨*1 type*⟩} [⟨*2 lof entry*⟩] [⟨*3 caption*⟩] {⟨*4 contents*⟩}

The outer minipage allows side-by-side subfloats with \hfill between.

```
2 \VerifyCommand[lwarp][subfig]{\sf@@@subfloat}{B29FEC2418FD15B9E58ACF593B81BA93}
3
4 \long\def\sf@@@subfloat#1[#2][#3]#4{%
5 \begin{minipage}{\linewidth}% lwarp

6 \IfValueTF{#2}{%
7     \LWR@setlatestname{#2}%
8 }{%
9     \IfValueTF{#3}{%
10        \LWR@setlatestname{#3}%
11    }{}%
12 }%
13 \LWR@stoppars% lwarp
14     \@ifundefined{FBsc@max}{}%
15        {\FB@readaux{\let\FBsuboheight\relax}}%
16    \@tempcnta=\@ne
17    \if@minipage
18      \@tempcnta=\z@
19    \else\ifdim \lastskip=\z@ \else
20      \@tempcnta=\tw@
21    \fi\fi
22    \ifmaincaptiontop
23      \sf@top=\sf@nearskip
24      \sf@bottom=\sf@farskip
```

```
25      \else
26        \sf@top=\sf@farskip
27        \sf@bottom=\sf@nearskip
28      \fi
29      \leavevmode

30 %    \setbox\@tempboxa \hbox{#4}%
31 %    \@tempdima=\wd\@tempboxa
32 %    \@ifundefined{FBsc@max}{}%
33 %        {\global\advance\Xhsize-\wd\@tempboxa
34 %          \dimen@=\ht\@tempboxa
35 %          \advance\dimen@\dp\@tempboxa
36 %          \ifdim\dimen@>\FBso@max
37 %            \global\FBso@max\dimen@
38 %          \fi}%
```

Do not use boxes, which interfere with lateximages:

```
39 %      \vtop%
40      \bgroup
41 %        \vbox%
42        \bgroup
43          \ifcase\@tempcnta
44            \@minipagefalse
45          \or
46 %            \vskip\sf@top
47          \or
48            \ifdim \lastskip=\z@ \else
49 %              \@tempskipb\sf@top\relax\@xaddvskip
50            \fi
51          \fi
52          \sf@ifpositiontop{%
53            \ifx \@empty#3\relax \else
54              \sf@subcaption{#1}{#2}{#3}%
55 %              \vskip\sf@capskip
56 %              \vskip\sf@captopadj
57            \fi\egroup
58 %            \hrule width0pt height0pt depth0pt
59              \LWR@startpars% lwarp
60 %      \box\@tempboxa
61              #4
62              \LWR@stoppars% lwarp
63          }{%
64            \LWR@startpars% lwarp
65            \@ifundefined{FBsc@max}%
66              {
67 %      \box\@tempboxa
68                #4
69              }%
70              {\ifx\FBsuboheight\relax
71 %                \box\@tempboxa
72                #4
73              \else
74 %                \vbox to \FBsuboheight{\FBafil\box\@tempboxa\FBbfil}%
75                #4
76              \fi}%
77            \LWR@stoppars% lwarp
78            \egroup
79            \ifx \@empty#3\relax \else
80 %              \vskip\sf@capskip
81 %                \hrule width0pt height0pt depth0pt
```

```
82              \sf@subcaption{#1}{#2}{#3}%
83          \fi
84        }%
85 %      \vskip\sf@bottom
86    \egroup
87    \@ifundefined{FBsc@max}{}%
88        {\addtocounter{FRobj}{-1}%
89          \ifnum\c@FRobj=0\else
90            \subfloatrowsep
91          \fi}%
92    \ifmaincaptiontop\else
93      \global\advance\@nameuse{c@\@captype}\m@ne
94      \fi
95 \end{minipage}% lwarp
96 \LWR@startpars% lwarp
97    \endgroup\ignorespaces%
98 }%
```

\sf@subcaption   {⟨*1 type*⟩} {⟨*2 lof entry*⟩} {⟨*3 caption*⟩}

```
 99 \VerifyCommand[lwarp][subfig]{\sf@subcaption}{63123F93BADE8F3BBC127012A832A4C4}
100
101 \long\def\sf@subcaption#1#2#3{%
102 \LWR@stoppars% lwarp
103    \ifx \relax#2\relax \else
104      \bgroup
105        \let\label=\@gobble
106        \let\protect=\string
107        \def\@subcaplabel{%
108          \caption@lstfmt{\@nameuse{p@#1}}{\@nameuse{the#1}}}%
109        \sf@updatecaptionlist{#1}{#2}{\the\value{\@captype}}{\the\value{#1}}%
110      \egroup
111    \fi
112    \bgroup
113      \ifx \relax#3\relax
114        \let\captionlabelsep=\relax
115      \fi
116 %    \setbox0\vbox{%
117 %        \hb@xt@\the\@tempdima{%
118 %
119 % %          \hss
120 % %        \parbox[t]{\the\@tempdima}{%
121 %            \caption@make
122 %              {\@nameuse{sub\@captype name}}%
123 %              {\@nameuse{thesub\@captype}}%
124 %              {#3}
125 % %    }%
126 % %          \hss
127 %    }
128 %  }%
129      \@ifundefined{FBsc@max}%
130 %        {\box0}%
131        {
132 %    \parbox[t]{\the\@tempdima}{%
133 \LWR@traceinfo{sfsubcap B1}%     lwarp
134          \LWR@figcaption%       lwarp
135          \caption@make
136            {\@nameuse{sub\@captype name}}%
137            {\@nameuse{thesub\@captype}}%
138            {\LWR@isolate{#3}}%
```

```
139            \endLWR@figcaption%        lwarp
140 \LWR@traceinfo{sfsubcap B2}%        lwarp
141 %    }%
142       }%
143     {\dimen@\ht0%
144      \advance\dimen@\dp0%
145      \ifdim\dimen@>\FBsc@max
146        \global\FBsc@max\dimen@
147      \fi
148      \FB@readaux{\let\FBsubcheight\relax}%
149      \ifx\FBsubcheight\relax
150        \def\next{
151 %    \parbox[t]{\the\@tempdima}
152            }%
153      \else
154        \def\next{
155 %    \parbox[t][\FBsubcheight][t]{\the\@tempdima}
156            }%
157      \fi
158 %      \vbox{%
159 %        \hb@xt@\the\@tempdima{%
160
161 %          \hss
162 %          \next{%
163 \LWR@traceinfo{sfsubcap C1}% lwarp
164          \caption@make
165             {\@nameuse{sub\@captype name}}%
166             {\@nameuse{thesub\@captype}}%
167             {#3}
168 \LWR@traceinfo{sfsubcap C1}% lwarp
169 %    }%
170 %          \hss
171
172 %    }
173 %          }
174        }%
175   \egroup
176 \LWR@startpars% lwarp
177 }
```

\subfloat@label   Patches for \sf@sub@label:

```
178 \xpretocmd{\subfloat@label}
179    {\LWR@ensuredoingapar}
180    {}
181    {\LWR@patcherror{subfig}{subfloat@label}}
```

Patches for \subref.

\sf@subref   {⟨*label*⟩}

The unstarred version uses a \ref link whose printed text comes from the
sub@<label>:

```
182 \renewcommand{\sf@subref}[1]{%
183    \LWR@subnewref{#1}{sub@#1}%
184 }
```

\sf@@subref   {⟨*label*⟩}

The starred version uses the printed sub@<label> which is stored as if it were a page number:

185 \renewcommand{\sf@@subref}[1]{\LWR@orig@pageref{sub@#1}}

Defining new subfloats. The l@sub<type> for each is redefined.

\@newsubfloat   [⟨*keys/values*⟩] {⟨*float name*⟩}

186 \LetLtxMacro\LWR@orig@newsubfloat\@newsubfloat
187
188 \def\@newsubfloat[#1]#2{%
189 \LWR@orig@newsubfloat[#1]{#2}%
190 \renewcommand{\l@sub#2}[2]{\hypertocfloat{2}{sub#2}{\ext@sub#2}{##1}{##2}}%
191 }

Pre-defined for figures and tables:

\l@subfigure   {⟨*text*⟩} {⟨*pagenum*⟩}

192 \renewcommand{\l@subfigure}[2]{\hypertocfloat{2}{subfigure}{lof}{#1}{#2}}

\l@subtable   {⟨*text*⟩} {⟨*pagenum*⟩}

193 \renewcommand{\l@subtable}[2]{\hypertocfloat{2}{subtable}{lot}{#1}{#2}}

File 480   **lwarp-subfigure.sty**

§ 592   Package   # subfigure

subfigure (*Pkg*)   subfigure is emulated by subfig.

for HTML output:
1 \LWR@ProvidesPackageDrop{subfigure}[2002/03/15]
2 \RequirePackage{subfig}

3 \LetLtxMacro\subfigure\subfloat
4 \LetLtxMacro\subtable\subfloat
5 \LetLtxMacro\Subref\subref
6 \@ifundefined{figuretopcaptrue}{\newif\iffiguretopcap}{}
7 \newif\ifsubfiguretopcap
8 \newif\ifsubcaphang
9 \newif\ifsubcapcenter
10 \newif\ifsubcapcenterlast
11 \newif\ifsubcapnooneline
12 \newif\ifsubcapraggedright
13 \newskip\subfigtopskip
14 \newskip\subfigcapskip
15 \newdimen\subfigcaptopadj
16 \newskip\subfigbottomskip
17 \newdimen\subfigcapmargin
18 \newskip\subfiglabelskip
19 \newcommand*{\subcapsize}{}
20 \newcommand*{\subcaplabelfont}{}
21 \newcommand*{\subcapfont}{}

File 481 **`lwarp-subsupscripts.sty`**

§ 593   Package **subsupscripts**

(*Emulates or patches code by* Riccardo Bresciani.)

subsupscripts (*Pkg*)   subsupscripts is used as-is for svg math, and is emulated for MathJax.

for HTML output:
```
1 \LWR@ProvidesPackagePass{subsupscripts}[2009/10/27]
```

The larger skips are used here.

```
2 \begin{warpMathJax}
3 \CustomizeMathJax{%
4     \newcommand{\fourscriptsC}[7]{%
5         {}^{#2}_{#3}\hspace{#6}#1\hspace{#7}{}^{#4}_{#5}%
6     }
7 }
8 \CustomizeMathJax{%
9     \newcommand{\lrsubscriptsC}[5]{%
10        \fourscriptsC{#1}{}{#2}{}{#3}{#4}{#5}%
11    }
12 }
13 \CustomizeMathJax{%
14    \newcommand{\lrsuperscriptsC}[5]{%
15        \fourscriptsC{#1}{#2}{}{#3}{}{#4}{#5}%
16    }
17 }
18 \CustomizeMathJax{%
19    \newcommand{\fourscripts}[5]{%
20        \fourscriptsC{#1}{#2}{#3}{#4}{#5}{0ex}{0ex}%
21    }
22 }
23 \CustomizeMathJax{%
24    \newcommand{\lrsubscripts}[3]{\fourscripts{#1}{}{#2}{}{#3}}
25 }
26 \CustomizeMathJax{%
27    \newcommand{\lrsuperscripts}[3]{\fourscripts{#1}{#2}{}{#3}{}}
28 }
29 \CustomizeMathJax{%
30    \newcommand{\twolscripts}[4][-.16ex]{{}^{#3}_{#4}\hspace{#1}#2}
31 }
32 \CustomizeMathJax{%
33    \newcommand{\tworscripts}[4][-.07ex]{#2\hspace{#1}{}^{#3}_{#4}}
34 }
35 \CustomizeMathJax{%
36    \newcommand{\lsubscript}[3][-.16ex]{\twolscripts[#1]{#2}{}{#3}}
37 }
38 \CustomizeMathJax{%
39    \newcommand{\lsuperscript}[3][-.16ex]{\twolscripts[#1]{#2}{#3}{}}
40 }
41 \CustomizeMathJax{%
42    \newcommand{\rsubscript}[3][-.07ex]{\tworscripts[#1]{#2}{}{#3}}
43 }
44 \CustomizeMathJax{%
45    \newcommand{\rsuperscript}[3][-.07ex]{\tworscripts[#1]{#2}{#3}{}}
```

46 }
47 \end{warpMathJax}

File 482 **`lwarp-supertabular.sty`**

§ 594   Package   # supertabular

(*Emulates or patches code by* Johannes Braams, Theo Jurriens.)

supertabular (*Pkg*)   supertabular is emulated.

for HTML output: 1 \LWR@ProvidesPackageDrop{supertabular}[2004/02/20]

⚠ Misplaced alignment   For \tablefirsthead, etc., enclose them as follows:
tab character &

> \StartDefiningTabulars
> \tablefirsthead
> . . .
> \StopDefiningTabulars

See section 8.10.1.

⚠   lateximage   supertabular and xtab are not supported inside a lateximage.

2 \newcommand{\LWRST@firsthead}{}
3
4 \newcommand{\tablefirsthead}[1]{%
5   \long\gdef\LWRST@firsthead{#1}%
6 }
7
8 \newcommand{\tablehead}[1]{}
9 \newcommand{\tabletail}[1]{}
10
11 \newcommand{\LWRST@lasttail}{}
12
13 \newcommand{\tablelasttail}[1]{%
14   \long\gdef\LWRST@lasttail{#1}%
15 }

16 \newcommand{\tablecaption}[2][]{%
17   \long\gdef\LWRST@caption{%
18   \ifblank{#1}%
19   {\caption{#2}}%
20   {\caption[#1]{#2}}%
21   }%
22 }
23
24 \let\topcaption\tablecaption
25 \let\bottomcaption\tablecaption

26 \newcommand*{\LWRST@caption}{}
27
28 \newcommand*{\shrinkheight}[1]{}
29
30 \NewDocumentEnvironment{supertabular}{s o m}
31 {%

```
32 \LWR@traceinfo{supertabular}%
33 \begin{table}%
34 \LWRST@caption%
35 \begin{tabular}{#3}%
36 \TabularMacro\ifdefvoid{\LWRST@firsthead}%
37 {\LWR@getmynexttoken}%
38 {\expandafter\LWR@getmynexttoken\LWRST@firsthead}%
39 }%
40 {%
41 \ifdefvoid{\LWRST@lasttail}%
42 {}%
43 {%
44 \TabularMacro\ResumeTabular%
45 \LWRST@lasttail%
46 }%
47 \end{tabular}%
48 \end{table}%


49 \gdef\LWRST@caption{}%


50 \LWR@traceinfo{supertabular done}%
51 }
52
53 \NewDocumentEnvironment{mpsupertabular}{s o m}
54 {\minipage{\linewidth}\supertabular{#3}}
55 {\endsupertabular\endminipage}
```

File 483 **lwarp-svg.sty**

§ 595   Package   **svg**

(Emulates or patches code by PHILIP ILTEN, FALK HANISCH.*)*

svg (*Pkg*)   svg is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{svg}[2020/10/23]

```
2 \xpretocmd\includesvg%
3     {\begin{lateximage}}%
4     {}%
5     {\LWR@patcherror{svg}{includesvg}}
6
7 \xapptocmd\includesvg%
8     {\end{lateximage}}%
9     {}%
10    {\LWR@patcherror{svg}{includesvg}}
11
12 \xpretocmd\includeinkscape%
13    {\begin{lateximage}}%
14    {}%
15    {\LWR@patcherror{svg}{includeinkscape}}
16
17 \xapptocmd\includeinkscape%
18    {\end{lateximage}}%
19    {}%
20    {\LWR@patcherror{svg}{includeinkscape}}
```

File 484 **`lwarp-swfigure.sty`**

§ 596 Package # swfigure

(*Emulates or patches code by* Claudio Beccari.)

swfigure (*Pkg*) swfigure is emulated.

for HTML output: 1 `\LWR@ProvidesPackageDrop{swfigure}[2020-11-10]`

```
2 \NewDocumentEnvironment{DFimage}%
3   {O{SW} m O{#4} m o D(){0.8} D<>{0} D||{0.25} D!!{}}%
4 {%
5     \begin{figure}
6     \centering
7     \includegraphics{#2}
8     \caption[#3]{#4}
9     \IfValueT{#5}{\label{#5}}
10    \end{figure}
11 }%
12 {}%
```

File 485 **`lwarp-sympytex.sty`**

§ 597 Package # sympytex

(*Emulates or patches code by* Tim Molteno.)

sympytex (*Pkg*) sympytex is patched for use by lwarp.

for HTML output: 1 `\LWR@ProvidesPackagePass{sympytex}[2014/05/16]`

```
2 \AfterEndPreamble{
3
4 \AtBeginEnvironment{sympyblock}{%
5     \ifnumcomp{\value{LWR@leximagedepth}}{>}{0}%
6         {}%
7         {%
8             \LWR@forcenewpage%
9             \LWR@atbeginverbatim{verbatim}%
10        }%
11 }
12
13 \AfterEndEnvironment{sympyblock}{%
14     \ifnumcomp{\value{LWR@leximagedepth}}{>}{0}%
15        {}%
16        {%
17            \LWR@afterendverbatim%
18        }%
19 }
20
21 }
```

File 486 **`lwarp-syntonly.sty`**

§ 598   Package **syntonly**

(*Emulates or patches code by* FRANK MITTELBACH, RAINER SCHÖPF.)

syntonly (*Pkg*)   syntonly is ignored.

for HTML output: Discard all options for lwarp-syntonly:

```
1 \LWR@ProvidesPackageDrop{syntonly}[2017/06/30]

2 \newif\ifsyntax@
3 \syntax@false
4
5 \newcommand*{\syntaxonly}{}
6
7 \@onlypreamble\syntaxonly

8 \def\nopages@{}
```

File 487 **`lwarp-tabfigures.sty`**

§ 599   Package **tabfigures**

tabfigures (*Pkg*)   tabfigures is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{tabfigures}[2012/01/24]`

File 488 **`lwarp-tablefootnote.sty`**

§ 600   Package **tablefootnote**

tablefootnote (*Pkg*)   tablefootnote is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{tablefootnote}[2014/01/26]`

This works because in HTML tables are no longer floats.

`2 \LetLtxMacro\tablefootnote\footnote`

File 489 **`lwarp-tabls.sty`**

§ 601   Package **tabls**

(*Emulates or patches code by* DONALD ARSENEAU.)

tabls (*Pkg*)   tabls is emulated. \LWR@hline is used to handle the optional argument when tabls is loaded.

for HTML output:

```
1 \LWR@ProvidesPackageDrop{tabls}
```

```
2 \newdimen\tablinesep
3 \newdimen\arraylinesep
4 \newdimen\extrarulesep
```

File 490   `lwarp-tabularx.sty`

§ 602   Package   # tabularx

(*Emulates or patches code by* DAVID CARLISLE.)

tabularx (*Pkg*)   tabularx is emulated by lwarp.

for HTML output: Discard all options for lwarp-tabularx:

```
1 \LWR@ProvidesPackageDrop{tabularx}[2016/02/03]
2 \RequirePackage{array}
```

\tabularxcolumn is ignored. All X columns will be p for now. The width is ignored.

```
3 \def\tabularxcolumn#1{p{#1}}
4 \newcolumntype{X}{p{1in}}
```

```
5 \DeclareDocumentEnvironment{tabularx}{m o m}
6     {\tabular{#3}}
7     {\endtabular}
8
9 \DeclareDocumentEnvironment{tabularx*}{m o m}
10     {\tabular{#3}}
11     {\endtabular}
```

File 491   `lwarp-tabulary.sty`

§ 603   Package   # tabulary

(*Emulates or patches code by* DAVID CARLISLE.)

tabulary (*Pkg*)   tabulary is emulated by lwarp.

for HTML output: Discard all options for lwarp-tabulary.

Column types L, C, R, and J are emulated by lwarp core code.

```
1 \LWR@ProvidesPackageDrop{tabulary}[2014/06/11]
2 \RequirePackage{array}
```

```
3 \NewDocumentEnvironment{tabulary}{m o m}
4 {\tabular{#3}}
5 {\endtabular}
```

```
6
7 \NewDocumentEnvironment{tabulary*}{m o m}
8 {\tabular{#3}}
9 {\endtabular}
```

```
10 \newcolumntype{L}{l}
11 \newcolumntype{C}{c}
12 \newcolumntype{R}{r}
13 \newcolumntype{J}{l}
```

```
14 \newdimen\tymin
15 \newdimen\tymax
16 \def\tyformat{}
```

File 492   **lwarp-tagpdf.sty**

§ 604   Package   **tagpdf**

tagpdf (*Pkg*)   tagpdf adds alt text, for images only. (HTML only has alternate text for images.)

The overall strategy is that tagpdf is deactivated, and slightly patched to process alt tags. Also see tagpdf-base, tagpdf-mc-code-generic, and tagpdf-mc-code-lua, following tagpdf.

for HTML output:
```
1 \RequirePackage{tagpdf-base}% *88*
2 \LWR@ProvidesPackagePass{tagpdf}[2022-08-24]
```

```
3 \ExplSyntaxOn
4
5 \keys_define:nn { __tag / struct }
6   {
7     alt .code:n      = % Alt property
8       {
9 %        \str_set_convert:Noon
10 %          \l__tag_tmpa_str
11 %          { #1 }
12 %          { default }
13 %          { utf16/hex }
14 %        \__tag_prop_gput:cnx
15 %          { g__tag_struct_\int_eval:n {\c@g__tag_struct_abs_int}_prop }
16 %          { Alt }
17 %          { <\l__tag_tmpa_str> }
18         \gdef\LWR@ThisAltText{\detokenize\expandafter{#1}}%        lwarp
19       },
20   }
21
22 \ExplSyntaxOff
```

The package is deactivated on load, and also each time \tagpdfsetup is used.

```
23 \LWR@tagpdf@deactivate
```

File 493   **lwarp-tagpdf-base.sty**

§ 605   Package   **tagpdf-base**

(*Emulates or patches code by* Ulrike Fischer.)

tagpdf-base (*Pkg*)   tagpdf-base is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{tagpdf-base}[2022-08-24]

```
2 \ExplSyntaxOn
3
4 \newcommand*{\LWR@tagpdf@deactivate}{
5     \keys_set:nn { __tag / setup } {
6         activate/spaces = false ,
7         activate/mc = false ,
8         activate/tree = false ,
9         activate/struct = false ,
10        activate/struct-dest = false
11    }
12 }
13
14 \RenewDocumentCommand \tagpdfsetup { m }{
15    \keys_set:nn { __tag / setup } { #1 }
16    \LWR@tagpdf@deactivate
17 }
18
19 \cs_set_protected:Npn \__tag_whatsits: {}% *88* new
20
21 % *88*
22 %\RenewDocumentCommand \tagmcbegin { m }
23 %  {
24 %%    \tag_mc_begin:n {#1}
25 %    \keys_set:nn { __tag / mc } {#1}
26 %  }
27
28 % *88*
29 %\RenewDocumentCommand \tagmcend {   }
30 %  {
31 %%    \tag_mc_end:
32 %    \ThisAltText{}%      lwarp
33 %  }
34 \cs_set_protected:Nn \tag_mc_end:{% *8* probably not needed because whatsits disabled above
35 %    \__tag_whatsits:
36    \ThisAltText{}%      lwarp
37 }
38
39 %\RenewDocumentCommand \tagmcuse { m }
40 %  {
41 %%    \tag_mc_use:n {#1}
42 %  }
43
44 %\RenewDocumentCommand \tagstructbegin { m }
45 %  {
46 %% *88*   \keys_set:nn { __tag / struct} { #1 }%       lwarp
47 %    \tag_struct_begin:n {#1}% *88* was disabled
```

```
48 %  }
49
50 %\RenewDocumentCommand \tagstructend {   }
51 %  {
52 %%   \tag_struct_end:
53 %  \ThisAltText{}%          lwarp
54 %  }
55 \cs_set_protected:Npn \tag_struct_end:{% *88* maybe not needed
56   \ThisAltText{}%          lwarp
57 }
58
59 %\RenewDocumentCommand \tagstructuse { m }
60 %  {
61 %    \tag_struct_use:n {#1}% *88* was disabled
62 %  }
63
64 \ExplSyntaxOff
```

File 494   **`lwarp-tagpdf-mc-code-generic.sty`**

§ 606   Package   # tagpdf-mc-code-generic

(*Emulates or patches code by* ULRIKE FISCHER.)

tagpdf-mc-code-generic (*Pkg*)   tagpdf-mc-code-generic is patched for use by lwarp.

for HTML output:
```
1 \LWR@ProvidesPackagePass{tagpdf-mc-code-generic}[2022-08-24]

2 \ExplSyntaxOn
3
4 % From tagpdf-mc-code-generic.sty:
5 \keys_define:nn { __tag / mc }
6   {
7     tag .code:n = % the name (H,P,Span) etc
8       {
9 %       \tl_set:Ne   \l__tag_mc_key_tag_tl { #1 }
10 %       \tl_gset:Ne  \g__tag_mc_key_tag_tl { #1 }
11      },
12    raw  .code:n =
13      {
14 %       \tl_put_right:Ne \l__tag_mc_key_properties_tl { #1 }
15      },
16    alt .code:n      = % Alt property
17      {
18 %       \str_set_convert:Noon
19 %         \l__tag_tmpa_str
20 %         { #1 }
21 %         { default }
22 %         { utf16/hex }
23 %       \tl_put_right:Nn \l__tag_mc_key_properties_tl { /Alt~< }
24 %       \tl_put_right:No \l__tag_mc_key_properties_tl { \l__tag_tmpa_str>~ }
25        \gdef\LWR@ThisAltText{\detokenize\expandafter{#1}}%          lwarp
26      },
27    alttext .meta:n = {alt=#1},
28    actualtext .code:n       = % ActualText property
29      {
30 %       \tl_if_empty:oF{#1}
```

```
31 %          {
32 %             \str_set_convert:Noon
33 %                \l__tag_tmpa_str
34 %                { #1 }
35 %                { default }
36 %                { utf16/hex }
37 %             \tl_put_right:Nn \l__tag_mc_key_properties_tl { /ActualText~< }
38 %          \tl_put_right:No \l__tag_mc_key_properties_tl { \l__tag_tmpa_str>~ }
39 %             }
40      },
41    label .tl_set:N        = \l__tag_mc_key_label_tl,
42    artifact .code:n       =
43      {
44 %        \exp_args:Nne
45 %           \keys_set:nn
46 %             { __tag / mc }
47 %             { __artifact-bool, __artifact-type=#1 }
48      },
49    artifact .default:n    = {notype}
50  }
51
52 \ExplSyntaxOff
```

File 495   **lwarp-tagpdf-mc-code-lua.sty**

§ 607   Package   **tagpdf-mc-code-lua**

(*Emulates or patches code by* ULRIKE FISCHER.)

tagpdf-mc-code-lua (*Pkg*)   tagpdf-mc-code-lua is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{tagpdf-mc-code-lua}[2022-08-24]

```
2 \ExplSyntaxOn
3
4 \keys_define:nn { __tag / mc }
5  {
6    tag .code:n = %
7      {
8 %        \tl_set:Ne   \l__tag_mc_key_tag_tl { #1 }
9 %        \tl_gset:Ne  \g__tag_mc_key_tag_tl { #1 }
10 %        \lua_now:e
11 %          {
12 %            ltx.__tag.func.store_mc_data(\__tag_get_mc_abs_cnt:,"tag","#1")
13 %          }
14      },
15    raw .code:n =
16      {
17 %        \tl_put_right:Ne \l__tag_mc_key_properties_tl { #1 }
18 %        \lua_now:e
19 %          {
20 %            ltx.__tag.func.store_mc_data(\__tag_get_mc_abs_cnt:,"raw","#1")
21 %          }
22      },
23    alt .code:n       = % Alt property
24      {
25 %        \str_set_convert:Noon
```

```
26 %              \l__tag_tmpa_str
27 %              { #1 }
28 %              { default }
29 %              { utf16/hex }
30 %          \tl_put_right:Nn \l__tag_mc_key_properties_tl { /Alt~< }
31 %         \tl_put_right:No \l__tag_mc_key_properties_tl { \l__tag_tmpa_str>~ }
32 %           \lua_now:e
33 %             {
34 %               ltx.__tag.func.store_mc_data
35 %                 (
36 %                \__tag_get_mc_abs_cnt:,"alt","/Alt~<\str_use:N \l__tag_tmpa_str>"
37 %                 )
38 %             }
39        \gdef\LWR@ThisAltText{\detokenize\expandafter{#1}}%        lwarp
40       },
41     alttext .meta:n = {alt=#1},
42     actualtext .code:n      = % Alt property
43       {
44 %         \tl_if_empty:oF{#1}
45 %           {
46 %               \str_set_convert:Noon
47 %                 \l__tag_tmpa_str
48 %                 { #1 }
49 %                 { default }
50 %                 { utf16/hex }
51 %             \tl_put_right:Nn \l__tag_mc_key_properties_tl { /Alt~< }
52 %          \tl_put_right:No \l__tag_mc_key_properties_tl { \l__tag_tmpa_str>~ }
53 %             \lua_now:e
54 %               {
55 %                 ltx.__tag.func.store_mc_data
56 %                   (
57 %                     \__tag_get_mc_abs_cnt:,
58 %                   "actualtext",
59 %                   "/ActualText~<\str_use:N \l__tag_tmpa_str>"
60 %                   )
61 %               }
62 %           }
63       },
64     label .code:n =
65       {
66 %         \tl_set:Nn\l__tag_mc_key_label_tl { #1 }
67 %         \lua_now:e
68 %           {
69 %             ltx.__tag.func.store_mc_data
70 %               (
71 %                 \__tag_get_mc_abs_cnt:,"label","#1"
72 %               )
73 %           }
74       },
75     __artifact-store .code:n =
76       {
77 %         \lua_now:e
78 %           {
79 %             ltx.__tag.func.store_mc_data
80 %               (
81 %                 \__tag_get_mc_abs_cnt:,"artifact","#1"
82 %               )
83 %           }
84       },
85     artifact .code:n        =
```

```
86        {
87 %        \exp_args:Nne
88 %          \keys_set:nn
89 %            { __tag / mc}
90 %            { __artifact-bool, __artifact-type=#1, tag=Artifact }
91 %        \exp_args:Nne
92 %          \keys_set:nn
93 %            { __tag / mc }
94 %            { __artifact-store=\l__tag_mc_artifact_type_tl }
95        },
96      artifact .default:n    = { notype }
97   }
98
99 \ExplSyntaxOff
```

File 496   `lwarp-tascmac.sty`

§ 608   Package   **tascmac**

tascmac (*Pkg*)   tascmac is emulated.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{tascmac}[2018/03/09]

2 \newenvironment*{boxnote}
3    {
4        \BlockClass[
5            padding: .5ex ;
6            border: 1px solid black ;
7            border-top: 1px dashed black ;
8        ]{boxnote}
9    }
10    {\endBlockClass}
11
12 \newenvironment*{screen}[1][]
13    {
14        \BlockClass[
15            padding: .5ex ;
16            border: 1px solid gray ;
17            border-radius: 8pt
18        ]{boxnote}
19    }
20    {\endBlockClass}
21
22 \newenvironment*{itembox}[2][]
23    {
24        \BlockClass[
25            padding: .5ex ;
26            border: 1px solid gray ;
27            border-radius: 8pt
28        ]{boxnote}
29        \InlineClass{itemboxtitle}{#2}\par
30    }
31    {\endBlockClass}
32
33 \newenvironment*{shadebox}
34    {
35        \BlockClass[
```

```
36              padding: .5ex ;
37              border: 1px solid black ;
38              box-shadow: 3px 3px 3px \#808080 ;
39        ]{boxnote}
40      }
41    {\endBlockClass}
42
43 \newcommand*{\mask}[2]{%
44    \InlineClass[background: lightgray]{mask}{#1}%
45 }
46
47 \newcommand*{\maskbox}[5]{%
48    \InlineClass[background: lightgray]{mask}{#5}%
49 }
50
51 \newcommand*{\Maskbox}[6]{%
52    \InlineClass[
53        background: lightgray ;
54        border: #5 solid black
55    ]{mask}{#6}%
56 }
57
58 \newcommand*{\keytop}[2][]{%
59    \InlineClass[%
60        padding: .2ex ;
61        border: 1px solid black ;
62        border-radius: .7ex ;
63    ]{keytop}{#2}%
64 }
65
66 \def\yen{\HTMLunicode{00A5}}
67
68 \def\return{\HTMLunicode{23CE}}
69
70 \def\Return{\HTMLunicode{23CE}}
71
72 \def\ascii{ASCII Corporation}
73
74 \def\Ascii{ASCII Corporation}
75
76 \def\ASCII{ASCII Corporation}
```

File 497   **`lwarp-tcolorbox.sty`**

§ 609   Package   **tcolorbox**

(*Emulates or patches code by* Thomas F. Sturm.)

tcolorbox (*Pkg*)   tcolorbox is patched for use by lwarp.

See section 8.3.8 for limitations.

for HTML output: 1 \LWR@ProvidesPackagePass{tcolorbox}[2023/06/19]

2 \newbool{LWR@havetcblower}
3 \boolfalse{LWR@havetcblower}

Colors are supported via HTML styles:

```
4 \newcommand{\LWR@tcolorbox@findcolors}{%
5     \convertcolorspec{named}{tcbcolback}{HTML}\LWR@tcbcolback
6     \convertcolorspec{named}{tcbcolframe}{HTML}\LWR@tcbcolframe
7     \iftcb@titlefilled%
8         \convertcolorspec{named}{tcbcolbacktitle}{HTML}\LWR@tcbcolbacktitle
9     \else
10         \convertcolorspec{named}{tcbcolframe}{HTML}\LWR@tcbcolbacktitle
11    \fi
12    \convertcolorspec{named}{tcbcoltitle}{HTML}\LWR@tcbcoltitle
13    \convertcolorspec{named}{tcbcolupper}{HTML}\LWR@tcbcolupper
14    \convertcolorspec{named}{tcbcollower}{HTML}\LWR@tcbcollower
15 }
16
17 \newcommand*{\LWR@tcolorbox@titlecolorstyles}{%
18    border-top: 1px solid \LWR@origpound\LWR@tcbcolframe ;
19    border-bottom: 1px solid \LWR@origpound\LWR@tcbcolframe ;
20    background: \LWR@origpound\LWR@tcbcolbacktitle ;
21    color: \LWR@origpound\LWR@tcbcoltitle ;
22 }
```

The title is placed inside its own <div> of class tcolorboxtitle.

```
23 \newcommand*{\LWR@showtitle@}[1]{%
24     \begin{BlockClass}[
25         \LWR@tcolorbox@titlecolorstyles
26     ]{tcolorboxtitle}
27 %                     \cmdKV@LWRtcolorbox@title\par
28     \kvtcb@before@title#1\kvtcb@after@title
29     \end{BlockClass}
30 }
```

If no title, a non-breakable space is used to take some vertical space.

```
31 \newcommand*{\LWR@showtitle}[1]{%
32     \iftcb@titlevisible
33     \LWR@showtitle@{#1}
34     \else
35     \LWR@showtitle@{~}
36     \fi
37 }
38
39 \newcommand*{\LWR@tcolorbox@dophantom}{%
40 %       \sbox\tcb@phantombox{\kvtcb@phantom}%
41 %       \iftcb@hasPhantom%
42 %           \box\tcb@phantombox%
43 %           \tcb@hasPhantomfalse%
44 %       \fi%
45     \kvtcb@phantom
46     \let\kvtcb@phantom\@empty%
47 }
```

The tcolorbox is placed inside an external <div> of class #1, which is tcolorbox or tcolorbox inlineminipage. The upper and lower parts are placed into their own internal <div>s of class tcolorboxupper and tcolorboxlower.

```
48 \newcommand*{\LWR@tcolorboxstart}[1]{
49     \LWR@tcolorbox@findcolors
```

```
50      \begin{BlockClass}[
51          border: 1px solid \LWR@origpound\LWR@tcbcolframe ;
52          background: \LWR@origpound\LWR@tcbcolback ;
53      ]{#1}
54      \LWR@tcolorbox@dophantom%
55      \ifdefvoid{\kvtcb@title}
56          {}
57          {
58              \LWR@showtitle{\kvtcb@title}
59          }
60      \begin{BlockClass}[
61          color: \LWR@origpound\LWR@tcbcolupper ;
62      ]{tcolorboxupper}
63 }
```

Floats enclose the tcolorbox.

```
64 \newcommand*{\LWR@tcolorbox@dostartfloat}{%
65      \ifx\kvtcb@float\@empty%
66 %        \tcb@set@normal@unbroken@beforeafter%
67      \else%
68 %        \edef\tcb@before@unbroken{%
69 %            \noexpand\tcb@float@env@begin{tcbfloat}[\kvtcb@float]%
70 %            \noexpand\kvtcb@everyfloat%
71 %        }%
72 %        \let\tcb@after@unbroken=\tcb@float@env@end%
73          \tcb@float@env@begin{tcbfloat}[\kvtcb@float]
74          \noexpand\kvtcb@everyfloat
75      \fi%
76 }
77
78 \newcommand*{\LWR@tcolorbox@doendfloat}{%
79      \ifx\kvtcb@float\@empty%
80      \else%
81          \tcb@float@env@end%
82      \fi%
83 }
```

Footnotes are handled via the main footnote mechanism, and pending notes are printed before and after each tcolorbox. Footnote numbering will not match the print output.

Not using \VerifyCommand here because tcolorbox changes meaning.

```
84 \renewenvironment{tcolorbox}[1][]
85      {
86          \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
87          {
88              \PackageError{lwarp}
89                  {%
90              Lwarp cannot process a tcolorbox inside a lateximage\MessageBreak
91                      or SVG math.\MessageBreak
92                      Enter 'H' for possible solutions%
93                  }
94                  {%
95                      Use \protect\tcbox, \protect\tcboxmath, or
96                      \protect\tcbhighmath\space instead.\MessageBreak%
97                      (Inside math, you probably want to use these anyhow.)%
98                  }
99          }{\relax}
```

```
100        \LWR@printpendingfootnotes
101        \tcb@layer@inc
102        \tcb@apply@box@options{#1}
103        \LWR@tcolorbox@dostartfloat%
104 %        \tcbset{title=,#1}
105        \boolfalse{LWR@havetcblower}
106        \LWR@tcolorboxstart{tcolorbox}
107        \tcb@insert@before@upper%
108    }
109    {
110        \ifbool{LWR@havetcblower}{%
111            \tcb@insert@after@lower%
112        }{%
113            \tcb@insert@after@upper%
114        }%
115        \end{BlockClass}
116        \LWR@printpendingfootnotes
117        \tcb@layer@dec
118        \end{BlockClass}
119        \LWR@tcolorbox@doendfloat%
120    }
```

For the lower part, the upper part is finished then the lower is started.

\tcblower is \let to \tcb@error@nolower globally, which gives an error in print mode, but is redefined here for HTML.

```
121 \newcommand{\LWR@HTML@tcb@error@nolower}{%
122    \tcb@insert@after@upper%
123    \end{BlockClass}
124    \begin{BlockClass}[%
125        border-top: 1px dashed \LWR@origpound\LWR@tcbcolframe ;
126        color: \LWR@origpound\LWR@tcbcollower ;
127    ]{tcolorboxlower}
128    \tcb@insert@before@lower%
129 }
130 \LWR@formatted{tcb@error@nolower}
```

Starred and unstarred \tcbline are simple \hrules.

```
131 \AtBeginDocument{
132 \ifdef{\tcbline}{
133    \newcommand*{\LWR@sub@tcbline}{%
134        \begin{BlockClass}{hrule}
135        \end{BlockClass}
136    }
137    \newcommand{\LWR@HTML@tcbline}{\@ifstar\LWR@sub@tcbline\LWR@sub@tcbline}
138    \LWR@formatted{tcbline}
139 }{}
140 }
141
142 \newcommand{\LWR@HTML@tcbox}[2][]{
143    \LWR@printpendingfootnotes
144    \LWR@tcolorbox@dostartfloat%
145    \begingroup
146    \tcb@layer@inc
147    \tcb@apply@box@options{#1}
148 %        \tcbset{title=,#1}
149    \boolfalse{LWR@havetcblower}
150    \LWR@tcolorboxstart{tcolorbox inlineminipage}
```

```
151     \tcb@insert@before@upper%
152     #2
153     \ifbool{LWR@havetcblower}{%
154         \tcb@insert@after@lower%
155     }{%
156         \tcb@insert@after@upper%
157     }%
158     \end{BlockClass}
159     \LWR@printpendingfootnotes
160     \end{BlockClass}
161     \tcb@layer@dec%
162     \endgroup%
163     \LWR@tcolorbox@dostartfloat%
164     \global\booltrue{LWR@minipagethispar}%
165 }
166 \LWR@formatted{tcbox}
167
168 \appto\LWR@restoreMathJaxformatting{%
169     \renewcommand{\tcbox}[2][]{#2}%
170 }
```

Patches for the subtitle, which is placed inside a `<div>` of class `tcolorboxsubtitle`.

```
171 \xpatchcmd{\tcbsubtitle}
172     {\begingroup}
173     {\begingroup\let\kvtcb@title\relax\begin{BlockClass}{tcolorboxsubtitle}}
174     {}
175     {\LWR@patcherror{tcolorbox}{tcbsubtitle}}
176
177 \xpatchcmd{\tcbsubtitle}
178     {\endgroup}
179     {\end{BlockClass}\endgroup}
180     {}
181     {\LWR@patcherror{tcolorbox}{tcbsubtitleB}}
```

`\tcboxfit` is the same as `\tcbox`.

```
182 \AtBeginDocument{
183     \ifdef{\tcboxfit}{%
184         \let\LWR@HTML@tcboxfit\tcbox%
185         \LWR@formatted{tcboxfit}
186     }{}
187 }
```

`\tcbtitle` is patched to support the text font.

```
188 \VerifyCommand[lwarp][tcolorbox]{\tcbtitle}{8C821A2BDC95C579A4FA340365D9A5CB}
189
190 \LetLtxMacro\LWR@HTML@tcbtitle\tcbtitle
191 \xpatchcmd{\LWR@HTML@tcbtitle}
192     {\tcb@insert@before@title\tcbtitletext}
193   {\tcb@insert@before@title\LWR@textcurrentfont{\LWR@textcurrentcolor{\tcbtitletext}}}
194     {}
195     {\LWR@patcherror{tcolorbox}{LWR@HTML@tcbtitle}}
196 \LWR@formatted{tcbtitle}
```

List-of:

```
197 \renewcommand*\l@tcolorbox[2]{\hypertocfloat{1}{tcolorbox}{lof}{#1}{#2}}
```

Theorem limitations. An error is printed if the document uses math, ams equation, etc. \tcboxmath and \tcbhighmath are ignored for HTML.

```
198 \AtBeginDocument{
199 \pgfkeysifdefined{/tcb/libload/theorems}{
200
201     \def\LWR@HTML@tcb@hack@amsmath{%
202         \PackageError{lwarp}
203             {%
204                 tcolorbox ``math'', ``ams equation'', and related\MessageBreak
205                     are not supported.\MessageBreak
206                     \protect\tcboxmath\space and
207                     \protect\tcbhighmath\space are emulated.\MessageBreak
208                     Enter 'H' for possible solutions%
209             }
210             {%
211                 Remove tcolorbox math-related options, and instead\MessageBreak
212                     use the usual math environments inside each tcolorbox.%
213             }
214     }
215     \LWR@formatted{tcb@hack@amsmath}
216
217     % Cause an error if using math:
218     \tcbset{%
219         math upper/.style={before upper*=\tcb@hack@amsmath,after upper*=$},%
220         math lower/.style={before lower*=\tcb@hack@amsmath,after lower*=$},%
221     }
222
223     \appto\LWR@restoreorigformatting{%
224     \tcbset{%
225         math upper/.style={before upper*=$\displaystyle,after upper*=$},%
226         math lower/.style={before lower*=$\displaystyle,after lower*=$},%
227     }%
228     }
229
230     \newcommand{\LWR@HTML@tcboxmath}[2][]{#2}
231     \LWR@formatted{tcboxmath}
232     \newcommand{\LWR@HTML@tcbhighmath}[2][]{#2}
233     \LWR@formatted{tcbhighmath}
234     \appto\LWR@restoreMathJaxformatting{%
235         \renewcommand{\tcboxmath}[2][]{#2}%
236         \renewcommand{\tcbhighmath}[2][]{#2}%
237     }
238 }{}% theorems loaded
239 }% AtBeginDocument
```

For MATHJAX:

```
240 \CustomizeMathJax{\newcommand{\tcbset}[1]{}}
241 \CustomizeMathJax{\newcommand{\tcbsetforeverylayer}[1]{}}
242 \CustomizeMathJax{\newcommand{\tcbox}[2][]{\boxed{\text{#2}}}}
243 \CustomizeMathJax{\newcommand{\tcboxfit}[2][]{\boxed{#2}}}
244 \CustomizeMathJax{\newcommand{\tcblower}{}}
245 \CustomizeMathJax{\newcommand{\tcbline}{}}
246 \CustomizeMathJax{\newcommand{\tcbtitle}{}}
247 \CustomizeMathJax{\newcommand{\tcbsubtitle[2][]{\mathrm{#2}}}}
248 \CustomizeMathJax{\newcommand{\tcboxmath}[2][]{\boxed{#2}}}
249 \CustomizeMathJax{\newcommand{\tcbhighmath}[2][]{\boxed{#2}}}
```

File 498   **`lwarp-tensor.sty`**

§ 610   Package   **tensor**

(*Emulates or patches code by* PHILIP G. RATCLIFFE.)

tensor (*Pkg*)   tensor is used as-is for SVG math, and is emulated for MATHJAX.

⚠ spacing   Compressed spacing and left justification are not possible with MATHJAX.

for HTML output: 1 `\LWR@ProvidesPackagePass{tensor}[2004/12/20]`

For MATHJAX. Special handling is required to parse the superscript and subscript arguments.

When a superscript or subscript is seen, it is processed and then the remainder is processesed recursively.

2 `\begin{warpMathJax}`
3 `\CustomizeMathJax{\def\LWRtensorindicesthreesub#1#2{{_{#2}}\LWRtensorindicesthree}}`
4 `\CustomizeMathJax{\def\LWRtensorindicesthreesup#1#2{{^{#2}}\LWRtensorindicesthree}}`

If not a superscript nor a subscript, processing stops.

5 `\CustomizeMathJax{\newcommand{\LWRtensorindicesthreenotsup}{}}`

Check ahead for a superscript or a subscript.

6 `\CustomizeMathJax{\newcommand{\LWRtensorindicesthreenotsub}{`
7 `    \ifnextchar ^ \LWRtensorindicesthreesup \LWRtensorindicesthreenotsup`
8 `}}`
9
10 `\CustomizeMathJax{\newcommand{\LWRtensorindicesthree}{`
11 `    \ifnextchar _ \LWRtensorindicesthreesub \LWRtensorindicesthreenotsub`
12 `}}`

Ignore star.

13 `\CustomizeMathJax{\newcommand{\LWRtensorindicestwo}{`
14 `    \ifstar\LWRtensorindicesthree\LWRtensorindicesthree`
15 `}}`

Remove the outer brace of the argument.

16 `\CustomizeMathJax{\newcommand{\indices}[1]{\LWRtensorindicestwo#1}}`

Attempting to use \vphantom here does not work:

17 `\CustomizeMathJax{\newcommand{\LWRtensortwo}[3][]{{}\indices{#1}{#2}\indices{#3}}}`

Ignore star.

18 `\CustomizeMathJax{\newcommand{\tensor}{\ifstar\LWRtensortwo\LWRtensortwo}}`

In text mode, \nuclide is converted to an svg image.

```
19 \CustomizeMathJax{%
20     \newcommand{\LWRnuclidetwo}[2][]{%
21         {%
22             \vphantom{\mathrm{#2}}%
23             {}^{\LWRtensornucleonnumber}_{#1}%
24             \mathrm{#2}%
25         }%
26     }%
27 }


28 \CustomizeMathJax{%
29     \newcommand{\nuclide}[1][]{%
30         \def\LWRtensornucleonnumber{#1}%
31         \LWRnuclidetwo%
32     }%
33 }
34 \end{warpMathJax}
```

File 499 **lwarp-termcal.sty**

§ 611 Package **termcal**

(*Emulates or patches code by* Bill Mitchell.)

termcal (*Pkg*) termcal is patched for use by lwarp.

for HTML output: `1 \LWR@ProvidesPackagePass{termcal}% questionable date in the .sty file`

Nullify the @ because everything is being done in a token list.

```
2 \xpatchcmd{\endcalendar}
3     {@{}}
4     {}
5     {}
6     {\LWR@patcherror{termcal}{endcalendar}}
```

Remove the hbox:

```
7 \xpatchcmd{\ca@doaday}
8     {\hbox to \hsize{\calprintdate\hfill\ifclassday\calprintclass\fi}}
9     {%
10         \calprintdate\hfill\ifclassday\calprintclass\fi%
11     }
12     {}
13     {\LWR@patcherror{termcal}{ca@doaday}}
```

Change each of two ampersands to call the lwarp tabular version:

```
14 \xpatchcmd{\calday}
15     {&}
16     {\LWR@tabularampersand}
17     {}
18     {\LWR@patcherror{termcal}{calday}}
19
```

```
20 \xpatchcmd{\calday}
21    {&}
22    {\LWR@tabularampersand}
23    {}
24    {\LWR@patcherror{termcal}{calday B}}
```

File 500   **`lwarp-textarea.sty`**

§ 612   Package   **textarea**

(*Emulates or patches code by* ALEXANDER I. ROZHENKO.)

textarea (*Pkg*)   textarea is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{textarea}[2005/12/26]`

```
2 \newcommand\StartFromTextArea{}
3 \newcommand\StartFromHeaderArea{}
4 \newcommand*\RestoreTextArea{}
5 \newcommand*\ExpandTextArea[1][*]{}
6 \let\NCC@restoretextarea\@empty
```

File 501   **`lwarp-textcomp.sty`**

§ 613   Package   **textcomp**

(*Emulates or patches code by* FRANK MITTELBACH, ROBIN FAIRBAIRNS, WERNER LEMBERG.)

textcomp (*Pkg*)   textcomp is patched for use by lwarp.

For MATHJAX, the MATHJAX packge is used.

§ 613.1   Limitations

Some textcomp symbols do not have Unicode equivalents, and thus are not supported.

⚠ missing symbols   Many textcomp symbols are not supported by many system / browser fonts. In the css try referencing fonts which are more complete, but expect to see gaps in coverage.

§ 613.2   Package loading

for HTML output: `1 \LWR@ProvidesPackagePass{textcomp}[2017/04/05]`

§ 613.3 HTML symbols

For HTML, use HTML entities or direct Unicode, depending on the engine.

\AtBeginDocument improves support for LuaLATEX and XƎLATEX.

§ 613.3.1 pdfLATEX symbols

```
2 \AtBeginDocument{
3 \ifPDFTeX% pdflatex or dvi latex
4 \newcommand*{\LWR@HTML@textdegree}{\HTMLentity{deg}}
5 \newcommand*{\LWR@HTML@textcelsius}{\HTMLunicode{2103}}
6 \newcommand*{\LWR@HTML@textohm}{\HTMLunicode{2126}}
7 \newcommand*{\LWR@HTML@textmu}{\HTMLunicode{00B5}}
8 \newcommand*{\LWR@HTML@textlquill}{\HTMLunicode{2045}}
9 \newcommand*{\LWR@HTML@textrquill}{\HTMLunicode{2046}}
10 \newcommand*{\LWR@HTML@textcircledP}{\HTMLunicode{2117}}
11 \newcommand*{\LWR@HTML@texttwelveudash}{\HTMLunicode{2014}}% emdash
12 \newcommand*{\LWR@HTML@textthreequartersemdash}{\HTMLunicode{2014}}% emdash
13 \newcommand*{\LWR@HTML@textmho}{\HTMLunicode{2127}}
14 \newcommand*{\LWR@HTML@textnaira}{\HTMLunicode{20A6}}
15 \newcommand*{\LWR@HTML@textpeso}{\HTMLunicode{20B1}}
16 \newcommand*{\LWR@HTML@textrecipe}{\HTMLunicode{211E}}
17 \newcommand*{\LWR@HTML@textinterrobang}{\HTMLunicode{203D}}
18 \newcommand*{\LWR@HTML@textinterrobangdown}{\HTMLunicode{2E18}}
19 \newcommand*{\LWR@HTML@textperthousand}{\HTMLunicode{2030}}
20 \newcommand*{\LWR@HTML@textpertenthousand}{\HTMLunicode{2031}}
21 \newcommand*{\LWR@HTML@textbaht}{\HTMLunicode{0E3F}}
22 \newcommand*{\LWR@HTML@textdiscount}{\%}
23 \newcommand*{\LWR@HTML@textservicemark}{\HTMLunicode{2120}}
24 \else
```

§ 613.3.2 XƎLATEX and LuaLATEX symbols

NOTE: Some of the following do not print well in the listing. Consult the .dtx or .sty file for the actual characters.

```
25 \newcommand*{\LWR@HTML@textdegree}{°}
26 \newcommand*{\LWR@HTML@textcelsius}{℃}
27 \newcommand*{\LWR@HTML@textohm}{Ω}
28 \newcommand*{\LWR@HTML@textmu}{µ}
29 \newcommand*{\LWR@HTML@textlquill}{⁅}
30 \newcommand*{\LWR@HTML@textrquill}{⁆}
31 \newcommand*{\LWR@HTML@textcircledP}{℗}
32 \newcommand*{\LWR@HTML@texttwelveudash}{—}% emdash
33 \newcommand*{\LWR@HTML@textthreequartersemdash}{—}% emdash
34 \newcommand*{\LWR@HTML@textmho}{℧}
35 \newcommand*{\LWR@HTML@textnaira}{₦}
36 \newcommand*{\LWR@HTML@textpeso}{₱}
37 \newcommand*{\LWR@HTML@textrecipe}{℞}
38 \newcommand*{\LWR@HTML@textinterrobang}{‽}
39 \newcommand*{\LWR@HTML@textinterrobangdown}{⸘}
40 \newcommand*{\LWR@HTML@textperthousand}{‰}
41 \newcommand*{\LWR@HTML@textpertenthousand}{‱}
42 \newcommand*{\LWR@HTML@textbaht}{฿}
43 \newcommand*{\LWR@HTML@textdiscount}{\%}
44 \newcommand*{\LWR@HTML@textservicemark}{℠}
45 \fi
```

```
46
47 \LWR@formatted{textdegree}
48 \LWR@formatted{textcelsius}
49 \LWR@formatted{textohm}
50 \LWR@formatted{textmu}
51 \LWR@formatted{textlquill}
52 \LWR@formatted{textrquill}
53 \LWR@formatted{textcircledP}
54 \LWR@formatted{texttwelveudash}
55 \LWR@formatted{textthreequartersemdash}
56 \LWR@formatted{textmho}
57 \LWR@formatted{textnaira}
58 \LWR@formatted{textpeso}
59 \LWR@formatted{textrecipe}
60 \LWR@formatted{textinterrobang}
61 \LWR@formatted{textinterrobangdown}
62 \LWR@formatted{textperthousand}
63 \LWR@formatted{textpertenthousand}
64 \LWR@formatted{textbaht}
65 \LWR@formatted{textdiscount}
66 \LWR@formatted{textservicemark}
```

§ 613.4   HTML diacritics

For HTML, Unicode diacritical marks are used:

```
67 \newcommand*{\LWR@HTML@capitalcedilla}[1]{#1\HTMLunicode{0327}}
68 \newcommand*{\LWR@HTML@capitalogonek}[1]{#1\HTMLunicode{0328}}
69 \newcommand*{\LWR@HTML@capitalgrave}[1]{#1\HTMLunicode{0300}}
70 \newcommand*{\LWR@HTML@capitalacute}[1]{#1\HTMLunicode{0301}}
71 \newcommand*{\LWR@HTML@capitalcircumflex}[1]{#1\HTMLunicode{0302}}
72 \newcommand*{\LWR@HTML@capitaltilde}[1]{#1\HTMLunicode{0303}}
73 \newcommand*{\LWR@HTML@capitaldieresis}[1]{#1\HTMLunicode{0308}}
74 \newcommand*{\LWR@HTML@capitalhungarumlaut}[1]{#1\HTMLunicode{30B}}
75 \newcommand*{\LWR@HTML@capitalring}[1]{#1\HTMLunicode{30A}}
76 \newcommand*{\LWR@HTML@capitalcaron}[1]{#1\HTMLunicode{30C}}
77 \newcommand*{\LWR@HTML@capitalbreve}[1]{#1\HTMLunicode{306}}
78 \newcommand*{\LWR@HTML@capitalmacron}[1]{#1\HTMLunicode{304}}
79 \newcommand*{\LWR@HTML@capitaldotaccent}[1]{#1\HTMLunicode{307}}
```

\textcircled becomes a span with a rounded border. \providecommand is used to avoid conflict with xunicode.

```
80 \providecommand*{\LWR@HTML@textcircled}[1]{%
81     \InlineClass[border: 1px solid \LWR@currenttextcolor]{textcircled}{#1}%
82 }
83
84 \LWR@formatted{capitalcedilla}
85 \LWR@formatted{capitalogonek}
86 \LWR@formatted{capitalgrave}
87 \LWR@formatted{capitalacute}
88 \LWR@formatted{capitalcircumflex}
89 \LWR@formatted{capitaltilde}
90 \LWR@formatted{capitaldieresis}
91 \LWR@formatted{capitalhungarumlaut}
92 \LWR@formatted{capitalring}
93 \LWR@formatted{capitalcaron}
94 \LWR@formatted{capitalbreve}
95 \LWR@formatted{capitalmacron}
96 \LWR@formatted{capitaldotaccent}
```

```
97
98 \LWR@formatted{textcircled}
```

Nullify **textcomp** macros when generating filenames:

```
 99 \FilenameNullify{%
100     \renewcommand*{\textdegree}{}%
101     \renewcommand*{\textcelsius}{}%
102     \renewcommand*{\textohm}{}%
103     \renewcommand*{\textmu}{}%
104     \renewcommand*{\textlquill}{}%
105     \renewcommand*{\textrquill}{}%
106     \renewcommand*{\textcircledP}{}%
107     \renewcommand*{\texttwelveudash}{}%
108     \renewcommand*{\textthreequartersemdash}{}%
109     \renewcommand*{\textmho}{}%
110     \renewcommand*{\textnaira}{}%
111     \renewcommand*{\textpeso}{}%
112     \renewcommand*{\textrecipe}{}%
113     \renewcommand*{\textinterrobang}{}%
114     \renewcommand*{\textinterrobangdown}{}%
115     \renewcommand*{\textperthousand}{}%
116     \renewcommand*{\textpertenthousand}{}%
117     \renewcommand*{\textbaht}{}%
118     \renewcommand*{\textdiscount}{}%
119     \renewcommand*{\textservicemark}{}%
120     \renewcommand*{\textcircled}[1]{#1}%
121     \renewcommand*{\capitalcedilla}[1]{#1}%
122     \renewcommand*{\capitalogonek}[1]{#1}%
123     \renewcommand*{\capitalgrave}[1]{#1}%
124     \renewcommand*{\capitalacute}[1]{#1}%
125     \renewcommand*{\capitalcircumflex}[1]{#1}%
126     \renewcommand*{\capitaltilde}[1]{#1}%
127     \renewcommand*{\capitaldieresis}[1]{#1}%
128     \renewcommand*{\capitalhungarumlaut}[1]{#1}%
129     \renewcommand*{\capitalring}[1]{#1}%
130     \renewcommand*{\capitalcaron}[1]{#1}%
131     \renewcommand*{\capitalbreve}[1]{#1}%
132     \renewcommand*{\capitalmacron}[1]{#1}%
133     \renewcommand*{\capitaldotaccent}[1]{#1}%
134 }% FilenameNullify
135
136 }% AtBeginDocument
```

For MATHJAX:

```
137 \CustomizeMathJax{\require{textcomp}}
```

File 502   **`lwarp-textfit.sty`**

§ 614   Package   **textfit**

textfit (*Pkg*)   **textfit** is emulated.

Text is placed into a <span> of class textfit. Sizes are approximated, and also limited by browser min/max font-size settings.

for HTML output: 1 \LWR@ProvidesPackageDrop{textfit}[1994/04/15]

2 \newsavebox{\LWR@textfitbox}
3
4 \newcommand*{\LWR@textfitscale}[2]{%
5 \setlength{\LWR@templengthone}{#1}%
6 \setlength{\LWR@templengthone}{%
7   1em*\ratio{\LWR@templengthone}{\LWR@templengthtwo}%
8 }%
9 \InlineClass[font-size:\LWR@printlength{\LWR@templengthone}]{textfit}{#2}%
10 }
11
12 \newcommand*{\scaletowidth}[2]{%
13 \sbox{\LWR@textfitbox}{#2}%
14 \settowidth{\LWR@templengthtwo}{\usebox{\LWR@textfitbox}}%
15 \LWR@textfitscale{#1}{#2}%
16 }
17
18 \newcommand*{\scaletoheight}[2]{%
19 \sbox{\LWR@textfitbox}{#2}%
20 \settoheight{\LWR@templengthtwo}{\usebox{\LWR@textfitbox}}%
21 \LWR@textfitscale{#1}{#2}%
22 }

File 503   **`lwarp-textpos.sty`**

§ 615   Package   # textpos

(*Emulates or patches code by* NORMAN GRAY.)

textpos (*Pkg*)   textpos is emulated.

for HTML output: 1 \LWR@ProvidesPackageDrop{textpos}[2020/09/26]

2 \NewDocumentEnvironment{textblock}{m o r()}{}{}
3 \NewDocumentEnvironment{textblock*}{m o r()}{}{}
4 \newcommand*{\TPGrid}[3][]{}
5 \def\TPShowGrid{\@ifstar{\@TPShowGrid}{\@TPShowGrid}}
6 \def\@TPShowGrid#1#2{}
7 \NewDocumentCommand{\TPMargin}{s o}{}
8 \newcommand*{\textblockcolour}[1]{}
9 \newcommand*{\textblockrulecolour}[1]{}
10 \newcommand*{\textblockcolor}[1]{}
11 \newcommand*{\textblockrulecolor}[1]{}
12 \newcommand*{\tekstblokkulur}[1]{}
13 \newcommand*{\tekstblokrulekulur}[1]{}
14 \newlength{\TPHorizModule}
15 \newlength{\TPVertModule}
16 \newlength{\TPboxrulesize}
17 \newcommand{\textblocklabel}[1]{}
18 \newcommand*{\showtextsize}{}
19 \newcommand{\textblockorigin}[2]{}
20 \newcommand*{\TPoptions}[1]{}
21 \newcommand*{\TPReferencePosition}[1]{}

File 504   `lwarp-theorem.sty`

§ 616   Package   **theorem**

(*Emulates or patches code by* FRANK MITTELBACH.)

theorem (*Pkg*)   theorem is patched for use by lwarp.

Table 21: Theorem package — CSS styling of theorems and proofs

Theorem: `<div>` of class `theorembody<theoremstyle>`

Theorem Header: `<span>` of class `theoremheader`

where `<theoremstyle>` is plain, break, etc.

for HTML output: 1 `\LWR@ProvidesPackagePass{theorem}[2023/07/05]`

§ 616.1   Remembering the theorem style

Storage for the style being used for new theorems:

2 `\newcommand{\LWR@newtheoremstyle}{plain}`

Patched to remember the style being used for new theorems:

```
3 \VerifyCommand[lwarp][theorem]{\theoremstyle}{B805673118A2EA934449A9B7D25A5D33}
4
5 \gdef\theoremstyle#1{%
6    \@ifundefined{th@#1}{\@warning
7        {Unknown theoremstyle `#1'. Using `plain'}%
8        \theorem@style{plain}%
9         \renewcommand{\LWR@newtheoremstyle}{plain}% lwarp
10       }%
11    {%
12        \theorem@style{#1}%
13        \renewcommand{\LWR@newtheoremstyle}{#1}% lwarp
14    }%
15    \begingroup
16      \csname th@\the\theorem@style \endcsname
17    \endgroup}
```

Patched to remember the style for this theorem type, and set it later when the environment is started.

```
18 \VerifyCommand[lwarp][theorem]{\@xnthm}{D6164703589C684059381DB798F89158}
19
20 \gdef\@xnthm#1#2[#3]{%
21    \expandafter\@ifdefinable\csname #1\endcsname
22    {%
23    \csedef{LWR@thmstyle#1}{\LWR@newtheoremstyle}% lwarp
24    \@definecounter{#1}\@newctr{#1}[#3]%
25    \expandafter\xdef\csname the#1\endcsname
```

```
26          {\expandafter \noexpand \csname the#3\endcsname
27           \@thmcountersep \@thmcounter{#1}}%
28        \def\@tempa{\global\@namedef{#1}}%
29        \expandafter \@tempa \expandafter{%
30          \csname th@\the \theorem@style
31               \expandafter \endcsname \the \theorem@bodyfont
32          \@thm{#1}{#2}}%
33        \global \expandafter \let \csname end#1\endcsname \@endtheorem
34      \AtBeginEnvironment{#1}{\edef\LWR@thisthmstyle{\@nameuse{LWR@thmstyle#1}}}% lwarp
35        }}
36
37  \VerifyCommand[lwarp][theorem]{\@ynthm}{C5A12EBEFDBCD5C5628C65B16A01DFB4}
38
39  \gdef\@ynthm#1#2{%
40       \expandafter\@ifdefinable\csname #1\endcsname
41       {
42       \csedef{LWR@thmstyle#1}{\LWR@newtheoremstyle}% lwarp
43       \@definecounter{#1}%
44       \expandafter\xdef\csname the#1\endcsname{\@thmcounter{#1}}%
45       \def\@tempa{\global\@namedef{#1}}\expandafter \@tempa
46        \expandafter{\csname th@\the \theorem@style \expandafter
47        \endcsname \the\theorem@bodyfont \@thm{#1}{#2}}%
48       \global \expandafter \let \csname end#1\endcsname \@endtheorem
49      \AtBeginEnvironment{#1}{\edef\LWR@thisthmstyle{\@nameuse{LWR@thmstyle#1}}}% lwarp
50        }}
51
52  \VerifyCommand[lwarp][theorem]{\@othm}{93B7CCDCEFDF36BEEF31477D6D390AC3}
53
54  \gdef\@othm#1[#2]#3{%
55    \expandafter\ifx\csname c@#2\endcsname\relax
56     \@nocounterr{#2}%
57    \else
58     \expandafter\@ifdefinable\csname #1\endcsname
59     {
60      \csedef{LWR@thmstyle#1}{\LWR@newtheoremstyle}% lwarp
61      \expandafter \xdef \csname the#1\endcsname
62        {\expandafter \noexpand \csname the#2\endcsname}%
63      \def\@tempa{\global\@namedef{#1}}\expandafter \@tempa
64        \expandafter{\csname th@\the \theorem@style \expandafter
65        \endcsname \the\theorem@bodyfont \@thm{#2}{#3}}%
66      \global \expandafter \let \csname end#1\endcsname \@endtheorem
67      \AtBeginEnvironment{#1}{\edef\LWR@thisthmstyle{\@nameuse{LWR@thmstyle#1}}}% lwarp
68       }%
69    \fi}
```

§ 616.2   css patches

The following are patched for css.

These were in individual files thp.sty for plain, thmb.sty for margin break, etc.
They are gathered together here.

Each theorem is encased in a BlockClass environment of class theorembody<style>.

Each header is encased in an \InlineClass of class theoremheader.

```
70  \gdef\th@plain{%
71    \def\@begintheorem##1##2{%
72          \item[
73              \InlineClass{theoremheader}{##1\ ##2}
```

```
 74         ]
 75     }%
 76 \def\@opargbegintheorem##1##2##3{%
 77    \item[
 78        \InlineClass{theoremheader}{##1\ ##2\ (##3)}
 79    ]
 80    }
 81 }
 82
 83 \gdef\th@break{%
 84   \def\@begintheorem##1##2{%
 85     \item[
 86        \InlineClass{theoremheader}{##1\ ##2}\newline%
 87    ]
 88    }%
 89 \def\@opargbegintheorem##1##2##3{%
 90   \item[
 91        \InlineClass{theoremheader}{##1\ ##2\ (##3)}\newline
 92   ]
 93   }
 94 }
 95
 96 \gdef\th@marginbreak{%
 97   \def\@begintheorem##1##2{
 98     \item[
 99        \InlineClass{theoremheader}{##2 \qquad ##1}\newline
100    ]
101    }%
102 \def\@opargbegintheorem##1##2##3{%
103    \item[
104        \InlineClass{theoremheader}{##2 \qquad ##1\ %
105        (##3)}\newline
106    ]
107    }
108 }
109
110 \gdef\th@changebreak{%
111   \def\@begintheorem##1##2{
112     \item[
113        \InlineClass{theoremheader}{##2\ ##1}\newline
114    ]
115    }%
116 \def\@opargbegintheorem##1##2##3{%
117    \item[
118        \InlineClass{theoremheader}{ ##2\ ##1\ %
119        (##3)}\newline
120    ]
121    }
122 }
123
124 \gdef\th@change{%
125   \def\@begintheorem##1##2{
126     \item[
127        \InlineClass{theoremheader}{##2\ ##1}
128    ]
129    }%
130 \def\@opargbegintheorem##1##2##3{%
131    \item[
132        \InlineClass{theoremheader}{##2\ ##1\ (##3)}
133    ]
```

```
134      }
135 }
136
137 \gdef\th@margin{%
138   \def\@begintheorem##1##2{
139     \item[
140         \InlineClass{theoremheader}{##2 \qquad ##1}
141     ]
142   }%
143 \def\@opargbegintheorem##1##2##3{%
144     \item[
145             \InlineClass{theoremheader}{##2 \qquad ##1\ (##3)}
146     ]
147   }
148 }
```

Patched for css:

```
149 \VerifyCommand[lwarp][theorem]{\@thm}{4632915C52ABB4DB5D462AA58A80BAF2}
150
151 \gdef\@thm#1#2{\refstepcounter{#1}%
152 \LWR@forcenewpage% lwarp


153     \LWR@printpendingfootnotes%                              lwarp


154     \BlockClass{theorembody\LWR@thisthmstyle}% lwarp
155   \trivlist
156   \@topsep \theorempreskipamount                   % used by first \item
157   \@topsepadd \theorempostskipamount               % used by \@endparenv
158   \@ifnextchar [%
159   {\@ythm{#1}{#2}}%
160   {\@begintheorem{#2}{\csname the#1\endcsname}\ignorespaces}}
161
162 \VerifyCommand[lwarp][theorem]{\@endtheorem}
163         {9798301819F2CA2E46673F8937BABE99}
164 \gdef\@endtheorem{%
165 \endtrivlist


166     \LWR@printpendingfootnotes%                        lwarp


167 \endBlockClass
168 }
```

File 505   **`lwarp-thinsp.sty`**

§ 617   Package   # thinsp

thinsp (*Pkg*)   thinsp is emulated.

for HTML output: `1 \LWR@ProvidesPackageDrop{thinsp}[2016/10/02]`

```
2 \AtBeginDocument{
3 \let\thinthinspace\relax% defined by some packages
4 \newcommand*{\thinthinspace}{\thinspace}
5 }
```

```
6
7 \newcommand*{\stretchthinspace}{\thinspace}
8 \newcommand*{\stretchthinthinspace}{\thinthinspace}
9 \newcommand*{\stretchnegthinspace}{\negthinspace}
```

File 506 **`lwarp-thm-listof.sty`**

§ 618 Package **thm-listof**

(*Emulates or patches code by* Ulrich M. Schwarz, Yukai Chou.)

thm-listof (*Pkg*) thm-listof is part of thmtools, and is patched for use by lwarp.

for HTML output: `1 \LWR@ProvidesPackagePass{thm-listof}[2019/12/22]`

For font control, see the generated HTML and use CSS per amsthm or ntheorem.

Other thm-* package may be loaded by thm-listof.

```
2 \IfPackageAtLeastTF{thm-listof}{2020/08/01}{% v0.72
3   \VerifyCommand[lwarp][thm-listof]{\thmtlo@newentry}{AF7BF291DC508ED71058DAF745F9C018}
4   \def\thmtlo@newentry{%
5     \csdef{l@\thmt@envname}##1##2{\hypertocfloat{1}{figure}{lof}{##1}{##2}}%
6   }
7 }{% earlier than v0.72
8   \xpatchcmd{\listoftheorems}
9     {%
10        \@xa\protected@edef\csname l@\thmt@envname\endcsname{%
11            \@nx\@dottedtocline{1}{1.5em}{\@nx\thmt@listnumwidth}%
12        }%
13     }
14     {%
15     \csdef{l@\thmt@envname}##1##2{\hypertocfloat{1}{figure}{lof}{##1}{##2}}%
16     }
17     {}
18     {\LWR@patcherror{thm-listof}{listoftheorems}}
19
20   \xpatchcmd{\thmt@mklistcmd}
21     {%
22        \@xa\protected@edef\csname l@\thmt@envname\endcsname{%
23            \@nx\@dottedtocline{1}{1.5em}{\@nx\thmt@listnumwidth}%
24        }%
25     }
26     {%
27     \csdef{l@\thmt@envname}##1##2{\hypertocfloat{1}{figure}{lof}{##1}{##2}}%
28     }
29     {}
30     {\LWR@patcherror{thm-listof}{thmt@mklistcmd}}
31 }
```

File 507 **`lwarp-thm-restate.sty`**

§ 619 Package **thm-restate**

(*Emulates or patches code by* Ulrich M. Schwarz.)

thm-restate (*Pkg*)   **thm-restate** is part of **thmtools**, and is patched for use by **lwarp**.

for HTML output: 1 \LWR@ProvidesPackagePass{thm-restate}[2020/08/01]

2 \VerifyCommand[lwarp][thm-restate]{\thmt@restatable}{C912622BBA051C5F22994335F66976AB}
3
4 \xpatchcmd{\thmt@restatable}
5   {\@ifstar}
6   {\edef\LWR@thisthmstyle{#2}\@ifstar}
7   {}
8   {\LWR@patcherror{thm-restate}{thmt@restatable}}

File 508   `lwarp-thmbox.sty`

§ 620   Package   # thmbox

(*Emulates or patches code by* EMMANUEL BEFFARA.)

thmbox (*Pkg*)   **thmbox** is emulated for use by **lwarp**.

for HTML output: 1 \LWR@ProvidesPackagePass{thmbox}[2005/04/24]

2 \renewenvironment{thmbox}[2][]%
3   {%
4   \begin{BlockClass}{thmbox}
5   \begin{BlockClass}{thmboxtitle}
6   #2
7   \end{BlockClass}
8   }
9   {\end{BlockClass}}
10
11 \renewenvironment{proof}[1][]
12   {%
13   \begin{BlockClass}{thmboxproof}%
14   \InlineClass{thmboxproofname}{\proofname\ #1\unskip\,:}
15   }
16   {%
17   \qquad\HTMLunicode{220E}
18   \end{BlockClass}
19   }
20
21 \renewenvironment{example}[1][\examplename]%
22   {%
23   \begin{BlockClass}{thmboxexample}%
24   \InlineClass{thmboxexamplename}{#1\,:}
25   }
26   {\end{BlockClass}}
27
28 \renewenvironment{leftbar}[1][]%
29   {\begin{BlockClass}{thmboxleftbar}}
30   {\end{BlockClass}}

File 509 **`lwarp-thmtools.sty`**

§ 621   Package   # thmtools

(*Emulates or patches code by* ULRICH M. SCHWARZ.)

thmtools (*Pkg*)   thmtools is patched for use by lwarp.

Also see thm-listof and thm-restate.

for HTML output: 1 \LWR@ProvidesPackagePass{thmtools}[2020/08/01]

The following patches either thm-amsthm or thm-ntheorem.

```
2 \def\thmt@headstyle@margin{%
3     \InlineClass{amsthmnnumbertheorem}{\NUMBER}
4     \
5     \InlineClass{amsthmnametheorem}{\NAME}
6     \InlineClass{amsthmnotetheorem}{\NOTE}
7 }
8
9 \let\thmt@headstyle@swapnumber\thmt@headstyle@margin
```

File 510   **`lwarp-threadcol.sty`**

§ 622   Package   # threadcol

threadcol (*Pkg*)   threadcol is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{threadcol}[2013/01/06]

2 \newcommand{\setthreadname}[1]{}

File 511   **`lwarp-threeparttable.sty`**

§ 623   Package   # threeparttable

(*Emulates or patches code by* DONALD ARSENEAU.)

threeparttable (*Pkg*)   threeparttable is emulated.

Table note are contained inside a CSS <div> of class tnotes. If enumitem is used, the note item labels are also individually highlighted with an additional CSS <span> of class tnoteitemheader, otherwise they are plain text.

for HTML output: 1 \LWR@ProvidesPackageDrop{threeparttable}[2003/06/13]

Env   threeparttable   [⟨*alignment*⟩]

```
2 \newenvironment*{threeparttable}[1][b]
3    {\def\@captype{table}}
4    {}
```

Env   tablenotes   [⟨*options*⟩]

```
5 \newenvironment*{tablenotes}[1][]
6 {%
7 \LWR@forcenewpage
8 \BlockClass{tnotes}%

9 \description%
10 }
11 {%
12 \enddescription%
13 \endBlockClass%
14 }
```

\tnote   {⟨*text*⟩}

```
15 \newcommand{\tnote}[1]{\LWR@htmlspan{sup}{#1}}
```

Env   measuredfigure   [⟨*alignment*⟩]

```
16 \newenvironment*{measuredfigure}[1][t]
17    {\def\@captype{figure}}
18    {}
```

File 512   **lwarp-threeparttablex.sty**

§ 624   Package   # threeparttablex

threeparttablex (*Pkg*)   threeparttablex is patched for use by lwarp.

threeparttablex is used with longtable and booktabs as follows:

```
\begin{longtable}{ [column specifiers] }
[ . . . ] \endfirsthead   % or \endhead, for print and HTML
\warpprintonly{          % not used in HTML
  [ . . . ] \endhead      % or \endfirsthead
  [ . . . ] \endfoot
  \bottomrule \insertTableNotes \endlastfoot
}
. . . table contents . . .
\warpHTMLonly{   % HTML last footer
  \bottomrule
  \UseMinipageWidths      % optional
  \insertTableNotes
  \endlastfoot
}
\end{longtable}
```

table width   The table notes are created using a \multicolumn. By default the width is not specified to the browser, so long table notes can cause the table to be spread out horizontally. For HTML output, lwarp guesses the width of the table depending on the number of columns, then restricts its guess to a min/max range. To

use this guess for the width of the table notes, use \UseMinipageWidths before \insertTableNotes. The width is then specified, and in many cases the result is an improvement in overall table layout.

1 \LWR@ProvidesPackagePass{threeparttablex}[2013/07/23]

The width is guessed depending on the number of columns, then limited to a min/max.

```
2 \renewcommand\insertTableNotes{%
3     \setlength{\LWR@templengthone}{.375in*\value{LWR@tabletotalLaTeXcols}}%
4     \setlength{\LWR@templengthone}{\minof{\textwidth}{\LWR@templengthone}}%
5     \setlength{\LWR@templengthone}{\maxof{2.5in}{\LWR@templengthone}}%
6  \multicolumn{\value{LWR@tabletotalLaTeXcols}}{c}{%
7     \parbox{\LWR@templengthone}{%
8        \begin{tablenotes}[\TPTL@optarg]%
9           \TPTL@font%
10          \TPTL@body%
11       \end{tablenotes}%
12     }%
13  }%
14 }
```

```
15 \providecommand{\TPTL@tnotex}{}
16 \renewcommand{\TPTL@tnotex}[2]{\tnote{\nameref{#2}}}
```

File 513   `lwarp-thumb.sty`

§ 625   Package **thumb**

thumb (*Pkg*)   thumb is ignored.

1 \LWR@ProvidesPackageDrop{thumb}[1997/12/24]

```
2 \newcommand*{\Overviewpage}{}
3 \newlength{\thumbheight}
4 \newlength{\thumbwidth}
```

File 514   `lwarp-thumbs.sty`

§ 626   Package **thumbs**

thumbs (*Pkg*)   thumbs is ignored.

1 \LWR@ProvidesPackageDrop{thumbs}[2014/03/09]

```
2 \newcommand{\addthumb}[4]{}
3 \newcommand{\addtitlethumb}[5]{}
4 \newcommand{\stopthumb}{}
5 \newcommand{\continuethumb}{}
6 \newcommand{\thumbsoverview}[1]{}
7 \newcommand{\thumbsoverviewback}[1]{}
```

```
8 \newcommand{\thumbsoverviewverso}[1]{}
9 \newcommand{\thumbsoverviewdouble}[1]{}
10 \newcommand{\thumbnewcolumn}{}
11 \newcommand{\addthumbsoverviewtocontents}[2]{}
12 \newcommand{\thumbsnophantom}{}
```

File 515   **`lwarp-tikz.sty`**

§ 627   Package   # tikz

(*Emulates or patches code by* TILL TANTAU.)

tikz (*Pkg*)   tikz is supported.

⚠ displaymath and   If using display math with tikzpicture or \tikz, along with matrices with the &
matrices   character, the document must be modified as follows:

```
\usepackage{tikz}
\tikzset{every picture/.style={ampersand replacement=\&}}
```

and each instance of & in the tikz expression must be replaced with \&.

Accept all options for lwarp-tikz:

```
1 \LWR@ProvidesPackagePass{tikz}[2015/08/07]
```

catcodes   lwarp changes the catcode of $ for its own use. The Ti*k*Z babel library temporarily
changes catcodes back to normal for Ti*k*Z's use. tikz v3.0.0 introduced the babel
library which handles catcode changes. For older versions, lwarp must change $'s
catcode itself.

Also see:

https://tex.stackexchange.com/questions/16199/
    test-if-a-package-or-package-option-is-loaded

```
2 \newbool{LWR@tikzbabel}
3
4 \IfPackageAtLeastTF{tikz}{2013/12/20}% Test for Tikz version v3.0.0
5 {\usetikzlibrary{babel}\booltrue{LWR@tikzbabel}}
6 {\boolfalse{LWR@tikzbabel}}
```

Env pgfpicture   The \pgfpicture environment is enclosed inside a \lateximage. Enclose the
low-level \pgfpicture in a lateximage. This is also used by the higher-level \tikz
and tikzpicture.

```
7 \preto\pgfpicture{%
8     \begin{lateximage}[-tikz-~\PackageDiagramAltText]?%
9     \ifbool{LWR@tikzbabel}% Test for Tikz version v3.0.0
10    {}%
11    {\catcode`\$=3}% dollar sign is math shift
12 }
13
14 \appto\endpgfpicture{\end{lateximage}}
```

Ti*k*Z is placed inside an svg image, so use the original meanings of the following:

```
15 \LetLtxMacro\pgfutil@minipage\LWR@print@minipage
16 \let\pgfutil@endminipage\endLWR@print@minipage
17
18 \let\pgfutil@raggedleft\LWR@print@raggedleft
19 \let\pgfutil@raggedright\LWR@print@raggedright

20 \def\pgfutil@font@tiny{\LWR@print@tiny}
21 \def\pgfutil@font@scriptsize{\LWR@print@scriptsize}
22 \def\pgfutil@font@footnotesize{\LWR@print@footnotesize}
23 \def\pgfutil@font@small{\LWR@print@small}
24 \def\pgfutil@font@normalsize{\LWR@print@normalsize}
25 \def\pgfutil@font@large{\LWR@print@large}
26 \def\pgfutil@font@Large{\LWR@print@Large}
27 \def\pgfutil@font@huge{\LWR@print@huge}
28 \def\pgfutil@font@Huge{\LWR@print@Huge}
29
30 \def\pgfutil@font@itshape{\LWR@print@itshape}
31 \def\pgfutil@font@bfseries{\LWR@print@bfseries}
32
33 \def\pgfutil@font@normalfont{\LWR@print@normalfont}
```

File 516   **`lwarp-tikz-imagelabels.sty`**

§ 628   Package   # tikz-imagelabels

(*Emulates or patches code by* Tobias Plüss.)

tikz-imagelabels (*Pkg*)   tikz-imagelabels is patched for use by lwarp.

for HTML output: `1 \LWR@ProvidesPackagePass{tikz-imagelabels}[2019/06/27]`

```
2 \BeforeBeginEnvironment{annotationimage}{%
3     \begin{lateximage}[-tikz-imagelabels-~\PackageDiagramAltText]?%
4 }
5
6 \AfterEndEnvironment{annotationimage}{\end{lateximage}}
```

File 517   **`lwarp-titleps.sty`**

§ 629   Package   # titleps

(*Emulates or patches code by* Javier Bezos.)

titleps (*Pkg*)   titleps is loaded and used by lwarp during HTML output. All user options and macros are ignored and disabled.

Discard all options for lwarp-titleps:

for HTML output: `1 \LWR@ProvidesPackageDrop{titleps}[2016/03/15]`

\pagestyle and \thispagestyle are already disabled in the lwarp code.

\newpagestyle   {⟨*name*⟩} [⟨*style*⟩] {⟨*commands*⟩}

`2 \NewDocumentCommand{\newpagestyle}{m o m}{}`

\renewpagestyle   {⟨*name*⟩} [⟨*style*⟩] {⟨*commands*⟩}

3 \NewDocumentCommand{\renewpagestyle}{m o m}{}

\sethead   [⟨*el*⟩] [⟨*ec*⟩] [⟨*er*⟩] {⟨*ol*⟩} {⟨*oc*⟩} {⟨*or*⟩}

4 \NewDocumentCommand{\sethead}{o o o m m m}{}

\setfoot   [⟨*el*⟩] [⟨*ec*⟩] [⟨*er*⟩] {⟨*ol*⟩} {⟨*oc*⟩} {⟨*or*⟩}

5 \NewDocumentCommand{\setfoot}{o o o m m m}{}

\settitlemarks   * {⟨*names*⟩}

6 \NewDocumentCommand{\settitlemarks}{s m}{}

\headrule

7 \newcommand*{\headrule}{}

\footrule

8 \newcommand*{\footrule}{}

\setheadrule   {⟨*length*⟩}

9 \newcommand*{\setheadrule}[1]{}

\setfootrule   {⟨*length*⟩}

10 \newcommand*{\setfootrule}[1]{}

\makeheadrule

11 \newcommand*{\makeheadrule}{}

\makefootrule

12 \newcommand*{\makefootrule}{}

\setmarkboth   {⟨*code*⟩}

13 \newcommand{\setmarkboth}[1]{}

\widenhead

14 \NewDocumentCommand{\widenhead}{s o o m m}{}

\bottitlemarks

15 \newcommand*{\bottitlemarks}{}

\toptitlemarks

16 \newcommand*{\toptitlemarks}{}

\firsttitlemarks

> 17 \newcommand*{\firsttitlemarks}{}

\nexttitlemarks

> 18 \newcommand*{\nexttoptitlemarks}{}

\outertitlemarks

> 19 \newcommand*{\outertitlemarks}{}

\innertitlemarks

> 20 \newcommand*{\innertitlemarks}{}

\newtitlemark   * {⟨*name*⟩}

> 21 \NewDocumentCommand{\newtitlemark}{s m}{}

\pretitlemark   * {⟨*section*⟩} {⟨*text*⟩}

> 22 \NewDocumentCommand{\pretitlemark}{s m m}{}

\ifsamemark   {⟨*group*⟩} {⟨*command*⟩} {⟨*true*⟩} {⟨*false*⟩}

> 23 \newcommand{\ifsamemark}[4]{}

\setfloathead   * [⟨.⟩] [⟨.⟩] [⟨.⟩] {⟨.⟩} {⟨.⟩} {⟨.⟩} {⟨*extra*⟩} [⟨*which*⟩]

> 24 \NewDocumentCommand{\setfloathead}{s o o o m m m m}{}

\setfloatfoot   * [⟨.⟩] [⟨.⟩] [⟨.⟩] {⟨.⟩} {⟨.⟩} {⟨.⟩} {⟨*extra*⟩} [⟨*which*⟩]

> 25 \NewDocumentCommand{\setfloatfoot}{s o o o m m m m}{}

\nextfloathead   * [⟨.⟩] [⟨.⟩] [⟨.⟩] {⟨.⟩} {⟨.⟩} {⟨.⟩} {⟨*extra*⟩} [⟨*which*⟩]

> 26 \NewDocumentCommand{\nextfloathead}{s o o o m m m m}{}

\nextfloatfoot   * [⟨.⟩] [⟨.⟩] [⟨.⟩] {⟨.⟩} {⟨.⟩} {⟨.⟩} {⟨*extra*⟩} [⟨*which*⟩]

> 27 \NewDocumentCommand{\nextfloatfoot}{s o o o m m m m}{}

\newmarkset   {⟨*markset*⟩}

> 28 \newcommand{\newmarkset}[1]{}

\newextramark   * {⟨*markset*⟩} {⟨*macro-name*⟩}

> 29 \NewDocumentCommand{\newextramarkset}{s m m}{}

\botextramarks   {⟨*markset*⟩}

> 30 \newcommand{\botextramarks}[1]{}

\topextramarks   {⟨*markset*⟩}

31 \newcommand{\topextramarks}[1]{}

\firstextramarks   {⟨*markset*⟩}

32 \newcommand{\firstextramarks}[1]{}

\nextextramarks   {⟨*markset*⟩}

33 \newcommand{\nexttopextramarks}[1]{}

\outerextramarks   {⟨*markset*⟩}

34 \newcommand{\outerextramarks}[1]{}

\innerextramarks   {⟨*markset*⟩}

35 \newcommand{\innerextramarks}[1]{}

File 518   `lwarp-titleref.sty`

§ 630   Package   **titleref**

titleref (*Pkg*)   titleref is emulated.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{titleref}[2001/04/05]
2
3 \LetLtxMacro\titleref\nameref
4
5 \providecounter{LWR@currenttitle}
6
7 \newcommand*{\currenttitle}{%
8     \addtocounter{LWR@currenttitle}{1}%
9     \label{currenttitle\arabic{LWR@currenttitle}}%
10    \nameref{currenttitle\arabic{LWR@currenttitle}}%
11 }
12
13 \newcommand*{\theTitleReference}[2]{}
```

File 519   `lwarp-titlesec.sty`

§ 631   Package   **titlesec**

(*Emulates or patches code by* JAVIER BEZOS.)

titlesec (*Pkg*)   titlesec is emulated. All user options and macros are ignored and disabled.

Discard all options for lwarp-titlesec:

for HTML output:
```
1 \LWR@ProvidesPackageDrop{titlesec}[2016/03/21]
2
3 \newbool{LWR@loadtitleps}
```

```
4 \boolfalse{LWR@loadtitleps}
5
6 \DeclareOption{pagestyles}{
7     \booltrue{LWR@loadtitleps}
8 }
9
10 \DeclareOption*{}
11
12 \ProcessOptions\relax
13
14 \ifbool{LWR@loadtitleps}{
15     \RequirePackage{lwarp-titleps}
16 }{}
```

\titlelabel   {⟨*label-format*⟩}

```
17 \newcommand*{\titlelabel}[1]{}
```

\titleformat*   {⟨*command*⟩} {⟨*format*⟩}

\titleformat   {⟨*command*⟩} [⟨*shape*⟩] {⟨*format*⟩} {⟨*label*⟩} {⟨*sep*⟩} {⟨*begfore*⟩} [⟨*after*⟩]

```
18 \newcommand\titleformat{%
19   \@ifstar{\ttl@format@s}%
20         {\ttl@format@i}}
21 \newcommand{\ttl@format@s}[1]{}
22 \NewDocumentCommand{\ttl@format@i}{m o m m m m o}{}
```

\chaptertitlename

```
23 \@ifundefined{@chapapp}{\let\@chapapp\chaptername}{}
24 \newcommand\chaptertitlename{\@chapapp}
```

\titlespacing   * {⟨*command*⟩} {⟨*left*⟩} {⟨*before*⟩} {⟨*after*⟩} [⟨*right*⟩]

```
25 \NewDocumentCommand{\titlespacing}{s m m m m o}{}
```

\filright

```
26 \newcommand*{\filright}{}
```

\filcenter

```
27 \newcommand*{\filcenter}{}
```

\filleft

```
28 \newcommand*{\filleft}{}
```

\fillast

```
29 \newcommand*{\fillast}{}
```

\filinner

```
30 \newcommand*{\filinner}{}
```

\filouter

31 \newcommand*{\filouter}{}

\wordsep

32 \newcommand\wordsep{\fontdimen\tw@\font \@plus
33   \fontdimen\thr@@\font \@minus \fontdimen4\font}

\titleline   * [⟨*align*⟩] {⟨*material*⟩}

34 \NewDocumentCommand{\titleline}{s o m}{}

\titlerule   [⟨*height*⟩]

35 \providecommand*\titlerule{\@ifstar{\ttl@row}{\ttl@rule}}
36 \newcommand*{\ttl@rule}[1][]{}
37 \newcommand*{\ttl@row}[2][]{}

\iftitlemeasuring   {⟨*true*⟩} {⟨*false*⟩}

38 \newcommand{\iftitlemeasuring}[2]{#2}

\assignpagestyle   {⟨*command*⟩} {⟨*pagestyle*⟩}

39 \newcommand{\assignpagestyle}[2]{#2}

\titleclass   {⟨*name*⟩} [⟨*startlevel*⟩] {⟨*class*⟩} [⟨*cmd*⟩]

40 \NewDocumentCommand{\titleclass}{m o m o}{}

File 520   **lwarp-titletoc.sty**

§ 632   Package   **titletoc**

(*Emulates or patches code by* Javier Bezos.)

titletoc (*Pkg*)   titletoc is emulated. All user options and macros are ignored and disabled.

Discard all options for lwarp-titletoc:

for HTML output: 1 \LWR@ProvidesPackageDrop{titletoc}[2011/12/15]

\dottedcontents   {⟨*section*⟩} [⟨*left*⟩] {⟨*above*⟩} {⟨*label*⟩} {⟨*leader*⟩}

2 \NewDocumentCommand{\dottedcontents}{m o m m m}{}

\titlecontents   * {⟨*section*⟩} [⟨*left*⟩] {⟨*above*⟩} {⟨*numbered*⟩} {⟨*numberless*⟩} {⟨*filler*⟩} [⟨*below
                          or begin*⟩] [⟨*separator*⟩] [⟨*end*⟩]

3 \newcommand{\titlecontents}{\@ifstar{\ttl@tcstar}{\ttl@tcnostar}}
4 \NewDocumentCommand{\ttl@tcstar}{m o m m m m o o o}{}
5 \NewDocumentCommand{\ttl@tcnostar}{m o m m m m o}{}

\contentsmargin   [⟨*correction*⟩] {⟨*right*⟩}

```
6 \newcommand{\contentsmargin}[2][]{}
```

\thecontentslabel

```
7 \newcommand*{\thecontentslabel}{thecontentslabel}
```

\thecontentspage

```
8 \newcommand*{\thecontentspage}{thecontentspage}
```

\contentslabel   [⟨*format*⟩] {⟨*space*⟩}

```
9 \newcommand{\contentslabel}[2][]{\thecontentslabel}
```

\contentspage   [⟨*format*⟩]

```
10 \newcommand{\contentspage}[1][]{\thecontentspage}
```

\contentspush   {⟨*text*⟩}

```
11 \newcommand{\contentspush}[1]{}
```

\contentsuse   {⟨*name*⟩} {⟨*text*⟩}

```
12 \newcommand{\contentsuse}[2]{}
```

\startcontents   [⟨*name*⟩]

```
13 \newcommand*{\startcontents}[1][]{}
```

\stopcontents   [⟨*name*⟩]

```
14 \newcommand*{\stopcontents}[1][]{}
```

\resumecontents   [⟨*name*⟩]

```
15 \newcommand*{\resumecontents}[1][]{}
```

\printcontents   [⟨*name*⟩] {⟨*prefix*⟩} {⟨*start*⟩} {⟨*code*⟩}

```
16 \newcommand{\printcontents}[4][]{}
```

\startlist   [⟨*name*⟩] {⟨*list*⟩}

```
17 \newcommand{\startlist}[2][]{}
```

\stoplist   [⟨*name*⟩] {⟨*list*⟩}

```
18 \newcommand{\stoplist}[2][]{}
```

\resumelist   [⟨*name*⟩] {⟨*list*⟩}

```
19 \newcommand{\resumelist}[2][]{}
```

\printlist   [⟨*name*⟩] {⟨*list*⟩} {⟨*prefix*⟩} {⟨*code*⟩}

                                    20 \newcommand{\printlist}[4][]{}

              File 521   **lwarp-titling.sty**

§ 633   Package   **titling**

                        (*Emulates or patches code by* PETER WILSON.)

titling (*Pkg*)

package support   lwarp supports the native LATEX titling commands, and also supports the packages
⚠   load order   authblk and titling. If both are used, authblk should be loaded before titling.

\published and \subtitle If using the titling package, additional titlepage fields for \published and \subtitle
                        may be added by using \AddSubtitlePublished in the preamble. See section 71.8.

                        The various titling footnote restyling commands have no effect.

                        Pass all options to lwarp-titling:

for HTML output: 1 \LWR@ProvidesPackagePass{titling}[2009/09/04]

\@bsmtitlempty   Patch \@bsmtitlempty:

                        2 \let\LWR@orig@bsmtitlempty\@bsmtitlempty
                        3 \renewcommand*{\@bsmtitlempty}{%
                        4 \LWR@orig@bsmtitlempty%
                        5 }

\keepthetitle   Patch \keepthetitle:

                        6 \let\LWR@origkeepthetitle\keepthetitle
                        7 \renewcommand*{\keepthetitle}{%
                        8 \LWR@orig@keepthetitle%
                        9 }

\killtitle   Patch \killtitle:

                        10 \let\LWR@origkilltitle\killtitle
                        11 \renewcommand*{\killtitle}{%
                        12 \LWR@orig@killtitle%
                        13 }

titlingpage (*env.*)

                        14 \renewenvironment*{titlingpage}
                        15 {%

                        Start an HTML titlepage div:

                        16 \LWR@printpendingfootnotes
                        17 \begin{titlepage}

Prepare for a custom version of \maketitle inside the titlingpage:

```
18 \LWR@maketitlesetup
19 \let\maketitle\LWR@titlingmaketitle
20 }
21 {
```

At the end of the environment, end the HTML titlepage div:

```
22 \end{titlepage}
23 }
```

Patch the pre/post title/author/date to add HTML tags, then initilize:

```
24 \AtBeginDocument{
25     \pretitle{}
26     \posttitle{}
27
28     \preauthor{}
29     \postauthor{}
30
31     \predate{}
32     \postdate{}
33 }
```

\LWR@maketitlesetup Patches \thanks macros.

```
34 \renewcommand*{\LWR@maketitlesetup}{%
```

Redefine the footnote mark:

```
35     \def\@makefnmark{\textsuperscript{\@thefnmark}}%
```

> \thefootnote ⟹ \nameuse{arabic}{footnote}, or
> \thefootnote ⟹ \nameuse{fnsymbol}{footnote}

Redefine the footnote text:

```
36     \long\def\@makefntext##1{%
```

Make the footnote mark and some extra horizontal space for the tags:

```
37         \makethanksmark~%
```

> \makethanksmark ⟹ \thanksfootmark ⟹ \tamark ⟹
>   \@thefnmark ⟹ \itshape a (or similar)

Print the text:

```
38         {##1}%
39     }% \@makefntext
40 }
```

\thanksfootmark

```
41 \renewcommand{\thanksfootmark}{%
42 %    \hb@xt@\thanksmarkwidth{\hfil\normalfont%
43        \thanksscript{%
44            \thanksfootpre \tamark \thanksfootpost%
45        }%
46 %    }%
47 }
```

\maketitle HTML mode. Creates an HTML titlepage div and typesets the title, etc.

Code from the titling package is adapted, simplified, and modified for HTML output.

```
48 \renewcommand*{\maketitle}{%
```

An HTML titlepage <div> is used for all classes.

```
49    \begin{titlepage}
```

Select which kind of footnote marks to use:

```
50    \@bsmarkseries
```

Set up special patches:

```
51    \LWR@maketitlesetup
```

Typeset the title, etc:

```
52    \@maketitle
```

Immediately generate any \thanks footnotes:

```
53    \LWR@stoppars\@thanks\LWR@startpars
```

Close the HTML titlepage div:

```
54    \end{titlepage}
```

Reset the footnote counter:

```
55    \@bscontmark
56 }
```

\@maketitle Typesets the title, etc. Patched for HTML.

```
57 \providecommand*{\@maketitle}{}
58 \renewrobustcmd\@maketitle{%
59    \maketitlehooka
60    {
61        \LWR@stoppars\LWR@htmltag{\LWR@tagtitle}%
62        \@bspretitle \@title \@bsposttitle%
63        \LWR@htmltag{\LWR@tagtitleend}\LWR@startpars%
64    }
```

```
65        \maketitlehookb
66        {
67            \begin{BlockClass}{author}
68            \renewcommand{\and}{%
69                \end{BlockClass}%
70                \begin{BlockClass}{oneauthor}%
71            }
72            \begin{BlockClass}{oneauthor}%
73            \@bspreauthor \@author  \@bspostauthor%
74            \end{BlockClass}%
75            \end{BlockClass}%
76        }
77        \maketitlehookc
78        {
79            \begin{BlockClass}{titledate}%
80            \@bspredate \@date \@bspostdate%
81            \end{BlockClass}%
82        }
83        \maketitlehookd
84 }
```

\LWR@titlingmaketitle \maketitle for use inside an HTML titlingpage environment.

```
85 \renewcommand*{\LWR@titlingmaketitle}{%
```

Keep pending footnotes out of the title block:

```
86 \LWR@stoppars\@thanks\LWR@startpars
```

Select which kind of footnote marks to use:

```
87 \@bsmarkseries
```

Set up special patches:

```
88 \LWR@maketitlesetup
```

Typeset the title, etc:

```
89 \@maketitle
```

Immediately generate any \thanks footnotes:

```
90 \LWR@stoppars\@thanks\LWR@startpars
```

Reset the footnote counter:

```
91 \@bscontmark
92 }
```

\thanksmarkseries {⟨series⟩}

Sets the type of footnote marks used by \thanks, where type is 'arabic', 'roman', 'fnsymbol', etc.

```
93 \renewcommand{\thanksmarkseries}[1]{%
94 \def\@bsmarkseries{\renewcommand{\thefootnote}{\@nameuse{#1}{footnote}}}%
95 }
```

Set default titlepage thanks footnote marks. See section 71.7.

```
96 \IfClassLoadedTF{memoir}{
97   \thanksmarkseries{arabic}
98 }{% not memoir
99 \if@titlepage
100   \thanksmarkseries{arabic}
101 \else
102   \thanksmarkseries{fnsymbol}
103 \fi
104 }% not memoir
```

File 522 **`lwarp-tocbasic.sty`**

§ 634   Package   # tocbasic

(*Emulates or patches code by* MARKUS KOHM.)

tocbasic (*Pkg*)   tocbasic is nullified for lwarp.

This package may be loaded standalone, but is also loaded automatically if koma-script classes are in use. \DeclareDocumentCommand is used to overwrite the koma-script definitions.

for HTML output: 1 \LWR@ProvidesPackagePass{tocbasic}[2018/12/30]

```
2 \DeclareDocumentCommand{\usetocbasicnumberline}{o}{}
3 \DeclareDocumentCommand{\DeclareTOCStyleEntry}{o m m}{}
4 \DeclareDocumentCommand{\DeclareTOCStyleEntries}{o m m}{}
5 \DeclareDocumentCommand{\DeclareTOCEntryStyle}{m o m}{}
6 \DeclareDocumentCommand{\DefineTOCEntryOption}{m o m}{}
7 \DeclareDocumentCommand{\DefineTOCEntryBooleanOption}{m o m m m}{}
8 \DeclareDocumentCommand{\DefineTOCEntryCommandOption}{m o m m m}{}
9 \DeclareDocumentCommand{\DefineTOCEntryIfOption}{m o m m m}{}
10 \DeclareDocumentCommand{\DefineTOCEntryLengthOption}{m o m m m}{}
11 \DeclareDocumentCommand{\DefineTOCEntryNumberOption}{m o m m m}{}
12 \DeclareDocumentCommand{\CloneTOCEntryStyle}{m m}{}
13 \DeclareDocumentCommand{\TOCEntryStyleInitCode}{m m}{}
14 \DeclareDocumentCommand{\TOCEntryStyleStartInitCode}{m m}{}
```

File 523   **`lwarp-tocbibind.sty`**

§ 635   Package   # tocbibind

(*Emulates or patches code by* PETER WILSON.)

tocbibind (*Pkg*)   tocbibind is patched for use by lwarp.

placement and TOC options   An index may be placed inline with other HTML text, or on its own HTML page:

makeidx (*Pkg*)   **Inline, with a manual TOC entry:**

A commonly-used method to introduce an index in a LaTeX document:

```
\cleardoublepage
\phantomsection
\addcontentsline{toc}{section}{\indexname}% or chapter
\printindex
```

makeidx (*Pkg*) **On its own HTML page, with a manual TOC entry:**

```
\begin{warpprint}
\cleardoublepage
\phantomsection
\addcontentsline{toc}{section}{\indexname}% or chapter
\end{warpprint}
\ForceHTMLPage
\ForceHTMLTOC
\printindex
```

tocbibind (*Pkg*) **Inline, with an automatic TOC entry:**

The tocbibind package may be used to automatically place an entry in the TOC.

```
\usepackage[nottoc]{tocbibind}
...
\cleardoublepage
\phantomsection % to fix print-version index link
\printindex
```

tocbibind (*Pkg*) **On its own HTML page, with an automatic TOC entry:**

```
\usepackage[nottoc]{tocbibind}
...
\cleardoublepage
\phantomsection % to fix print-version index link
\ForceHTMLPage
\printindex
```

numindex (*Opt*) [tocbibind]
numbered index section
Use the tocbibind numindex option to generate a numbered index. Without this option, the index heading has no number.

Other packages, such as imakeidx, may also have options for including the index in the Table of Contents.

for HTML output:
```
1 \let\simplechapterdelim\relax
2
3 \LWR@ProvidesPackagePass{tocbibind}[2010/10/13]

4 \renewenvironment{theindex}%
5 {%
6     \if@bibchapter
7         \if@donumindex
8             \chapter{\indexname}
9         \else
10           \if@dotocind
11             \chapter*{\indexname}
12             \addcontentsline{toc}{chapter}{\LWR@isolate{\indexname}}}
13           \else
14             \chapter*{\indexname}
15           \fi
16         \fi
17     \else
18         \if@donumindex
19             \section{\indexname}
```

```
20        \else
21          \if@dotocind
22            \section*{\indexname}
23            \addcontentsline{toc}{\@tocextra}{\LWR@isolate{\indexname}}
24          \else
25            \section*{\indexname}
26          \fi
27        \fi
28      \fi
29 \LetLtxMacro\item\LWR@indexitem%
30 \LetLtxMacro\subitem\LWR@indexsubitem%
31 \LetLtxMacro\subsubitem\LWR@indexsubsubitem%
32 }{}
```

The following code is shared by anonchap.

```
33 \DeclareDocumentCommand{\simplechapter}{O{\@empty}}{%
34     \def\@chapcntformat##1{%
35         #1~\csname the##1\endcsname\simplechapterdelim\quad%
36     }%
37 }
38
39 \DeclareDocumentCommand{\restorechapter}{}{%
40 \let\@chapcntformat\@seccntformat%
41 }
```

File 524   **`lwarp-tocdata.sty`**

§ 636   Package   **tocdata**

(*Emulates or patches code by* Brian Dunn.)

tocdata (*Pkg*)   tocdata is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{tocdata}[2019/07/06]

```
2 \renewcommand*{\LWR@maybetocdata}{%
3     \ifdefempty{\TD@thistocdata}{}{%
4         \qquad \InlineClass{authorartist}{\tocdataformat{\TD@thistocdata}}%
5         \def\TD@thistocdata{}
6     }
7 }
```

```
8 \renewrobustcmd{\tocdatapartprint}[4]
9 {%
10     \InlineClass{authorartist}{%
11         \qquad --- %
12         \TDoptionalnameprint{#1}\TDoptionalnameprint{#2}#3#4%
13     }%
14 }
15
16 \@ifundefined{chapter}{}{
17     \let\tocdatachapterprint\tocdatapartprint
18 }
19 \let\tocdatasectionprint\tocdatapartprint
```

```
20 \let\tocdatasubsectionprint\tocdatapartprint
21
22 \newcommand*{\LWR@TD@settextalign}[1]{%
23     \def\LWR@TD@textalign{justify}%
24     \ifcsstring{TD@#1align}{\centering}%
25         {\def\LWR@TD@textalign{center}}%
26         {}%
27     \ifcsstring{TD@#1align}{\raggedleft}%
28         {\def\LWR@TD@textalign{right}}%
29         {}%
30     \ifcsstring{TD@#1align}{\raggedright}%
31         {\def\LWR@TD@textalign{left}}%
32         {}%
33 }
34
35 \renewcommand{\TDartistauthorprint}[5]{%
36     \LWR@TD@settextalign{#1}%
37     \begin{BlockClass}[text-align:\LWR@TD@textalign]{floatnotes}%
38     \InlineClass{authorartist}{\TDoptionalnameprint{#2}\TDoptionalnameprint{#3}#4#5}%
39     \end{BlockClass}%
40 }
41
42 \newcommand*{\LWR@TD@setnamealign}[1]{%
43     \def\LWR@TD@textalign{justify}%
44     \ifcsstring{TD@#1textalign}{\centering}%
45         {\def\LWR@TD@textalign{center}}%
46         {}%
47     \ifcsstring{TD@#1textalign}{\raggedleft}%
48         {\def\LWR@TD@textalign{right}}%
49         {}%
50     \ifcsstring{TD@#1textalign}{\raggedright}%
51         {\def\LWR@TD@textalign{left}}%
52         {}%
53 }
54
55 \renewcommand{\TDartistauthortextprint}[2]{%
56     \LWR@TD@setnamealign{#1}%
57     \begin{BlockClass}[text-align:\LWR@TD@textalign]{floatnotes}%
58     #2%
59     \end{BlockClass}%
60 }
```

File 525   `lwarp-tocenter.sty`

§ 637   Package   **tocenter**

tocenter (*Pkg*)   **tocenter** is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{tocenter}[2004/12/09]

2 \NewDocumentCommand{\ToCenter}{s o m m}{}
3 \NewDocumentCommand{\FromMargins}{s o m m m m}{}

File 526   **`lwarp-tocloft.sty`**

§ 638   Package   **tocloft**

(*Emulates or patches code by* PETER WILSON.)

tocloft (*Pkg*)   **tocloft** is emulated. Most user options and macros are ignored and disabled.
\newlistof and \cftchapterprecis are supported.

tocloft (*Pkg*)   If using **tocloft** with **tocbibind**, **anonchap**, **fncychap**, or other packages which
⚠ tocloft & other packages   change chapter title formatting, load **tocloft** with its titles option, which tells
tocloft to use standard LATEX commands to create the titles, allowing other pack-
ages to work with it.

Discard all options for lwarp-tocloft:

for HTML output: 1 \LWR@ProvidesPackageDrop{tocloft}[2017/08/31]

\tocloftpagestyle   {⟨*style*⟩}

2 \newcommand{\tocloftpagestyle}[1]{}

\cftmarktoc

3 \newcommand*{\cftmarktoc}{}

\cfttoctitlefont

4 \newcommand*{\cfttoctitlefont}{}

\cftaftertoctitle

5 \newcommand*{\cftaftertoctitle}{}

6 \newlength{\cftbeforetoctitleskip}
7 \newlength{\cftaftertoctitleskip}

\cftmarklof

8 \newcommand*{\cftmarklof}{}

\cftloftitlefont

9 \newcommand*{\cftloftitlefont}{}

\cftafterloftitle

10 \newcommand*{\cftafterloftitle}{}

11 \newlength{\cftbeforeloftitleskip}
12 \newlength{\cftafterloftitleskip}

\cftmarklot

13 \newcommand*{\cftmarklot}{}

\cftlottitlefont

14 \newcommand*{\cftlottitlefont}{}

\cftafterlottitle

15 \newcommand*{\cftafterlottitle}{}

16 \newlength{\cftbeforelottitleskip}
17 \newlength{\cftafterlottitleskip}

\cftdot

18 \providecommand*{\cftdot}{.}

\cftdotsep

19 \providecommand*{\cftdotsep}{1}

\cftnodots

20 \providecommand*{\cftnodots}{5000}

\cftdotfill   {⟨*sep*⟩}

21 \providecommand{\cftdotfill}[1]{}

\cftsetpnumwidth   {⟨*length*⟩}

22 \DeclareDocumentCommand{\cftsetpnumwidth}{m}{}

\cftsetrmarg   {⟨*length*⟩}

23 \DeclareDocumentCommand{\cftsetrmarg}{m}{}

\cftpnumalign   {⟨*alignment*⟩}

24 \DeclareDocumentCommand{\cftpnumalign}{m}{}

25 \LWR@providelength{\cftparskip}

The part-related items are also provided by memoir:

26 \LWR@providelength{\cftbeforepartskip}
27 \LWR@providelength{\cftpartindent}
28 \LWR@providelength{\cftpartnumwidth}
29 \providecommand*{\cftpartfont}{}
30 \providecommand*{\cftpartpresnum}{}
31 \providecommand*{\cftpartaftersnum}{}
32 \providecommand*{\cftpartaftersnumb}{}
33 \providecommand*{\cftpartleader}{}

```
34 \providecommand*{\cftpartdotsep}{1}
35 \providecommand*{\cftpartpagefont}{}
36 \providecommand*{\cftpartafterpnum}{}
```

memoir uses the full name "chapter" instead of "chap":

```
37 \LWR@providelength{\cftbeforechapskip}
38 \LWR@providelength{\cftchapindent}
39 \LWR@providelength{\cftchapnumwidth}
40 \newcommand*{\cftchapfont}{}
41 \newcommand*{\cftchappresnum}{}
42 \newcommand*{\cftchapaftersnum}{}
43 \newcommand*{\cftchapaftersnumb}{}
44 \newcommand*{\cftchapleader}{}
45 \newcommand*{\cftchapdotsep}{1}
46 \newcommand*{\cftchappagefont}{}
47 \newcommand*{\cftchapafterpnum}{}
```

The following do not appear in memoir:

```
48 \LWR@providelength{\cftbeforesecskip}
49 \LWR@providelength{\cftsecindent}
50 \LWR@providelength{\cftsecnumwidth}
51 \newcommand*{\cftsecfont}{}
52 \newcommand*{\cftsecpresnum}{}
53 \newcommand*{\cftsecaftersnum}{}
54 \newcommand*{\cftsecaftersnumb}{}
55 \newcommand*{\cftsecleader}{}
56 \newcommand*{\cftsecdotsep}{1}
57 \newcommand*{\cftsecpagefont}{}
58 \newcommand*{\cftsecafterpnum}{}
```

```
59 \LWR@providelength{\cftbeforesubsecskip}
60 \LWR@providelength{\cftsubsecindent}
61 \LWR@providelength{\cftsubsecnumwidth}
62 \newcommand*{\cftsubsecfont}{}
63 \newcommand*{\cftsubsecpresnum}{}
64 \newcommand*{\cftsubsecaftersnum}{}
65 \newcommand*{\cftsubsecaftersnumb}{}
66 \newcommand*{\cftsubsecleader}{}
67 \newcommand*{\cftsubsecdotsep}{1}
68 \newcommand*{\cftsubsecpagefont}{}
69 \newcommand*{\cftsubsecafterpnum}{}
```

```
70 \LWR@providelength{\cftbeforesubsubsecskip}
71 \LWR@providelength{\cftsubsubsecindent}
72 \LWR@providelength{\cftsubsubsecnumwidth}
73 \newcommand*{\cftsubsubsecfont}{}
74 \newcommand*{\cftsubsubsecpresnum}{}
75 \newcommand*{\cftsubsubsecaftersnum}{}
76 \newcommand*{\cftsubsubsecaftersnumb}{}
77 \newcommand*{\cftsubsubsecleader}{}
78 \newcommand*{\cftsubsubsecdotsep}{1}
79 \newcommand*{\cftsubsubsecpagefont}{}
80 \newcommand*{\cftsubsubsecafterpnum}{}
```

```
81 \LWR@providelength{\cftbeforeparaskip}
82 \LWR@providelength{\cftparaindent}
83 \LWR@providelength{\cftparanumwidth}
```

```
 84 \newcommand*{\cftparafont}{}
 85 \newcommand*{\cftparapresnum}{}
 86 \newcommand*{\cftparaaftersnum}{}
 87 \newcommand*{\cftparaaftersnumb}{}
 88 \newcommand*{\cftparaleader}{}
 89 \newcommand*{\cftparadotsep}{1}
 90 \newcommand*{\cftparapagefont}{}
 91 \newcommand*{\cftparaafterpnum}{}

 92 \LWR@providelength{\cftbeforesubparaskip}
 93 \LWR@providelength{\cftsubparaindent}
 94 \LWR@providelength{\cftsubparanumwidth}
 95 \newcommand*{\cftsubparafont}{}
 96 \newcommand*{\cftsubparapresnum}{}
 97 \newcommand*{\cftsubparaaftersnum}{}
 98 \newcommand*{\cftsubparaaftersnumb}{}
 99 \newcommand*{\cftsubparaleader}{}
100 \newcommand*{\cftsubparadotsep}{1}
101 \newcommand*{\cftsubparapagefont}{}
102 \newcommand*{\cftsubparaafterpnum}{}

103 \LWR@providelength{\cftbeforefigskip}
104 \LWR@providelength{\cftfigindent}
105 \LWR@providelength{\cftfignumwidth}
106 \newcommand*{\cftfigfont}{}
107 \newcommand*{\cftfigpresnum}{}
108 \newcommand*{\cftfigaftersnum}{}
109 \newcommand*{\cftfigaftersnumb}{}
110 \newcommand*{\cftfigleader}{}
111 \newcommand*{\cftfigdotsep}{1}
112 \newcommand*{\cftfigpagefont}{}
113 \newcommand*{\cftfigafterpnum}{}

114 \LWR@providelength{\cftbeforesubfigskip}
115 \LWR@providelength{\cftsubfigindent}
116 \LWR@providelength{\cftsubfignumwidth}
117 \newcommand*{\cftsubfigfont}{}
118 \newcommand*{\cftsubfigpresnum}{}
119 \newcommand*{\cftsubfigaftersnum}{}
120 \newcommand*{\cftsubfigaftersnumb}{}
121 \newcommand*{\cftsubfigleader}{}
122 \newcommand*{\cftsubfigdotsep}{1}
123 \newcommand*{\cftsubfigpagefont}{}
124 \newcommand*{\cftsubfigafterpnum}{}

125 \LWR@providelength{\cftbeforetabskip}
126 \LWR@providelength{\cfttabindent}
127 \LWR@providelength{\cfttabnumwidth}
128 \newcommand*{\cfttabfont}{}
129 \newcommand*{\cfttabpresnum}{}
130 \newcommand*{\cfttabaftersnum}{}
131 \newcommand*{\cfttabaftersnumb}{}
132 \newcommand*{\cfttableader}{}
133 \newcommand*{\cfttabdotsep}{1}
134 \newcommand*{\cfttabpagefont}{}
135 \newcommand*{\cfttabafterpnum}{}

136 \LWR@providelength{\cftbeforesubtabskip}
137 \LWR@providelength{\cftsubtabindent}
```

```
138 \LWR@providelength{\cftsubtabnumwidth}
139 \newcommand*{\cftsubtabfont}{}
140 \newcommand*{\cftsubtabpresnum}{}
141 \newcommand*{\cftsubtabaftersnum}{}
142 \newcommand*{\cftsubtabaftersnumb}{}
143 \newcommand*{\cftsubtableader}{}
144 \newcommand*{\cftsubtabdotsep}{1}
145 \newcommand*{\cftsubtabpagefont}{}
146 \newcommand*{\cftsubtabafterpnum}{}
```

```
147 \DeclareDocumentCommand{\cftsetindents}{m m m}{}
```

```
148 \providecommand{\cftpagenumbersoff}[1]{}
149 \providecommand{\cftpagenumberson}[1]{}
```

\newlistentry   [⟨*within*⟩] {⟨*counter*⟩} {⟨*ext*⟩} {⟨*level-1*⟩}

```
150 \DeclareDocumentCommand{\newlistentry}{o m m m}
151 {%
152 \LWR@traceinfo{newlistentry #2 #3 #4}%
153 \IfValueTF{#1}%
154 {%
155     \@ifundefined{c@#2}{%
156         \newcounter{#2}[#1]%
157         \expandafter\edef\csname the#2\endcsname{%
158           \expandafter\noexpand\csname the#1\endcsname.\noexpand\arabic{#2}%
159         }%
160     }{}%
161 }%
162 {%
163     \@ifundefined{c@#2}{%
164         \newcounter{#2}%
165     }{}%
166 }%
167 \@namedef{l@#2}##1##2{%
168     \hypertocfloat{1}{#2}{#3}{##1}{##2}%
169     \def\cftwhatismyname{#2}% from memoir
170 }%
171 \expandafter\newlength\csname cftbefore#2skip\endcsname%
172 \expandafter\newlength\csname cft#2indent\endcsname%
173 \expandafter\newlength\csname cft#2numwidth\endcsname%
174 \@namedef{cft#2font}{}%
175 \@namedef{cft#2presnum}{}%
176 \@namedef{cft#2aftersnum}{}%
177 \@namedef{cft#2aftersnumb}{}%
178 \@namedef{cft#2leader}{}%
179 \@namedef{cft#2dotsep}{1}%
180 \@namedef{cft#2pagefont}{}%
181 \@namedef{cft#2afterpnum}{}%
182 \@namedef{toclevel@#2}{#4}%
183 \@namedef{cft#2fillnum}##1{}%
184 \LWR@traceinfo{newlistentry done}%
185 }
```

\newlistof   [⟨*within*⟩] {⟨*type*⟩} {⟨*ext*⟩} {⟨*listofname*⟩}

Emulated through the \newfloat mechanism.

```
186 \DeclareDocumentCommand{\newlistof}{o m m m}
```

```
187 {%
188     \IfValueTF{#1}%
189         {\newlistentry[#1]{#2}{#3}{0}}%
190         {\newlistentry{#2}{#3}{0}}%
191     \@namedef{ext@#2}{#3}%
192     \@ifundefined{c@#3depth}{\newcounter{#3depth}}{}%
193     \setcounter{#3depth}{1}%
194     \@namedef{cftmark#3}{}%
195     \@namedef{listof#2}{\LWR@listof{#2}{#4}}%
196     \@namedef{@cftmake#3title}{}%
197     \expandafter\newlength\csname cftbefore#3titleskip\endcsname%
198     \expandafter\newlength\csname cftafter#3titleskip\endcsname%
199     \@namedef{cft#3titlefont}{}%
200     \@namedef{cftafter#3title}{}%
201     \@namedef{cft#3prehook}{}%
202     \@namedef{cft#3posthook}{}%
203 }
```

\cftchapterprecis   {⟨*text*⟩}

```
204 \newcommand{\cftchapterprecis}[1]{%
205   \cftchapterprecishere{#1}
206   \cftchapterprecistoc{#1}}
207 \newcommand{\cftchapterprecishere}[1]{%
208   \begin{quote}\textit{#1}\end{quote}}
209 \newcommand{\cftchapterprecistoc}[1]{
210   \addtocontents{toc}{%
211   {
212      \protect\begin{quote}#1\protect\end{quote}}
213   }
214 }
```

File 527   **lwarp-tocstyle.sty**

§ 639   Package   # tocstyle

tocstyle (*Pkg*)   tocstyle is ignored.

⚠ Not fully tested!   Please send bug reports!

for HTML output: 1 \LWR@ProvidesPackageDrop{tocstyle}[2017/02/23]

```
2 \newcommand*{\usetocstyle}[2][]{}
3 \newcommand*{\deactivatetocstyle}[1][]{}
4 \newcommand*{\reactivatetocstyle}[1][]{}
5 \NewDocumentCommand{\settocfeature}{o o m m}{}
6 \NewDocumentCommand{\settocstylefeature}{o m m}{}
7 \NewDocumentCommand{\newtocstyle}{o o m m}{}
8 \newcommand*{\aliastoc}[2]{}
9 \newcommand*{\showtoc}[2][]{}
10 \newcommand{\iftochasdepth}[4]{}
```

File 528   `lwarp-todo.sty`

§ 640   Package **todo**

(*Emulates or patches code by* FEDERICO GARCIA.)

todo (*Pkg*)   todo is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{todo}[2010/03/31]


```
2 \renewcommand\todoitem[2]{%
3     \refstepcounter{todo}%
4     \item[%
5         \HTMLunicode{2610} \quad%
6         \ref{todopage:\thetodo}
7      ] : {\todoformat\ifx#1\todomark\else\textbf{#1} \fi}#2%
8     \label{todolbl:\thetodo}%
9 }%
10
11 \renewcommand\doneitem[2]{%
12     \stepcounter{todo}%
13     \item[%
14         \HTMLunicode{2611} \quad%
15         \ref{todopage:\thetodo}
16     ] \@nameuse{@done\the\c@todo}:
17         {\todoformat\ifx#1\todomark\else\textbf{#1} \fi}#2%
18 }
```


The following are not errors because the code will still compile and be usable if
the patch is not possible.


```
19 \xpatchcmd{\@displaytodo}
20     {\todoformat #1}{\todoformat \textbf{#1}}{}
21     {\PackageWarning{lwarp-todo}{Unable to patch @displaytodo.}}
22
23 \xpatchcmd{\@displayfulltodo}
24     {\todoformat #1}{\todoformat \textbf{#1}}{}
25     {\PackageWarning{lwarp-todo}{Unable to patch @displayfulltodo.}}
26
27 \patchcmd{\todoenv}{\itshape see text.}{\textit{see text.}}{}
28     {\PackageWarning{lwarp-todo}{Unable to patch todoenv.}}
29
30 \patchcmd{\astodos}{\todoformat #1}{\todoformat \textbf{#1}}{}
31     {\PackageWarning{lwarp-todo}{Unable to patch astodos.}}
```


If cleveref is in use, name the new todo notes:


```
32 \AtBeginDocument{
33 \ifdef{\crefname}{
34     \crefname{todo}{todo}{todos}
35     \Crefname{todo}{Todo}{Todos}
36 }{}
37 }
```

File 529 **`lwarp-todonotes.sty`**

§ 641   Package   **todonotes**

(*Emulates or patches code by* Henrik Skov Midtiby.)

todonotes (*Pkg*)   **todonotes** is emulated.

The documentation for **todonotes** and **luatodonotes** have an example with a todo inside a caption. If this example does not work it will be necessary to move the todo outside of the caption.

for HTML output: 1 `\LWR@ProvidesPackagePass{todonotes}[2012/07/25]`

2 `\if@todonotes@disabled`
3 `\else`
4
5 `\newcommand{\ext@todo}{tdo}`
6
7 `\renewcommand{\l@todo}[2]{\hypertocfloat{1}{todo}{ldo}{#1}{#2}}`

8 `\let\LWRTODONOTES@orig@todototoc\todototoc`
9
10 `\renewcommand*{\todototoc}{%`
11 `\LWR@phantomsection%`
12 `\LWRTODONOTES@orig@todototoc%`
13 `}`
14
15 `\renewcommand{\@todonotes@drawMarginNoteWithLine}{`
16 `\fcolorbox`
17 `    {\@todonotes@currentbordercolor}`
18 `    {\@todonotes@currentbackgroundcolor}`
19 `    {\arabic{@todonotes@numberoftodonotes}}`
20 `\marginpar{\@todonotes@drawMarginNote}`
21 `}`
22
23 `\renewcommand{\@todonotes@drawInlineNote}{%`
24 `\fcolorboxBlock%`
25 `    {\@todonotes@currentbordercolor}%`
26 `    {\@todonotes@currentbackgroundcolor}%`
27 `    {%`
28 `        \if@todonotes@authorgiven%`
29 `        {\@todonotes@author:\,}%`
30 `        \fi%`
31 `        \@todonotes@text%`
32 `    }%`
33 `}`
34
35 `\renewcommand{\@todonotes@drawMarginNote}{%`
36 `    \if@todonotes@authorgiven%`
37 `        \@todonotes@author\par%`
38 `    \fi%`
39 `    \arabic{@todonotes@numberoftodonotes}: %`
40 `    \fcolorbox%`
41 `    {\@todonotes@currentbordercolor}%`

```
42      {\@todonotes@currentbackgroundcolor}%
43      {%
44          \@todonotes@sizecommand%
45          \@todonotes@text %
46      }%
47 }%
48
49 \renewcommand{\@todonotes@drawLineToRightMargin}{}
50
51 \renewcommand{\@todonotes@drawLineToLeftMargin}{}
52
53 \renewcommand{\missingfigure}[2][]{%
54 \setkeys{todonotes}{#1}%
55 \addcontentsline{tdo}{todo}{\@todonotes@MissingFigureText: #2}%
56 \fcolorboxBlock%
57      {\@todonotes@currentbordercolor}%
58      {\@todonotes@currentfigcolor}%
59      {%
60          \setlength{\fboxrule}{4pt}%
61          \fcolorbox{red}{white}{Missing figure} \quad #2%
62      }
63 }
64
65 \LetLtxMacro\LWRTODONOTES@orig@todo\@todo
66
67 \RenewDocumentCommand{\@todo}{o m}{%
68 \begingroup%
69 \renewcommand*{\phantomsection}{}%
70 \IfValueTF{#1}{%
71      \LWRTODONOTES@orig@todo[#1]{#2}%
72 }{%
73      \LWRTODONOTES@orig@todo{#2}%
74 }
75 \endgroup%
76 }
77
78 \fi% \if@todonotes@disabled
```

File 530   **lwarp-topcapt.sty**

§ 642   Package   # topcapt

topcapt (*Pkg*)   topcapt is emulated.

for HTML output: 1 \LWR@ProvidesPackageDrop{topcapt}[2004/12/11]

2 \LetLtxMacro\topcaption\caption

File 531   **lwarp-tram.sty**

§ 643   Package   # tram

tram (*Pkg*)   tram is emulated.

⚠ block only   The HTML emulation uses a <div>, which must not appear inside an HTML <span>

or an HTML paragraph. For this reason, the tram environment should only be used to contain paragraphs inside a \parbox or minipage. tram should not be used to mark up inline text.

To disable tram, allowing source compatibility with inline uses:

```
\begin{warpHTML}
\renewenvironment{tram}[1][]{}{}
\end{warpHTML}
```

for HTML output: 1 \LWR@ProvidesPackageDrop{tram}[2013/04/04]

```
2 \newenvironment{tram}[1][]%
3     {\BlockClass[background:lightgray]{tram}}
4     {\endBlockClass}
```

File 532   **`lwarp-transparent.sty`**

§ 644   Package   # transparent

(*Emulates or patches code by* Heiko Oberdiek.)

transparent (*Pkg*)   transparent is emulated. \texttransparent works for inline objects. \transparent only works for \includegraphics.

⚠   Not X∃LATEX!   Note that transparent does not work with X∃LATEX.

for HTML output: 1 \LWR@ProvidesPackagePass{transparent}[2019/11/29]

```
2 \newcommand*{\LWR@HTML@transparent}[1]{\edef\LWR@opacity{#1}}
3
4 \LWR@formatted{transparent}
5
6
7 \newcommand*{\LWR@HTML@texttransparent}[2]{%
8 \begingroup%
9 \transparent{#1}%
10 \InlineClass[opacity: #1]{transparent}{#2}%
11 \endgroup%
12 }
13
14 \LWR@formatted{texttransparent}
```

File 533   **`lwarp-trimclip.sty`**

§ 645   Package   # trimclip

trimclip (*Pkg*)   trimclip is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{trimclip}[2018/04/08]

The third argument, the text, is not touched. This allows \bgroup / \egroup, and verbatim content.

```
 2 \csdef{trimbox}{\@ifstar\@gobble\@gobble}
 3 \csletcs{trimbox*}{trimbox}
 4 \def\endtrimbox{}
 5 \csletcs{endtrimbox*}{endtrimbox}
 6
 7 \csletcs{clipbox}{trimbox}
 8 \csletcs{clipbox*}{trimbox}
 9 \csletcs{endclipbox}{endtrimbox}
10 \csletcs{endclipbox*}{endtrimbox}
11
12 \csletcs{marginbox}{trimbox}
13 \csletcs{marginbox*}{trimbox}
14 \csletcs{endmarginbox}{endtrimbox}
15 \csletcs{endmarginbox*}{endtrimbox}
```

File 534   **`lwarp-trivfloat.sty`**

§ 646   Package   **trivfloat**

(*Emulates or patches code by* JOSEPH WRIGHT.)

trivfloat (*Pkg*)   **trivfloat** is forced to use the built-in **lwarp** emulation for floats.

To create a new float type and change its name:

```
\trivfloat{example}
\renewcommand{\examplename}{Example Name}
\crefname{example}{example}{examples}
\Crefname{example}{Example}{Examples}
```

Discard all options for **lwarp-trivfloat**. This tells **trivfloat** not to use **floatrow** or **memoir**.

```
1 \LWR@ProvidesPackageDrop{trivfloat}[2009/04/23]
2 \LWR@origRequirePackage{trivfloat}
```

\tfl@chapter@fix   Nullified at the beginning of the document. Is used by **trivfloat** to correct float chapter numbers, but is not needed for **lwarp**.

```
3 \AtBeginDocument{\DeclareDocumentCommand{\tfl@chapter@fix}{m m}{}}
```

§ 646.1   **Combining \newfloat, \trivfloat, and algorithmicx**

For both print and HTML output:

⚠   When using **float**, **trivfloat**, or **algorithmicx** at the same time, be aware of conflicting file usage. **algorithmicx** uses `.loa`. **trivfloat** by default starts with `.loa` and goes up for additional floats, skipping `.lof` and `.lot`.

⚠   When using \newfloat, be sure to manually assign higher letters to the \newfloat files to avoid `.loa` used by **algorithmicx**, and any files used by **trivfloat**. Also avoid using `.lof` and `.lot`.

⚠   When using \trivfloat, you may force it to avoid conflicting with **algorithmicx** by starting **trivfloat**'s file extensions with `.lob`:

```
\makeatletter
\setcounter{tfl@float@cnt}{1} % start trivfloats with .lob
\makeatletter
```

File 535 **lwarp-truncate.sty**

§ 647　Package **truncate**

truncate (*Pkg*)　truncate is ignored.

for HTML output:　1 `\LWR@ProvidesPackageDrop{truncate}[2001/08/20]`

2 `\providecommand{\TruncateMarker}{}`
3 `\newcommand{\truncate}[3][\TruncateMarker]{#3}`

File 536 **lwarp-turnthepage.sty**

§ 648　Package **turnthepage**

turnthepage (*Pkg*)　turnthepage is ignored.

for HTML output:　1 `\LWR@ProvidesPackageDrop{turnthepage}[2011/03/24]`

2 `\newcommand{\turnthepage}{}`

File 537 **lwarp-twoup.sty**

§ 649　Package **twoup**

twoup (*Pkg*)　twoup is ignored.

for HTML output:　1 `\LWR@ProvidesPackageDrop{twoup}[2007/02/26]`

2 `\newcommand{\cleartolastpage}{}`

File 538 **lwarp-txfonts.sty**

§ 650　Package **txfonts**

(*Emulates or patches code by* Young Ryu.)

txfonts (*Pkg*)　txfonts is used as-is for svg math, and is emulated for MathJax.

for HTML output:　1 `\LWR@ProvidesPackagePass{txfonts}[2008/01/22]`

For MATHJAX:

```
2 \LWR@origRequirePackage{lwarp-common-mathjax-letters}
3
4 \begin{warpMathJax}
5 \LWR@infoprocessingmathjax{txfonts}
6
7 \LWR@mathjax@addgreek@l@up{}{up}
8 \end{warpMathJax}
```

File 539   **lwarp-txgreeks.sty**

§ 651   Package   **txgreeks**

(*Emulates or patches code by* JEAN-FRANÇOIS BURNOL.)

txgreeks (*Pkg*)   **txgreeks** is used as-is for SVG math, and is emulated for MATHJAX.

The MATHJAX emulation honors all package options.

for HTML output:
```
1 \LWR@ProvidesPackagePass{txgreeks}[2011/03/16]
2
3 \LWR@infoprocessingmathjax{txgreeks}

4 \LWR@origRequirePackage{lwarp-common-mathjax-letters}
5
6 \begin{warpMathJax}
7 \iftgs@uplower% upright lowercase Greek
8     \LWR@mathjax@addgreek@l@up{}{}
9     \LWR@mathjax@addgreek@l@it{other}{}
10 \else% italic lowercase Greek
11     \LWR@mathjax@addgreek@l@it{}{}
12     \LWR@mathjax@addgreek@l@up{other}{}
13 \fi
14
15 \iftgs@itupper % italic uppercase Greek
16     \LWR@mathjax@addgreek@u@it*{}{}
17     \LWR@mathjax@addgreek@u@up*{other}{}
18     \LWR@mathjax@addgreek@u@up*{var}{}
19 \else% upright uppercase Greek
20     \LWR@mathjax@addgreek@u@up*{}{}
21     \LWR@mathjax@addgreek@u@it*{other}{}
22     \LWR@mathjax@addgreek@u@it*{var}{}
23 \fi
24 \end{warpMathJax}
```

File 540   **lwarp-typearea.sty**

§ 652   Package   **typearea**

(*Emulates or patches code by* MARKUS KOHM.)

typearea (*Pkg*)   **typearea** is emulated.

This package may be loaded standalone, but is also loaded automatically if koma-script classes are in use. \DeclareDocumentCommand is used to overwrite the koma-script definitions.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{typearea}[2018/03/30]
```

```
2 \DeclareDocumentCommand{\typearea}{o m}{}
3 \DeclareDocumentCommand{\recalctypearea}{}{}
4 \@ifundefined{footheight}{\newlength\footheight}{}
5 \DeclareDocumentCommand{\areaset}{o m m}{}
6 \DeclareDocumentCommand{\activateareas}{}{}
7 \DeclareDocumentCommand{\storeareas}{m}{}
8 \DeclareDocumentCommand{\BeforeRestoreareas}{s m}{}
9 \DeclareDocumentCommand{\AfterRestoreareas}{s m}{}
10 \DeclareDocumentCommand{\AfterCalculatingTypearea}{s m}{}
11 \DeclareDocumentCommand{\AfterSettingArea}{s m}{}
```

File 541 **`lwarp-typicons.sty`**

§ 653 Package **typicons**

(*Emulates or patches code by* ARTHUR VIGIL, XAVIER DANAUX.)

typicons (*Pkg*) typicons is patched for use by lwarp.

If \ticon is used, the name of the icon is used in the alt tag. Otherwise, for each of the individual icon macros, a generic alt tag is used.

for HTML output:
```
1 \LWR@ProvidesPackagePass{typicons}[2015/05/20]
```

```
2 \LetLtxMacro\LWR@orig@symbol\symbol
3
4 \let\LWR@orig@typicon@TI\TI
5
6 \newcommand*{\LWR@typicon@symbol}[1]{%
7     \begin{lateximage}*[typicon][typicon#1]%
8     \begingroup%
9     \LWR@orig@typicon@TI%
10    \LWR@orig@symbol{#1}%
11    \endgroup%
12    \end{lateximage}%
13 }
14
15 \renewcommand*{\TI}{%
16    \LetLtxMacro\symbol\LWR@typicon@symbol%
17 }
18
19 \renewcommand*{\ticon}[1]
20 {%
21    \begin{lateximage}*[#1 icon]?[typicon#1]%
22    \TI\csname ticon@#1\endcsname%
23    \end{lateximage}%
24 }
```

File 542 **`lwarp-ulem.sty`**

§ 654 Package **ulem**

(*Emulates or patches code by* DONALD ARSENEAU.)

ulem (*Pkg*) Patched for use by lwarp.

for HTML output: Use the original package:

```
1 \LWR@ProvidesPackagePass{ulem}[2012/05/18]
```

Basic markup commands, using CSS:

```
2 \NewDocumentCommand{\LWR@HTML@uline}{+m}{%
3     \InlineClass%
4         (text-decoration:underline; text-decoration-skip: auto)%
5         {uline}{\LWR@isolate{#1}}%
6 }
7 \LWR@formatted{uline}
8
9 \NewDocumentCommand{\LWR@HTML@uuline}{+m}{%
10     \InlineClass%
11         (%
12             text-decoration:underline; text-decoration-skip: auto;%
13             text-decoration-style:double%
14         )%
15         {uuline}{\LWR@isolate{#1}}%
16 }
17 \LWR@formatted{uuline}
18
19 \NewDocumentCommand{\LWR@HTML@uwave}{+m}{%
20     \InlineClass%
21         (%
22             text-decoration:underline; text-decoration-skip: auto;%
23             text-decoration-style:wavy%
24         )%
25         {uwave}{\LWR@isolate{#1}}%
26 }
27 \LWR@formatted{uwave}
28
29 \NewDocumentCommand{\LWR@HTML@sout}{+m}{%
30     \InlineClass%
31         (text-decoration:line-through)%
32         {sout}{\LWR@isolate{#1}}%
33 }
34 \LWR@formatted{sout}
35
36 \NewDocumentCommand{\LWR@HTML@xout}{+m}{%
37     \InlineClass%
38         (text-decoration:line-through)%
39         {xout}{\LWR@isolate{#1}}%
40 }
41 \LWR@formatted{xout}
42
43 \NewDocumentCommand{\LWR@HTML@dashuline}{+m}{%
```

```
44      \InlineClass%
45          (%
46              text-decoration:underline;%
47              text-decoration-skip: auto;%
48              text-decoration-style:dashed%
49          )%
50          {dashuline}{\LWR@isolate{#1}}%
51 }
52 \LWR@formatted{dashuline}
53
54 \NewDocumentCommand{\LWR@HTML@dotuline}{+m}{%
55      \InlineClass%
56          (%
57              text-decoration:underline;%
58              text-decoration-skip: auto;%
59              text-decoration-style: dotted%
60          )%
61          {dotuline}{\LWR@isolate{#1}}%
62 }
63 \LWR@formatted{dotuline}
```

Nullified/emulated macros:

```
64 \NewDocumentCommand{\LWR@HTML@markoverwith}{m}{}
65 \LWR@formatted{markoverwith}
66
67 \NewDocumentCommand{\LWR@HTML@ULon}{+m}{\uline{#1}\egroup}
68 \LWR@formatted{ULon}
```

File 543   **`lwarp-umoline.sty`**

§ 655   Package   # umoline

(*Emulates or patches code by* Hiroshi Nakashima.)

umoline (*Pkg*)   umoline is patched for use by lwarp.

for HTML output: `1 \LWR@ProvidesPackagePass{umoline}[2000/07/11]`

```
2 \newcommand*{\LWR@HTML@Underline}[1]{%
3      \InlineClass{uline}{#1}%
4 }
5 \LWR@formatted{Underline}
6
7 \newcommand*{\LWR@HTML@Midline}[1]{%
8      \InlineClass{sout}{#1}%
9 }
10 \LWR@formatted{Midline}
11
12 \newcommand*{\LWR@HTML@Overline}[1]{%
13      \InlineClass{oline}{#1}%
14 }
15 \LWR@formatted{Overline}
16
17 \newcommand*{\LWR@HTML@UMOline}[2]{%
18      \InlineClass{uline}{#2}%
19 }
```

```
20 \LWR@formatted{UMOline}
21
22 \NewDocumentCommand{\LWR@HTML@UMOspace}{s m o}{\hspace*{#2}}
23 \LWR@formatted{UMOspace}
24
25 \NewDocumentCommand{\LWR@HTML@UMOnewline}{s}{\newline}
26 \LWR@formatted{UMOnewline}
```

File 544   **`lwarp-underscore.sty`**

§ 656   Package   # underscore

underscore (*Pkg*)   underscore is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{underscore}[2006/09/13]`

File 545   **`lwarp-unicode-math.sty`**

§ 657   Package   # unicode-math

(*Emulates or patches code by* WILL ROBERTSON.)

unicode-math (*Pkg*)   unicode-math is supported as-is for HTML with svgmath.

⚠ MATHJAX   If the document source includes embedded Unicode characters, these may not be reproduced correctly for *pdftotext*, and thus not display correctly in MATHJAX.

Symbol font commands are emulated, but not all combinations are supported by MATHJAX, especially with the dedicated Greek macros. Symbol macros such as \symbfsf may not be sans or bold. For Greek, use the Unicode equivalent, if necessary.

⚠ \mathversion   The MATHJAX emulation does not change with the use of \mathversion. Whatever emulation is established at the begin of the document will remain.

The option sans-style honors upright and italic, but italic will not be sans, in order to support Greek macros.

Greek macros such as \alpha respond to the math-style option. Latin symbols does not, per MATHJAX limitations, unless placed inside \symbit or similar.

Macros from the categories \mathopen, \mathclose, and \mathfence are emulated. Due to current MATHJAX limitations, not all stretch to the correct height.

Also emulated are macros from the categories \mathpunct, \mathover, \mathunder, \mathaccent, \mathbotaccent, and \mathop.

The individual unicode-math macros of categories \mathbin, \mathord, and \mathrel are not emulated for MATHJAX, as there are more than two thousand of them, but they may be added as needed. Place the following in the document preamble after loading unicode-math, including a definition for each macro which is used in the document but undefined in MATHJAX:

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\uplus}{\mathbin{\unicode{x0228E}}}}
...
\end{warpMathJax}
```

Use \mathrel, \mathbin, etc. depending on the category of each macro. For a list of macro names and symbols, see **texdoc unimath-symbols**.

```
1 \LWR@ProvidesPackagePass{unicode-math}[2019/09/26]

2 \LWR@origRequirePackage{lwarp-common-mathjax-letters}
3
4 \begin{warpMathJax}
5 \LWR@infoprocessingmathjax{unicode-math}
6
7 % Not all are possible in MathJax.
8 \CustomizeMathJax{\let\symnormal\mathit}
9 \CustomizeMathJax{\let\symliteral\mathrm}
10 \CustomizeMathJax{\let\symbb\mathbb}
11 \CustomizeMathJax{\let\symbbit\mathbb}% not italic
12 \CustomizeMathJax{\let\symcal\mathcal}
13 \CustomizeMathJax{\let\symscr\mathscr}
14 \CustomizeMathJax{\let\symfrak\mathfrak}
15
16 \CustomizeMathJax{\let\symsfup\mathsf}
17
18 \CustomizeMathJax{\let\symsfit\mathit}% not sans
19 % \CustomizeMathJax{\newcommand{\symsfit}[1]{%
20 %     \mmlToken{mi}[mathvariant="sans-serif-italic"]{#1}}% not greek
21 % }
22
23 \CustomizeMathJax{\let\symbfsf\mathbf}% not sans
24 % \CustomizeMathJax{\newcommand{\symbfsf}[1]{%
25 %     \mmlToken{mi}[mathvariant="bold-sans-serif"]{#1}}% not greek
26 % }
27
28 \CustomizeMathJax{\let\symbfup\mathbf}
29 \CustomizeMathJax{\newcommand{\symbfit}[1]{\boldsymbol{#1}}}
30 \CustomizeMathJax{\let\symbfcal\mathcal}% not bold
31
32 \CustomizeMathJax{\let\symbfscr\mathscr}% not bold
33 % \CustomizeMathJax{\newcommand{\symbfscr}[1]{
34 %     \mmlToken{mi}[mathvariant="math-bold-script"]{#1}}% not greek
35 % }
36
37 \CustomizeMathJax{\let\symbffrak\mathfrak}% not bold
38 % \CustomizeMathJax{\newcommand{\symbffrak}[1]{%
39 %     \mmlToken{mi}[mathvariant="math-bold-fraktur"]{#1}}% not greek
40 % }
41
42 \CustomizeMathJax{\let\symbfsfup\mathbf}% not sans
43 % \CustomizeMathJax{\newcommand{\symbfsfup}[1]{%
44 %     \mmlToken{mi}[mathvariant="bold-sans-serif"]{#1}}% not greek
45 % }
46
47 \CustomizeMathJax{\newcommand{\symbfsfit}[1]{\boldsymbol{#1}}}% not sans
48 % \CustomizeMathJax{\newcommand{\symbfsfit}[1]{%
49 %     \mmlToken{mi}[mathvariant="sans-serif-bold-italic"]{#1}}% not greek
50 % }
```

```
51
52 % Duplicates below are commented out.
53 \CustomizeMathJax{\let\symup\mathrm}
54 \CustomizeMathJax{\let\symbf\mathbf}% \symbfup defined above
55 \CustomizeMathJax{\let\symit\mathit}
56 % \CustomizeMathJax{\let\symbfit\mathit}% not bold
```

```
57 \ExplSyntaxOn
58 \AtBeginDocument{
59 \bool_if:NTF \g__um_sfliteral_bool
60     {\CustomizeMathJax{\let\symsf\symsfup}}
61     {
62         \bool_if:NTF \g__um_upsans_bool
63             {\CustomizeMathJax{\let\symsf\symsfup}}
64             {\CustomizeMathJax{\let\symsf\symsfit}}
65     }
66 }
67 \ExplSyntaxOff
```

```
68 % \CustomizeMathJax{\let\symbfsfup\mathbf}% not sans
69 % \CustomizeMathJax{\let\symsfit\mathit}% not sans
70 % \CustomizeMathJax{\let\symbfsfit\mathit}% not bold nor sans
71 \CustomizeMathJax{\let\symtt\mathtt}
72 % \CustomizeMathJax{\let\symbb\mathbb}
73 % \CustomizeMathJax{\let\symbbit\mathbb}% not italic
74 % \CustomizeMathJax{\let\symscr\mathscr}
75 % \CustomizeMathJax{\let\symbfscr\mathscr}% not bold
76 % \CustomizeMathJax{\let\symfrak\mathfrak}
77 \CustomizeMathJax{\let\symbffrac\mathbffrac}
```

Some symbol categories defined by unicode-math, in case they are used inside custom macros:

```
78 \CustomizeMathJax{\newcommand{\mathfence}[1]{\mathord{#1}}}
79 \CustomizeMathJax{\newcommand{\mathover}[1]{#1}}
80 \CustomizeMathJax{\newcommand{\mathunder}[1]{#1}}
81 \CustomizeMathJax{\newcommand{\mathaccent}[1]{#1}}
82 \CustomizeMathJax{\newcommand{\mathbotaccent}[1]{#1}}
83 \CustomizeMathJax{\newcommand{\mathalpha}[1]{\mathord{#1}}}
```

math-style is one of: ISO, TeX, french, upright, or literal, which set \g__um_upGreek_bool and \g__um_upgreek_bool.

```
84 \ExplSyntaxOn
85
86 \AtBeginDocument{
87 \bool_if:NTF \g__um_upGreek_bool
88     {\LWR@mathjax@addgreek@u@up*{}{}}
89     {\LWR@mathjax@addgreek@u@it*{}{}}
90
91 \bool_if:NTF \g__um_upgreek_bool
92     {\LWR@mathjax@addgreek@l@up{}{}}
93     {\LWR@mathjax@addgreek@l@it{}{}}
94 }
95
96 \LWR@mathjax@addgreek@u@up*{up}{}
97 \LWR@mathjax@addgreek@u@it*{it}{}
98 \LWR@mathjax@addgreek@l@up{up}{}
```

```
 99 \LWR@mathjax@addgreek@l@it{it}{}
100
101 \ExplSyntaxOff
102
103 \CustomizeMathJax{\let\lparen(}
104 \CustomizeMathJax{\let\rparen)}
105 \CustomizeMathJax{\newcommand{\cuberoot}[1]{\,{}^3\!\!\sqrt{#1}}\,}
106 \CustomizeMathJax{\newcommand{\fourthroot}[1]{\,{}^4\!\!\sqrt{#1}}\,}
```

Many \mathopen / \mathclose delimiters are defined in lwarp_mathjax.txt, where
\left / \right support is added.

```
107 \CustomizeMathJax{\newcommand{\longdivision}[1]{\mathord{\unicode{x027CC}#1}}}
108
109 \CustomizeMathJax{\newcommand{\mathcomma}{,}}
110 \CustomizeMathJax{\newcommand{\mathcolon}{:}}
111 \CustomizeMathJax{\newcommand{\mathsemicolon}{;}}
112
113 \CustomizeMathJax{\newcommand{\overbracket}[1]{\mathinner{\overline{\ulcorner{#1}\urcorner}}}}
114 \CustomizeMathJax{\newcommand{\underbracket}[1]{\mathinner{\underline{\llcorner{#1}\lrcorner}}}}
115
116 \CustomizeMathJax{\newcommand{\overbar}[1]{\mathord{#1\unicode{x00305}}}}
117 \CustomizeMathJax{\newcommand{\ovhook}[1]{\mathord{#1\unicode{x00309}}}}
118 \CustomizeMathJax{\newcommand{\ocirc}[1]{\mathord{#1\unicode{x0030A}}}}
119 \CustomizeMathJax{\newcommand{\candra}[1]{\mathord{#1\unicode{x00310}}}}
120 \CustomizeMathJax{\newcommand{\oturnedcomma}[1]{\mathord{#1\unicode{x00312}}}}
121 \CustomizeMathJax{\newcommand{\ocommatopright}[1]{\mathord{#1\unicode{x00315}}}}
122 \CustomizeMathJax{\newcommand{\droang}[1]{\mathord{#1\unicode{x0031A}}}}
123 \CustomizeMathJax{\newcommand{\leftharpoonaccent}[1]{\mathord{#1\unicode{x020D0}}}}
124 \CustomizeMathJax{\newcommand{\rightharpoonaccent}[1]{\mathord{#1\unicode{x020D1}}}}
125 \CustomizeMathJax{\newcommand{\vertoverlay}[1]{\mathord{#1\unicode{x020D2}}}}
126 \CustomizeMathJax{\newcommand{\leftarrowaccent}[1]{\mathord{#1\unicode{x020D0}}}}
127 \CustomizeMathJax{\newcommand{\annuity}[1]{\mathord{#1\unicode{x020E7}}}}
128 \CustomizeMathJax{\newcommand{\widebridgeabove}[1]{\mathord{#1\unicode{x020E9}}}}
129 \CustomizeMathJax{\newcommand{\asteraccent}[1]{\mathord{#1\unicode{x020F0}}}}
130 \CustomizeMathJax{\newcommand{\threeunderdot}[1]{\mathord{#1\unicode{x020E8}}}}
131
132 \CustomizeMathJax{\newcommand{\Bbbsum}{\mathop{\unicode{x2140}}\limits}}
133 \CustomizeMathJax{\newcommand{\oiint}{\mathop{\unicode{x222F}}\limits}}
134 \CustomizeMathJax{\newcommand{\oiiint}{\mathop{\unicode{x2230}}\limits}}
135 \CustomizeMathJax{\newcommand{\intclockwise}{\mathop{\unicode{x2231}}\limits}}
136 \CustomizeMathJax{\newcommand{\ointclockwise}{\mathop{\unicode{x2232}}\limits}}
137 \CustomizeMathJax{\newcommand{\ointctrclockwise}{\mathop{\unicode{x2233}}\limits}}
138 \CustomizeMathJax{\newcommand{\varointclockwise}{\mathop{\unicode{x2232}}\limits}}
139 \CustomizeMathJax{\newcommand{\leftouterjoin}{\mathop{\unicode{x27D5}}\limits}}
140 \CustomizeMathJax{\newcommand{\rightouterjoin}{\mathop{\unicode{x27D6}}\limits}}
141 \CustomizeMathJax{\newcommand{\fullouterjoin}{\mathop{\unicode{x27D7}}\limits}}
142 \CustomizeMathJax{\newcommand{\bigbot}{\mathop{\unicode{x27D8}}\limits}}
143 \CustomizeMathJax{\newcommand{\bigtop}{\mathop{\unicode{x27D9}}\limits}}
144 \CustomizeMathJax{\newcommand{\xsol}{\mathop{\unicode{x29F8}}\limits}}
145 \CustomizeMathJax{\newcommand{\xbsol}{\mathop{\unicode{x29F9}}\limits}}
146 \CustomizeMathJax{\newcommand{\bigcupdot}{\mathop{\unicode{x2A03}}\limits}}
147 \CustomizeMathJax{\newcommand{\bigsqcap}{\mathop{\unicode{x2A05}}\limits}}
148 \CustomizeMathJax{\newcommand{\conjquant}{\mathop{\unicode{x2A07}}\limits}}
149 \CustomizeMathJax{\newcommand{\disjquant}{\mathop{\unicode{x2A08}}\limits}}
150 \CustomizeMathJax{\newcommand{\bigtimes}{\mathop{\unicode{x2A09}}\limits}}
151 \CustomizeMathJax{\newcommand{\modtwosum}{\mathop{\unicode{x2A0A}}\limits}}
152 \CustomizeMathJax{\newcommand{\sumint}{\mathop{\unicode{x2A0B}}\limits}}
153 \CustomizeMathJax{\newcommand{\intbar}{\mathop{\unicode{x2A0D}}\limits}}
154 \CustomizeMathJax{\newcommand{\intBar}{\mathop{\unicode{x2A0E}}\limits}}
```

```
155 \CustomizeMathJax{\newcommand{\fint}{\mathop{\unicode{x2A0F}}\limits}}
156 \CustomizeMathJax{\newcommand{\cirfnint}{\mathop{\unicode{x2A10}}\limits}}
157 \CustomizeMathJax{\newcommand{\awint}{\mathop{\unicode{x2A11}}\limits}}
158 \CustomizeMathJax{\newcommand{\rppolint}{\mathop{\unicode{x2A12}}\limits}}
159 \CustomizeMathJax{\newcommand{\scpolint}{\mathop{\unicode{x2A13}}\limits}}
160 \CustomizeMathJax{\newcommand{\npolint}{\mathop{\unicode{x2A14}}\limits}}
161 \CustomizeMathJax{\newcommand{\pointint}{\mathop{\unicode{x2A15}}\limits}}
162 \CustomizeMathJax{\newcommand{\sqint}{\mathop{\unicode{x2A16}}\limits}}
163 \CustomizeMathJax{\newcommand{\intlarhk}{\mathop{\unicode{x2A17}}\limits}}
164 \CustomizeMathJax{\newcommand{\intx}{\mathop{\unicode{x2A18}}\limits}}
165 \CustomizeMathJax{\newcommand{\intcap}{\mathop{\unicode{x2A19}}\limits}}
166 \CustomizeMathJax{\newcommand{\intcup}{\mathop{\unicode{x2A1A}}\limits}}
167 \CustomizeMathJax{\newcommand{\upint}{\mathop{\unicode{x2A1B}}\limits}}
168 \CustomizeMathJax{\newcommand{\lowint}{\mathop{\unicode{x2A1C}}\limits}}
169 \CustomizeMathJax{\newcommand{\bigtriangleleft}{\mathop{\unicode{x2A1E}}\limits}}
170 \CustomizeMathJax{\newcommand{\zcmp}{\mathop{\unicode{x2A1F}}\limits}}
171 \CustomizeMathJax{\newcommand{\zpipe}{\mathop{\unicode{x2A20}}\limits}}
172 \CustomizeMathJax{\newcommand{\zproject}{\mathop{\unicode{x2A21}}\limits}}
173 \CustomizeMathJax{\newcommand{\biginterleave}{\mathop{\unicode{x2AFC}}\limits}}
174 \CustomizeMathJax{\newcommand{\bigtalloblong}{\mathop{\unicode{x2AFF}}\limits}}
175 \CustomizeMathJax{\newcommand{\arabicmaj}{\mathop{\unicode{x1EEF0}}\limits}}
176 \CustomizeMathJax{\newcommand{\arabichad}{\mathop{\unicode{x1EEF1}}\limits}}
177
178 \end{warpMathJax}
```

File 546 **`lwarp-units.sty`**

§ 658   Package **units**

(*Emulates or patches code by* AXEL REICHERT.)

units (*Pkg*)   units is patched for use by lwarp.

Values are not styled by css, and take the style of the surrounding HTML text.

Units are styled according to the print version, so they will be forced to upright roman in HTML if the print version does so. It may be necessary to adjust the document's body css to match the print version.

for HTML output:
```
1 \LWR@ProvidesPackagePass{units}[1998/08/04]
```

```
2 \DeclareRobustCommand*{\LWR@HTML@unit}[2][]{%
3 \ifblank{#1}%
4     {\LWR@textcurrentfont{#2}}%
5     {%
6         #1%
7         \ifthenelse{\boolean{B@UnitsLoose}}{~}{\,}%
8         \LWR@textcurrentfont{#2}%
9     }%
10 }
11 \LWR@formatted{unit}
```

```
12 \DeclareRobustCommand*{\LWR@HTML@unitfrac}[3][]{%
13 \ifblank{#1}%
14     {%
15             \nicefrac{#2}{#3}%
16     }%
```

```
17     {%
18             #1%
19             \ifthenelse{\boolean{B@UnitsLoose}}{~}{\,}%
20             \nicefrac{#2}{#3}%
21     }%
22 }
23
24 \LWR@formatted{unitfrac}
```

For MATHJAX:

```
25 \begin{warpMathJax}
26 \CustomizeMathJax{\newcommand{\unit}[2][]{#1 \mathinner{#2}}}
27 \CustomizeMathJax{\newcommand{\unitfrac}[3][]{#1 \mathinner{{}^{#2}\!/\!_{#3}}}}
28 \end{warpMathJax}
```

File 547 **lwarp-unitsdef.sty**

§ 659   Package   # unitsdef

(*Emulates or patches code by* PATRICK HAPPEL.)

unitsdef (*Pkg*)   unitsdef is patched for use by lwarp.

for HTML output: 1 \LWR@ProvidesPackagePass{unitsdef}[2005/01/04]

```
2 \newcommand{\LWR@HTML@unitvaluesep}{\,}
3 \LWR@formatted{unitvaluesep}
4
5 \newcommand{\LWR@HTML@unittimes}{\@@setunitsepfalse\HTMLunicode{22c5}}% \cdot
6 \LWR@formatted{unittimes}
7
8 \newunit{\LWR@HTML@arcmin}{%
9     \HTMLunicode{2032}% prime
10 }
11 \LWR@formatted{arcmin}
12
13 \newunit{\LWR@HTML@arcsec}{%
14     \HTMLunicode{2033}% dbl prime
15 }
16 \LWR@formatted{arcsec}
17
18 \newrobustcmd{\LWR@HTML@SI}[2]{%
19   \begingroup%
20     \let\unit@@xspace\relax%
21     \unitSIdef\selectfont%
22       \LWR@textcurrentfont{#1#2}% lwarp
23   \endgroup%
24 }
25 \LWR@formatted{SI}
```

File 548 **`lwarp-upgreek.sty`**

§ 660 Package **upgreek**

(*Emulates or patches code by* WALTER SCHMIDT.)

upgreek (*Pkg*) upgreek is used as-is for SVG math, and is emulated for MATHJAX.

for HTML output:

```
1 \LWR@ProvidesPackagePass{upgreek}[2003/02/12]
```

For MATHJAX:

```
2 \begin{warpMathJax}
3 \CustomizeMathJax{\require{upgreek}}
4 \end{warpMathJax}
```

File 549 **`lwarp-upref.sty`**

§ 661 Package **upref**

upref (*Pkg*) upref is ignored.

for HTML output: Discard all options for lwarp-upref:

```
1 \LWR@ProvidesPackageDrop{upref}[2007/03/14]
```

File 550 **`lwarp-url.sty`**

§ 662 Package **url**

(*Emulates or patches code by* DONALD ARSENEAU.)

url (*Pkg*) url is patched for use by lwarp.

for HTML output:

```
1 \LetLtxMacro\LWR@url@orig@url\LWR@url
2
3 \LWR@ProvidesPackagePass{url}[2013/09/16]

4 \newcommand*{\LWR@HTML@Url@FormatString}{%
5     \expandafter\LWR@url@orig@url\expandafter{\Url@String}%
6 }
7 \LWR@formatted{Url@FormatString}
```

File 551 **`lwarp-ushort.sty`**

§ 663 Package **ushort**

(*Emulates or patches code by* MARTIN VÄTH.)

ushort (*Pkg*) ushort is used as-is, and emulated for MATHJAX.

for HTML output:
```
1 \LWR@ProvidesPackagePass{ushort}[2001/06/13]
```

```
2 \begin{warpMathJax}
3 \CustomizeMathJax{\newcommand{\ushortdline}[1]{%
4     \kern{.1em}\underline{\underline{{#1}}}\kern{.1em}%
5 }}
6 \CustomizeMathJax{\newcommand{\ushort}[1]{\kern{.1em}\underline{#1}\kern{.1em}}}
7 \CustomizeMathJax{\newcommand{\ushortd}[1]{\ushortdline{#1}}}
8 \CustomizeMathJax{\newcommand{\ushortw}[1]{\kern{.1em}\underline{#1}\kern{.1em}}}
9 \CustomizeMathJax{\newcommand{\ushortdw}[1]{\ushortdline{#1}}}
10 \end{warpMathJax}
```

File 552 `lwarp-uspace.sty`

§ 664   Package **uspace**

uspace (*Pkg*) uspace is ignored.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{uspace}[2016/11/06]
```

File 553 `lwarp-varioref.sty`

§ 665   Package **varioref**

(*Emulates or patches code by* FRANK MITTELBACH.)

varioref (*Pkg*) varioref is patched for use by lwarp.

for HTML output:
```
1 \LWR@ProvidesPackagePass{varioref}[2020/01/23]
```

Page-related output is not used for HTML output.

```
2 \def\reftextfaceafter {\unskip}%
3 \def\reftextfacebefore{\unskip}%
4 \def\reftextafter     {\unskip}%
5 \def\reftextbefore    {\unskip}%
6 \def\reftextcurrent   {\unskip}%
7 \def\reftextfaraway#1{\unskip}%
8 \def\reftextpagerange#1#2{\unskip}%
```

File 554 `lwarp-verse.sty`

§ 666   Package **verse**

(*Emulates or patches code by* PETER WILSON.)

verse (*Pkg*) verse is supported and patched by lwarp.

for HTML output: Pass all options for lwarp-verse:

```
1 \LWR@ProvidesPackagePass{verse}[2009/09/04]
```

When using **verse** or **memoir**, always place a \\ after each line.

\attrib   The documentation for the **verse** and **memoir** packages suggest defining an \attrib command, which may already exist in current documents, but it will only work for print output. lwarp provides \attribution, which works for both print and HTML output. To combine the two so that \attrib is used for print and \attribution is used for HTML:

>   \begin{warpHTML}
>   \let\attrib\attribution
>   \end{warpHTML}

\vleftskip (*Len*)   These lengths are used by **verse** and **memoir** to control the left margin, and they
\vleftmargini (*Len*)   may already be set by the user for print output. New lengths \HTMLvleftskip and
\HTMLvleftskip (*Len*)   \HTMLleftmargini are provided to control the margins in HTML output. These
\HTMLleftmargini (*Len*) new lengths may be set by the user before any verse environment, and persist until they are manually changed again. One reason to change \HTMLleftmargini is if there is a wide \flagverse in use, such as the word "Chorus", in which case the value of \HTMLleftmargini should be set to a wide enough length to contain "Chorus". The default is wide enough for a stanza number.

⚠ spacing   Horizontal spacing relies on *pdftotext*'s ability to discern the layout (-layout option) of the text in the HTML-tagged PDF output. For some settings of \HTMLleftmargini or \HTMLleftskip the horizontal alignment may not work out exactly, in which

⚠ verse margin   case a label may be shifted by one space. During translation to HTML, the stanza numbers are kept out of the left margin, which would have caused *pdftotext* to shift everything over.

verse (*env.*)   The verse environment will be placed inside a HTML <pre>.

>   2 \AfterEndPreamble{
>   3 \LWR@traceinfo{Patching verse.}

At the beginning of the verse environment:

>   4 \AtBeginEnvironment{verse}
>   5 {%

Use the original list environment inside a <pre> to attempt to preserve formatting.

>   6 \LWR@restoreoriglists%

verse (*Pkg*)   The **verse** or **memoir** packages can place stanza numbers to the left with their
memoir (*Cls*)   \flagverse command. The following does not allow them to go into the left
\flagverse   margin, which would cause *pdfcrop* to crop the entire page further to the left.

\vleftskip (*Len*)
>   7 \ifdef{\vleftskip}{%
>   8 \setlength{\vleftskip}{\HTMLvleftskip}
>   9 \setlength{\leftmargini}{\HTMLleftmargini}
>   10 }{}
>   11 \LWR@forcenewpage
>   12 \LWR@atbeginverbatim{verse}%
>   13 }

After the end of the verse environment, which places the <pre> tag at the regular left margin:

```
14 \AtEndEnvironment{verse}{%
15 \leavevmode%
16 \LWR@afterendverbatim%
17 }
```

Patch to place poemtitle inside an HTML <span> of class poemtitle:

```
18 \ifdef{\poemtitle}{
19 \DeclareDocumentCommand{\@vstypeptitle}{m}{%
20    \vspace{\beforepoemtitleskip}%
21    {\InlineClass{poemtitle}{\poemtitlefont #1}\par}%
22    \vspace{\afterpoemtitleskip}%
23 }
24 }{}
25
26 \LWR@traceinfo{Finished patching verse.}
27 }% AfterEndPreamble
```

File 555   **`lwarp-versonotes.sty`**

§ 667   Package   # versonotes

(*Emulates or patches code by* NORMAN GRAY.)

versonotes (*Pkg*)   **versonotes** is emulated.

for HTML output: `1 \LWR@ProvidesPackageDrop{versonotes}[2019/07/06]`

```
2 \newcommand{\versonote}[1]{\marginpar{#1}}
3 \newdimen\versotextwidth
4 \newdimen\versoleftmargin
5 \newcommand*{\versolayout}{}
```

In case the user changed the page number before loading versonotes:

```
6 \setcounter{page}{1}
```

File 556   **`lwarp-vertbars.sty`**

§ 668   Package   # vertbars

(*Emulates or patches code by* PETER WILSON.)

vertbars (*Pkg*)   **vertbars** is emulated.

for HTML output: `1 \LWR@ProvidesPackageDrop{vertbars}[2010/11/27]`

```
2 \newlength{\barwidth}
3 \setlength{\barwidth}{0.4pt}
4 \newlength{\barspace}
```

```
 5 \setlength{\barspace}{1em}
 6
 7 \newenvironment{vertbar}{
 8     \LWR@forcenewpage
 9     \LWR@forceminwidth{\barwidth}
10    \begin{BlockClass}[%
11        border-left: \LWR@printlength{\LWR@atleastonept} solid black ; %
12        padding-left: \LWR@printlength{\barspace}%
13    ]{vertbar}
14 }{
15    \end{BlockClass}
16 }
```

File 557   **`lwarp-vmargin.sty`**

§ 669   Package   # vmargin

vmargin (*Pkg*)   vmargin is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{vmargin}[2004/07/15]

```
 2 \newcommand*{\LWRVM@customsize}[2]{}
 3 \newcommand*{\setpapersize}[2][]{\ifstrequal{#2}{custom}{\LWRVM@customsize}{}}
 4 \newcommand*{\setmargins}[8]{}
 5 \newcommand*{\setmarginsrb}[8]{}
 6 \newcommand*{\setmargnohf}[4]{}
 7 \newcommand*{\setmargnohfrb}[4]{}
 8 \newcommand*{\setmarg}[4]{}
 9 \newcommand*{\setmargrb}[4]{}
10 \newlength{\PaperWidth}
11 \setlength{\PaperWidth}{8.5in}
12 \newlength{\PaperHeight}
13 \setlength{\PaperHeight}{11in}
14 \newif\ifLandscape
```

File 558   **`lwarp-vowel.sty`**

§ 670   Package   # vowel

(*Emulates or patches code by* FUKUI REI.)

vowel (*Pkg*)   vowel is patched for use by lwarp.

This package has been tested with *pdflatex* and the Type 1 TIPA fonts using the following package load sequence:

```
\usepackage[T3,T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[noenc]{tipa}
\usepackage{vowel}
```

for HTML output: 1 \LWR@ProvidesPackagePass{vowel}[2002/08/08]

```
2 \renewenvironment{vowel}[1][]
3    {%
4        \begin{lateximage}[-vowel-~\PackageDiagramAltText]?%
5        \@vowel[#1]%
6    }
7    {%
8        \@@vowel%
9        \end{lateximage}%
10    }
```

File 559 **lwarp-vpe.sty**

§ 671   Package   **vpe**

vpe (*Pkg*)   vpe is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{vpe}[2012/04/18]

File 560   **lwarp-vwcol.sty**

§ 672   Package   **vwcol**

(*Emulates or patches code by* WILL ROBERTSON.)

vwcol (*Pkg*)   vwcol is patched for use with lwarp.

The width option is ignored. All vwcol environments adjust to 1–3 equal-width columns, depening on the width of the browser window.

The remaining options are supported, except for lines and maxrecursion.

for HTML output: 1 \LWR@ProvidesPackagePass{vwcol}[2015/02/10]

Factored from \vwcol. Each is given a style tag to append to the final style.

\LWR@vwcol@addrule   {⟨*style tag*⟩}

```
2 \newcommand*{\LWR@vwcol@addrule}[1]{%
3    \appto\LWR@vwcolstyle{%
4        #1: %
5        \LWR@printlength{\vwcol@rule} solid \LWR@origpound\LWR@vwcol@rulecolor ; %
6    }%
7 }
```

\LWR@vwcol@addrule   {⟨*style tag*⟩}

```
8 \newcommand*{\LWR@vwcol@addgap}[1]{%
9    \appto\LWR@vwcolstyle{%
10        #1: %
11        \LWR@printlength{\vwcol@sep} ; %
12    }%
13 }
```

Env vwcol   {⟨*key/values*⟩}

Redefine the environment to add a HTML style. The style is built depending on the required options.

```
14 \renewenvironment*{vwcol}[1][]{%
```

New paragraph, and process the options:

```
15 \LWR@stoppars%
16 \vwcolsetup{#1}%
```

Begin with no style:

```
17 \newcommand*{\LWR@vwcolstyle}{}
```

presep and postsep are created with HTML margins:

```
18 \if@vwcol@presep
19     \appto{\LWR@vwcolstyle}{margin-left: 1em ; padding-left: .5em ; }
20 \fi
21 \if@vwcol@postsep
22     \appto{\LWR@vwcolstyle}{margin-right: 1em ; padding-right: .5em ; }
23 \fi
```

sep becomes column-gap:

```
24 \ifdimgreater{\vwcol@sep}{1sp}{
25     \LWR@vwcol@addgap{column-gap}
26     \LWR@vwcol@addgap{-moz-column-gap}
27     \LWR@vwcol@addgap{-webkit-column-gap}
28 }{}
```

rule become column-rule, while prerule and postrule become HTML borders:

```
29 \convertcolorspec{named}{\vwcol@rulecol}{HTML}\LWR@vwcol@rulecolor%
30 \ifdimgreater{\vwcol@rule}{0pt}{
31     \ifdimless{\vwcol@rule}{1pt}{
32         \setlength{\vwcol@rule}{1pt}
33     }{}
34     \LWR@vwcol@addrule{column-rule}
35     \LWR@vwcol@addrule{-moz-column-rule}
36     \LWR@vwcol@addrule{-webkit-column-rule}
37     \if@vwcol@prerule\LWR@vwcol@addrule{border-left}\fi
38     \if@vwcol@postrule\LWR@vwcol@addrule{border-right}\fi
39 }{}
```

Each of the justify options becomes a text-align. Indentation is added where appropriate.

```
40 \ifdefequal{\vwcol@justify}{\RaggedRight}{
41     \appto{\LWR@vwcolstyle}{text-align: left ; }
42     \ifdimgreater{\vwcol@parindent}{0pt}{
43         \appto{\LWR@vwcolstyle}{%
44             text-indent: \LWR@printlength{\vwcol@parindent} ; %
45         }
46     }{}
47 }{}
```

```
48 \ifdefequal{\vwcol@justify}{\RaggedLeft}{
49     \appto{\LWR@vwcolstyle}{text-align: right ; }
50 }{}
```

```
51 \ifdefequal{\vwcol@justify}{\Centering}{
52     \appto{\LWR@vwcolstyle}{text-align: center ; }
53 }{}
```

```
54 \ifdefequal{\vwcol@justify}{\justifying}{
55     \appto{\LWR@vwcolstyle}{text-align: justify ; }
56     \ifdimgreater{\vwcol@parindent}{0pt}{
57         \appto{\LWR@vwcolstyle}{%
58             text-indent: \LWR@printlength{\vwcol@parindent} ; %
59         }
60     }{}
61 }{}
```

Create the <div> with the assembled style:

```
62 \BlockClass[\LWR@vwcolstyle]{multicols}
63 }
```

When the environment ends:

```
64 {
65     \endBlockClass
66     \LWR@startpars
67 }
```

File 561 **`lwarp-wallpaper.sty`**

§ 673   Package   # wallpaper

(*Emulates or patches code by* Michael H.F. Wilkinson.)

wallpaper (*Pkg*)   wallpaper is ignored.

1 \LWR@ProvidesPackageDrop{wallpaper}[2005/01/18]

```
2 \newcommand*{\CenterWallPaper}[2]{}
3 \newcommand*{\ThisCenterWallPaper}[2]{}
4 \newcommand*{\TileWallPaper}[3]{}
5 \newcommand*{\ThisTileWallPaper}[3]{}
6 \newcommand*{\TileSquareWallPaper}[2]{}
7 \newcommand*{\ThisTileSquareWallPaper}[2]{}
8 \newcommand*{\ULCornerWallPaper}[2]{}
9 \newcommand*{\ThisULCornerWallPaper}[2]{}
10 \newcommand*{\LLCornerWallPaper}[2]{}
11 \newcommand*{\ThisLLCornerWallPaper}[2]{}
12 \newcommand*{\URCornerWallPaper}[2]{}
13 \newcommand*{\ThisURCornerWallPaper}[2]{}
14 \newcommand*{\LRCornerWallPaper}[2]{}
15 \newcommand*{\ThisLRCornerWallPaper}[2]{}
16 \newcommand*{\ClearWallPaper}{}
17 \newlength{\wpXoffset}
18 \newlength{\wpYoffset}
```

File 562   **`lwarp-watermark.sty`**

§ 674   Package   # watermark

(*Emulates or patches code by* Alexander I. Rozhenko.)

watermark (*Pkg*)   watermark is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{watermark}[2004/12/09]

2 \newcommand{\watermark}[1]{}
3 \newcommand{\leftwatermark}[1]{}
4 \newcommand{\rightwatermark}[1]{}
5 \newcommand{\thiswatermark}[1]{}
6 \newcommand{\thispageheading}[1]{}

File 563   **`lwarp-widetable.sty`**

§ 675   Package   # widetable

(*Emulates or patches code by* CLAUDIO BECCARI.)

widetable (*Pkg*)   widetable is emulated.

for HTML output: 1 \LWR@ProvidesPackageDrop{widetable}[2019-06-25]

2 \newenvironment{widetable}{\begin{tabular*}}{\end{tabular*}}

File 564   **`lwarp-widows-and-orphans.sty`**

§ 676   Package   # widows-and-orphans

widows-and-orphans (*Pkg*)   widows-and-orphans is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{widows-and-orphans}[2018/09/01]

2 \NewDocumentCommand\WaOsetup{m}{}
3 \NewDocumentCommand\WaOparameters{}{}
4 \NewDocumentCommand\WaOignorenext{}{}

File 565   **`lwarp-witharrows.sty`**

§ 677   Package   # witharrows

(*Emulates or patches code by* F. PANTIGNY.)

witharrows (*Pkg*)   witharrows is patched for use by lwarp. Emulation is provided for MATHJAX.

for HTML output: 1 \LWR@ProvidesPackagePass{witharrows}[2024/10/19]

For MATHJAX, define a few things to be used in the hidden print version during
HTML output.

2 \ifbool{mathjax}{
3   \newcommand{\Arrow}[2][]{}
4   \newcommand{\unicode}[1]{}
5   \NewDocumentEnvironment { LWR@HTML@DispWithArrows } { ! d < > ! O { } +b}
6   {

```
7              \IfValueTF{#1}{
8                  \begin{displaymath}
9                      #1 \left\lbrace
10                     \begin{align}
11                         #3
12                     \end{align}
13                     \right .
14                 \end{displaymath}
15             }{
16                 \begin{displaymath}
17                     \begin{align}
18                         #3
19                     \end{align}
20                 \end{displaymath}
21             }
22         }
23         {}
24     \LWR@formattedenv{DispWithArrows}
25     \NewDocumentEnvironment { LWR@HTML@DispWithArrows* } { ! d < > ! O { } +b}
26         {
27             \IfValueTF{#1}{
28                 \begin{displaymath}
29                     #1 \left\lbrace
30                     \begin{align*}
31                         #3
32                     \end{align*}
33                     \right .
34                 \end{displaymath}
35             }{
36                 \begin{displaymath}
37                     \begin{align*}
38                         #3
39                     \end{align*}
40                 \end{displaymath}
41             }
42         }
43         {}
44     \LWR@formattedenv{DispWithArrows*}
45 }% MathJax
```

For svg output, use svg images.

```
46 {% SVG
47     \BeforeBeginEnvironment{WithArrows}{\global\booltrue{LWR@unknownmathsize}}
48     \BeforeBeginEnvironment{DispWithArrows}{%
49         \begin{BlockClass}{displaymathnumbered}%
50         \begin{lateximage}%
51     }
52     \AfterEndEnvironment{DispWithArrows}{\end{lateximage}\end{BlockClass}}
53     \BeforeBeginEnvironment{DispWithArrows*}{%
54         \begin{BlockClass}{displaymath}%
55         \begin{lateximage}%
56     }
57     \AfterEndEnvironment{DispWithArrows*}{\end{lateximage}\end{BlockClass}}
58 }% SVG
```

For Mathjax, emulate the commands which are defined only inside the envs.

```
59 \begin{warpMathJax}
60 \CustomizeMathJax{\newenvironment{WithArrows}[1][]{\begin{aligned}}{\end{aligned}}}
```

```
61 % Unable to make a sized box.
62 \CustomizeMathJax{\newcommand{\Arrow}[2][]{&{\Large\unicode{x2938}}~\textit{#2}}}
63 \end{warpMathJax}
```

File 566   **`lwarp-wrapfig.sty`**

§ 678   Package   # wrapfig

(*Emulates or patches code by* DONALD ARSENEAU.)

wrapfig (*Pkg*)   wrapfig is emulated.

for HTML output: 1 \LWR@ProvidesPackageDrop{wrapfig}[2003/01/31]

```
2 \newcommand*{\LWR@wrapposition}{}
3
4 \newcommand{\LWR@wrapfig@printHTMLwidth}{\LWR@printlength{\LWR@templengthone}}
5
6 \AtBeginDocument{
7     \IfPackageLoadedTF{keyfloat}{
8         \renewcommand{\LWR@wrapfig@printHTMLwidth}{%
9             \ifboolexpr{
10                 test {\ifnumgreater{\value{KFLT@keyfloatdepth}}{0}} or
11                 bool {KFLT@inkeysubfloats}
12             }%
13             {\LWR@printpercentlength{\LWR@templengthone}{\linewidth}\%; }%
14             {\LWR@printlength{\LWR@templengthone}}%
15         }%
16     }{}
17 }
18
19 \newcommand*{\LWR@subwrapfigure}[2]{%
20     \renewcommand*{\LWR@wrapposition}{}%
21     \ifthenelse{%
22         \equal{#1}{r}\OR\equal{#1}{R}\OR%
23         \equal{#1}{o}\OR\equal{#1}{O}%
24     }%
25         {\renewcommand*{\LWR@wrapposition}{float:right}}%
26         {\renewcommand*{\LWR@wrapposition}{float:left}}%
27     \setlength{\LWR@templengthone}{#2}%
28     \LWR@BlockClassWP{%
29         width:\LWR@printlength{\LWR@templengthone}; \LWR@wrapposition; %
30         margin:10pt%
31     }%
32     {%
33         width:\LWR@wrapfig@printHTMLwidth; %
34         \LWR@wrapposition; %
35     }%
36     (note)%
37     {marginblock}%

38     \setlength{\linewidth}{\LWR@templengthone}%
39 }
40
41
42 \NewDocumentEnvironment{wrapfigure}{o m o m}
43 {%
```

```
44      \begin{LWR@setvirtualpage}*%
45      \LWR@subwrapfigure{#2}{#4}%
46      \renewcommand*{\@captype}{figure}%
47 }
48 {%
49      \endLWR@BlockClassWP%
50      \end{LWR@setvirtualpage}%
51 }
52
53
54 \NewDocumentEnvironment{wraptable}{o m o m}
55 {%
56      \begin{LWR@setvirtualpage}*%
57      \LWR@subwrapfigure{#2}{#4}%
58      \renewcommand*{\@captype}{table}%
59 }
60 {%
61      \endLWR@BlockClassWP%
62      \end{LWR@setvirtualpage}%
63 }
64
65
66 \NewDocumentEnvironment{wrapfloat}{m o m o m}
67 {%
68      \begin{LWR@setvirtualpage}*%
69      \LWR@subwrapfigure{#3}{#5}%
70      \renewcommand*{\@captype}{#1}%
71 }
72 {%
73      \endLWR@BlockClassWP%
74      \end{LWR@setvirtualpage}%
75 }
76
77 \newlength{\wrapoverhang}
```

File 567   **`lwarp-wrapfig2.sty`**

§ 679   Package   # wrapfig2

(*Emulates or patches code by* Donald Arseneau, Claudio Beccari.)

wrapfig2 (*Pkg*)   wrapfig2 is emulated via a modified version of the wrapfig emulation.

for HTML output:
```
1 \@ifpackageloaded{color}{}{%
2    \@ifpackageloaded{xcolor}{}{\LWR@origRequirePackage{xcolor}}%
3 }
4
5 \RequirePackage{float}
6
7 \IfPackageLoadedWithOptionsTF{wrapfig2}{WFold}
8 {}% v4.0
9 {% v5+
10   \floatstyle{plain}
11   \ifcsname chapter\endcsname
12     \newfloat{text}{tbp}{lotx}[chapter]
13   \else
14     \newfloat{text}{tbp}{lotx}
15   \fi
```

```
16   \floatname{text}{Text}
17 %   \let\WF@text@caption\float@caption
18 }
19
20
21 \LWR@ProvidesPackageDrop{wrapfig2}[2022-02-16]
22
23 \LWR@origRequirePackage{lwarp-wrapfig}

24 \RenewDocumentEnvironment{wrapfigure}{o m o G{0pt} s}% original
25   {\wrapfloat{figure}[#1]{#2}[#3]{#4}}%
26   {\endwrapfloat}
27
28 \RenewDocumentEnvironment{wraptable}{o m o G{0pt} s}% original
29   {\wrapfloat{table}[#1]{#2}[#3]{#4}}%
30   {\endwrapfloat}
31
32 \RenewDocumentEnvironment{wrapfloat}{m o m o G{0pt}}% lwarp
33 {%
34     \begin{LWR@setvirtualpage}*%
35     \LWR@subwrapfigure{#3}{#5}%
36     \renewcommand*{\@captype}{#1}%
37 }
38 {%
39     \endLWR@BlockClassWP%
40     \end{LWR@setvirtualpage}%
41 }


42 \IfPackageLoadedWithOptionsTF{wrapfig2}{WFold}
43 {% v4.0:
44     \NewDocumentEnvironment{wraptext}%
45         {O{l}    D||{0.5\columnwidth} D<>{0}    D(){figure}}%
46     {%
47         \wrapfloat{#4}[]{#1}[]{#2}%
48         \tcolorbox%
49     }
50     {%
51         \endtcolorbox%
52         \endwrapfloat%
53         \ignorespaces%
54     }
55 }{}
56
57 \IfPackageLoadedWithOptionsTF{wrapfig2}{WFfive}
58 {% v5
59     \definecolor{WFbackground}{rgb}{0.95,0.95,0.95}
60     \definecolor{WFframe}{rgb}{0.1,0.1,0.1}
61     \colorlet{WFtext}{black}
62     \def\SetWFbgd#1{\colorlet{WFbackground}{#1}}
63     \def\SetWFfrm#1{\colorlet{WFframe}{#1}}
64     \def\SetWFtxt#1{\colorlet{WFtext}{#1}}
65     \def\WFsplitdimens#1,#2!{\fboxrule=#1\relax\fboxsep=#2\relax}
66
67     \NewDocumentEnvironment{wraptext}{O{0} m O{0pt} G{0.5\columnwidth}}
68     {%
69         \wrapfloat{text}[]{#2}[]{#4}%
70     }
71     {%
72         \endwrapfloat%
```

```
73          \ignorespaces%
74      }
75
76      \NewDocumentCommand\includeframedtext{O{\insertwidth} m O{1pt,1ex} o}%
77      {%
78          \WFsplitdimens #3!
79          \convertcolorspec{named}{WFtext}{HTML}\LWR@tempcolor%
80          \LWR@HTML@fcolorboxBlock%
81              [named]{WFframe}[named]{WFbackground}{#2}%
82              (%
83                  color:\ \LWR@origpound\LWR@tempcolor ; %
84                  border-radius:\ 1ex%
85              )%
86      }
87 }{% v6+
88      \RequirePackage{xkeyval}
89
90      \definecolor{WFbackground}{rgb}{0.95,0.95,0.95}
91      \definecolor{WFframe}{rgb}{0.1,0.1,0.1}
92      \colorlet{WFtext}{black}
93      \def\SetWFbgd#1{\colorlet{WFbackground}{#1}}
94      \def\SetWFfrm#1{\colorlet{WFframe}{#1}}
95      \def\SetWFtxt#1{\colorlet{WFtext}{#1}}
96      \def\WFsplitdimens#1,#2!{\fboxrule=#1\relax\fboxsep=#2\relax}
97
98      \newlength{\LWR@wrapfigtwo@radius}
99      \setlength{\LWR@wrapfigtwo@radius}{1ex}
100
101      \DeclareOptionX<wraptext>{scalefactor}[0.8]{%
102 %        \def\WFscalefactor{#1}%
103      }
104      \DeclareOptionX<wraptext>{fboxrule}[1pt]{\fboxrule=#1}
105      \DeclareOptionX<wraptext>{fboxsep}[1ex]{\fboxsep=#1}
106      \DeclareOptionX<wraptext>{framecolor}[WFframe]{\SetWFfrm{#1}}
107      \DeclareOptionX<wraptext>{backgroundcolor}[WFbackground]{\SetWFbgd{#1}}
108      \DeclareOptionX<wraptext>{textcolor}[WFtext]{\SetWFtxt{#1}}
109      \DeclareOptionX<wraptext>{fontstyle}[\normalfont]{#1}
110      \DeclareOptionX<wraptext>{radius}[\fboxsep]{%
111          \setlength{\LWR@wrapfigtwo@radius}{#1}%
112      }
113      \DeclareOptionX<wraptext>{insertionwidth}[0.5\columnwidth]{%
114 %        \insertwidth=#1%
115      }
116
117      \DeclareOptionX*{\PackageWarning{wrapfig2}{`\CurrentOption' ignored}}
118
119      \ExecuteOptionsX<wraptext>{scalefactor, fboxrule, fboxsep, framecolor,
120      backgroundcolor, textcolor, fontstyle, radius, insertionwidth}
121
122      \ProcessOptionsX*
123
124      \NewDocumentEnvironment{wraptext}{O{0} m O{0pt} G{0.5\columnwidth}}
125      {%
126          \wrapfloat{text}[]{#2}[]{#4}%
127      }
128      {%
129          \endwrapfloat%
130          \ignorespaces%
131      }
132
```

```
133    \NewDocumentCommand\includeframedtext{O{\insertwidth} m O{} o}
134    {%
135      \ExecuteOptionsX<wraptext>{#3}%    executes possible key=value options
136        \convertcolorspec{named}{WFtext}{HTML}\LWR@tempcolor%
137        \LWR@HTML@fcolorboxBlock%
138          [named]{WFframe}[named]{WFbackground}%
139          {\LWR@textcurrentfont{#2}}%
140          (%
141              color:\ \LWR@origpound\LWR@tempcolor ; %
142              border-radius:\ \LWR@printlength{\LWR@wrapfigtwo@radius}%
143          )%
144    }
145 }
```

File 568   **`lwarp-xbmks.sty`**

§ 680   Package   # xbmks

xbmks (*Pkg*)   xbmks is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{xbmks}[2018/07/04]

2 \newcommand{\xbmksetup}[1]{}
3 \NewDocumentCommand{\pdfbookmarkx}{o m o m}{}
4 \NewDocumentCommand{\currentpdfbookmarkx}{m o m}{}
5 \NewDocumentCommand{\subpdfbookmarkx}{m o m}{}
6 \NewDocumentCommand{\belowpdfbookmarkx}{m o m}{}

File 569   **`lwarp-xcolor.sty`**

§ 681   Package   # xcolor

(*Emulates or patches code by* Dr. Uwe Kern.)

xcolor (*Pkg*)   xcolor is supported by lwarp.

§ 681.1   Limitations

\colorboxBlock and   \colorboxBlock and \fcolorboxBlock are provided for increased HTML compati-
\fcolorboxBlock   bility, and they are identical to \colorbox and \fcolorbox in print mode. In HTML
mode they place their contents into a <div> instead of a <span>. These <div>s are
set to display: inline-block so adjacent \colorboxBlocks appear side-by-side
in HTML, although text is placed before or after each.

Print-mode definitions for \colorboxBlock and \fcolorboxBlock are created by
lwarp's core if xcolor is loaded.

background: none   \fcolorbox and \fcolorboxBlock allow a background color of none, in which case
only the frame is drawn, which can be useful for HTML.

color support   Color definitions, models, and mixing are fully supported without any changes
required.

colored text and boxes   \textcolor, \colorbox, and \fcolorbox are supported.

\color and \pagecolor   \color and \pagecolor are ignored. Use CSS or \textcolor where possible.

§ 681.2 **xcolor definitions: location and timing**

The lwarp core and its lwarp-xcolor package are tightly integrated to allow comparable results for print, HTML, and print inside an HTML lateximage. This requires a number of definitions and redefintions depending on whether each of xcolor and lateximage is being used, and whether print or HTML is being generated. Some of these actions are one-time when xcolor is loaded, and others are temporary as lateximage is used.

When xcolor is loaded in print mode: No special actions are taken at the time that xcolor is loaded in print mode, but see \AtBeginDocument below.

When lwarp-xcolor is loaded in HTML mode: xcolor's original definitions are saved for later restoration. \LWR@restoreorigformatting is appended to restore these definitions for use inside a lateximage. New HTML-mode definitions are created for \textcolor, \pagecolor, \nopagecolor, \colorbox, \colorboxBlock, \fcolorbox, \fcolorboxBlock, and fcolorminipage.

\AtBeginDocument in print or HTML mode: See Section 91. If xcolor has been loaded, the print-mode \fcolorbox is modified to accept a background color of none, and additional definitions are created for lwarp's new macros print-mode macros \colorboxBlock, \fcolorboxBlock, and fcolorminipage. The HTML versions of these macros will already have been created by lwarp-xcolor if it has been loaded.

For use inside an HTML lateximage, \LWR@restoreorigformatting is appended to temporarily set these functions to their print-mode versions.

In a lateximage in HTML mode: \LWR@restoreorigformatting temporarily restores the print-mode definitions of xcolor's functions. See \LWR@restoreorigformatting on page 544.

\color:

> **Print:** Used as-is.
>
> **HTML:** Ignored by *pdftotext*, and will not appear.
>
> **HTML lateximage:** Colors will appear in a lateximage.

\textcolor:

> **Print:** Used as-is.
>
> **HTML:** Redefined by lwarp-xcolor, page 1264.
>
> **HTML lateximage:** Remembers and reuses the print version.

\pagecolor:

> **Print:** Used as-is.
>
> **HTML:** Ignored.
>
> **HTML lateximage:** Colors will be picked up in a lateximage.

\nopagecolor:

> **Print:** Used as-is.
>
> **HTML:** Ignored.

HTML lateximage: Colors will be picked up in a lateximage.

\colorbox:

> **Print:** Used as-is.
>
> **HTML:** Redefined by lwarp-xcolor, page 1265.
>
> **HTML lateximage:** Remembers and reuses the print version.

\colorboxBlock:

> **Print:** Becomes \colorbox.
>
> **HTML:** Newly defined by lwarp-xcolor to use a <div>, page 1265.
>
> **HTML lateximage:** Remembers and reuses the print version \colorbox.

\fcolorbox:

> **Print:** Modified to allow a background of none.
>   \LWR@print@fcolorbox at section 91
>
> **HTML:** Redefined by lwarp-xcolor, page 1266.
>
> **HTML lateximage:** Remembers and reuses the print version.

\fcolorboxBlock:

> **Print:** Becomes \fcolorbox. Section 91
>
> **HTML:** Newly defined by lwarp-xcolor to use a <div>, page 1266.
>
> **HTML lateximage:** Remembers and reuses the print version \fcolorbox.

fcolorminipage:

> **Print:** Newly defined in the lwarp core.
>   LWR@print@fcolorminipage at section 91
>
> **HTML:** Newly defined by lwarp-xcolor, page 1267.
>
> **HTML lateximage:** Uses the print version.

\boxframe:

> **Print:** Used as-is.
>
> **HTML:** Redefined by lwarp-xcolor, page 1268.
>
> **HTML lateximage:** Remembers and reuses the print version.

§ 681.3 **Package loading**

for HTML output: 1 \LWR@ProvidesPackagePass{xcolor}[2023-11-15]

\color@endgroup's \endgraf was conflicting with lwarp's paragraph handling.

2 \let\color@endgroup\endgroup

§681.4 **Remembering and restoring original definitions**

Remember the following print-mode actions to be restored when inside a lateximage environment:

```
3 \LetLtxMacro\LWR@print@pagecolor\pagecolor
4 \LetLtxMacro\LWR@print@nopagecolor\nopagecolor
```

\LWR@restoreorigformatting Inside a lateximage the following gets restored to their print-mode actions:

```
5 \appto\LWR@restoreorigformatting{%
6     \LetLtxMacro\pagecolor\LWR@print@pagecolor%
7     \LetLtxMacro\nopagecolor\LWR@print@nopagecolor%
8 }
```

§681.5 **\normalcolor**

\normalcolor

```
9 \DeclareRobustCommand{\LWR@HTML@normalcolor}{\color{black}}%
10
11 \LWR@formatted{normalcolor}
```

§681.6 **HTML color style**

\LWR@findcurrenttextcolor   Sets \LWR@tempcolor to the current color.

```
12 \renewcommand*{\LWR@findcurrenttextcolor}{%
13     \LWR@traceinfo{LWR@findcurrenttextcolor}%
14     \protect\colorlet{LWR@current@color}{.}%
15     \LWR@traceinfo{LWR@findcurrenttextcolor B}%
16   \protect\convertcolorspec{named}{LWR@current@color}{HTML}\LWR@tempcolor\relax%
17     \LWR@traceinfo{LWR@findcurrenttextcolor: done}%
18 }
```

\LWR@textcurrentcolor {⟨*text*⟩} Like \textcolor but uses the current \color instead.

```
19 \DeclareDocumentCommand{\LWR@textcurrentcolor}{m}{%
20     \begingroup%
21     \LWR@hook@processingtags%
22     \LWR@findcurrenttextcolor%
23     \InlineClass[color:\LWR@origpound\LWR@tempcolor]{textcolor}{%
24         \renewcommand*{\LWR@currenttextcolor}{\LWR@origpound\LWR@tempcolor}%
25         #1%
26     }%
27     \endgroup%
28 }
```

\LWR@colorstyle   {⟨*1: model*⟩} {⟨*2: color*⟩}

For a color style, prints the color converted to HTML colors.

```
29 \NewDocumentCommand{\LWR@colorstyle}{m m}{%
30     \begingroup%
31     \LWR@hook@processingtags%
```

Use the xcolor package to convert to an HTML color space:

```
32     \convertcolorspec{#1}{#2}{HTML}\LWR@tempcolor%
```

Print the converted color:

```
33      \LWR@origpound\LWR@tempcolor%
34      \endgroup%
35 }
```

\LWR@backgroundcolor [⟨*model*⟩] {⟨*color*⟩} {⟨*text*⟩}

Similar to \textcolor, but prints black text against a color background.

Converted into an HTML hex color span.

```
36 \NewDocumentCommand{\LWR@backgroundcolor}{O{named} m m}{%
37      \begingroup%
38      \LWR@hook@processingtags%
39      \InlineClass[background:\LWR@colorstyle{#1}{#2}]{backgroundcolor}{%
40          #3%
41      }%
42      \endgroup%
43 }
```

§ 681.7 HTML border

\LWR@borderpadding   {⟨*colorstyle*⟩} {⟨*color*⟩} Prints the HTML attributes for a color border and padding.

\LWR@forceminwidth must be used first in order to set the border width.

```
44 \newcommand*{\LWR@borderpadding}[2]{%
45    border:\LWR@printlength{\LWR@atleastonept} solid \LWR@colorstyle{#1}{#2} ; %
46      padding:\LWR@printlength{\fboxsep}%
47 }
```

§ 681.8 High-level macros

\color [⟨*model*⟩] {⟨*color*⟩}

⚠️   The current \color is used by HTML rules and frames, but does not affect the
current HTML text output, due to the lack of HTML states and scoping limitations.
Use \textcolor if possible.

```
48 \NewDocumentCommand{\LWR@HTML@color}{o m}{%
49      \IfValueTF{#1}{%
50          \LWR@print@color[#1]{#2}%
51          \convertcolorspec{#1}{#2}{HTML}\LWR@tempcolor%
52      }{%
53          \LWR@print@color{#2}%
54          \convertcolorspec{named}{#2}{HTML}\LWR@tempcolor%
55      }%
56      \edef\LWR@currenttextcolor{\LWR@origpound\LWR@tempcolor}%
57 }
58
59 \LWR@formatted{color}
```

\textcolor [⟨*model*⟩] {⟨*color*⟩} {⟨*text*⟩}

Converted into an HTML hex color span.

```
60 \NewDocumentCommand{\LWR@HTML@textcolor}{o m m}{%
```

```
61     \begingroup%
62     \LWR@hook@processingtags%
63     \IfValueTF{#1}{%
64         \color[#1]{#2}%
65     }{%
66         \color{#2}%
67     }%
68     \InlineClass[color:\LWR@currenttextcolor]{textcolor}{#3}%
69     \endgroup%
70 }%
71
72 \LWR@formatted{textcolor}
```

\pagecolor [⟨*model*⟩] {⟨*color*⟩}

Ignored. Use css instead.

```
73 \renewcommand*{\pagecolor}[2][named]{}
```

\nopagecolor Ignored.

```
74 \renewcommand*{\nopagecolor}{}
```

\colorbox [⟨*model*⟩] {⟨*color*⟩} {⟨*text*⟩}

Converted into an HTML hex background color <span>.

```
75 \NewDocumentCommand{\LWR@HTML@colorbox}{O{named} m +m}{%
76     \begingroup%
77     \LWR@hook@processingtags%
78     \InlineClass[%
79     background:\LWR@colorstyle{#1}{#2} ; %
80     padding:\LWR@printlength{\fboxsep}%
81     ]{colorbox}{#3}%
82     \endgroup%
83 }
```

\colorboxBlock [⟨*model*⟩] {⟨*color*⟩} {⟨*text*⟩}

Converted into an HTML hex background color <div>.

```
84 \NewDocumentCommand{\LWR@HTML@colorboxBlock}{O{named} m +m}{%
85     \begingroup%
86     \LWR@hook@processingtags%

87     \LWR@stoppars%

88     \begin{BlockClass}[%
89         background:\LWR@colorstyle{#1}{#2} ; %
90         padding:\LWR@printlength{\fboxsep}%
91     ]{colorboxBlock}
92     #3
93     \end{BlockClass}%
94     \endgroup%
```

Prevent paragraph tags around horizontal white space until the start of the next paragraph:

```
95      \global\booltrue{LWR@minipagethispar}%
96 }
```

\fcolorbox [⟨*framemodel*⟩] {⟨*framecolor*⟩} [⟨*boxmodel*⟩] {⟨*boxcolor*⟩} {⟨*text*⟩}

Converted into a framed HTML hex background color span.

A background color of none creates a colored frame without a background color.

```
97  \NewDocumentCommand{\LWR@HTML@fcolorbox}{O{named} m O{#1} m +m}{%
98      \LWR@traceinfo{HTML fcolorbox #2 #4}%
99      \begingroup%
100     \LWR@hook@processingtags%
101     \LWR@forceminwidth{\fboxrule}%
102     \ifthenelse{\equal{#4}{none}}%
103         {% no background color
104             \InlineClass[%
105             \LWR@borderpadding{#1}{#2}%
106             ]{fcolorbox}{#5}%
107         }%
108         {% yes background color
109             \InlineClass[%
110             \LWR@borderpadding{#1}{#2} ; %
111             background:\LWR@colorstyle{#3}{#4}%
112             ]{fcolorbox}{#5}%
113         }%
114     \endgroup%
115 }
```

\fcolorboxBlock [⟨*framemodel*⟩] {⟨*framecolor*⟩} [⟨*boxmodel*⟩] {⟨*boxcolor*⟩} {⟨*text*⟩} (⟨*add'l html style*⟩)

Converted into a framed HTML hex background color span.

A background color of none creates a colored frame without a background color.

```
116 \NewDocumentCommand{\LWR@HTML@fcolorboxBlock}{O{named} m O{#1} m +m d()}{%
117     \LWR@traceinfo{HTML fcolorboxBlock #2 #4}%
118     \begingroup%
119     \LWR@hook@processingtags%
120     \LWR@forceminwidth{\fboxrule}%


121     \LWR@stoppars%


122     \ifthenelse{\equal{#4}{none}}%
123         {% no background color
124             \begin{BlockClass}[%
125                 \LWR@borderpadding{#1}{#2}%
126                 \IfValueT{#6}{ ; #6}%
127             ]{fcolorboxBlock}
128             #5
129             \end{BlockClass}%
130         }%
```

```
131              {% yes background color
132                  \convertcolorspec{#3}{#4}{HTML}\LWR@tempcolortwo%
133                  \begin{BlockClass}[%
134                      background:\LWR@origpound\LWR@tempcolortwo\ ; %
135                      \LWR@borderpadding{#1}{#2}%
136                      \IfValueT{#6}{ ; #6}%
137                  ]{fcolorboxBlock}
138                  #5
139                  \end{BlockClass}%
140          }%
141      \endgroup%
```

Prevent paragraph tags around horizontal white space until the start of the next
paragraph:

```
142      \global\booltrue{LWR@minipagethispar}%
143      \LWR@traceinfo{HTML fcolorboxBlock done}%
144 }
```

Creates a framed HTML <div> around its contents.

A print-output version is defined in the lwarp core: section 91

\LWR@subfcolorminipage   {⟨*framemodel*⟩} {⟨*framecolor*⟩} {⟨*background tag*⟩} {⟨*height*⟩}

```
145 \NewDocumentCommand{\LWR@subfcolorminipage}{m m m m}{%
```

```
146      \LWR@stoppars%
```

```
147      \begin{BlockClass}[%
148          #3%
149          \LWR@borderpadding{#1}{#2} ; %
150          \IfValueT{#4}{height:\LWR@printlength{\LWR@tempheight} ; }%
151          width:\LWR@printlength{\LWR@tempwidth}%
152      ]{fcolorminipage}%
153 }
```

fcolorminipage (*env.*) [⟨*1:framemodel*⟩] {⟨*2:framecolor*⟩} [⟨*3:boxmodel*⟩] {⟨*4:boxcolor*⟩} [⟨*5:align*⟩]
                          [⟨*6:height*⟩] [⟨*7:inner-align*⟩] {⟨*8:width*⟩}

```
154 \NewDocumentEnvironment{LWR@HTML@fcolorminipage}{O{named} m O{#1} m O{c} o o m}
155 {%
156      \LWR@hook@processingtags%
157      \setlength{\LWR@tempwidth}{#8}%
158      \IfValueT{#6}{\setlength{\LWR@tempheight}{#6}}%
159      \LWR@forceminwidth{\fboxrule}%
160      \convertcolorspec{#1}{#2}{HTML}\LWR@tempcolor%
161      \ifthenelse{\equal{#4}{none}}%
162          {\LWR@subfcolorminipage{#1}{#2}{}{#6}}%
163          {%
164              \convertcolorspec{#3}{#4}{HTML}\LWR@tempcolortwo%
165              \LWR@subfcolorminipage{#1}{#2}%
166                  {background:\LWR@origpound\LWR@tempcolortwo\ ; }%
167                  {#6}%
168          }%
169 }%
170 {%
171      \end{BlockClass}%
```

Prevent paragraph tags around horizontal white space until the start of the next paragraph:

```
172        \global\booltrue{LWR@minipagethispar}%
173 }
```

\boxframe {⟨*width*⟩} {⟨*height*⟩} {⟨*depth*⟩}

The depth is added to the height, but the box is not decended below by the depth. \textcolor is honored.

```
174 \newcommand*{\LWR@HTML@boxframe}[3]{%
175     {%
176         \setlength{\LWR@tempwidth}{#1}%
177         \setlength{\LWR@tempheight}{#2}%
178         \addtolength{\LWR@tempheight}{#3}%
179         \LWR@forceminwidth{\fboxrule}%
180         \LWR@findcurrenttextcolor%
181         \InlineClass[%
182             display:inline-block ; %
183             border:%
184                 \LWR@printlength{\LWR@atleastonept} % space
185                 solid % space
186                 \LWR@currenttextcolor{} ; % space
187             width:\LWR@printlength{\LWR@tempwidth} ; %
188             height:\LWR@printlength{\LWR@tempheight}%
189         ]{boxframe}{}%
190     }%
191 }
192
193 \LWR@formatted{boxframe}
```

File 570   **lwarp-xechangebar.sty**

§ 682   Package   **xechangebar**

xechangebar (*Pkg*)   xechangebar is ignored

for HTML output: 1 \LWR@ProvidesPackageDrop{xechangebar}[2017/08/03]
2 \LWR@origRequirePackage{lwarp-changebar}

File 571   **lwarp-xellipsis.sty**

§ 683   Package   **xellipsis**

(*Emulates or patches code by* Donald P. Goodman III.)

xellipsis (*Pkg*)   xellipsis is patched for use by lwarp.

When non-zero, each of the spaces is converted to an HTML thin unbreakable space.

for HTML output: 1 \LWR@ProvidesPackagePass{xellipsis}[2015/11/01]

```
2 \newcommand*{\LWR@xellipsespace}[1]{%
3 \ifdim#1=0pt\else%
4    \ifdim#1<\fontdimen2\font%
5       \,%
6    \else%
7       ~%
8    \fi%
9 \fi%
10 }
11
12 \def\xelip{%
13 \mbox{%
14    \LWR@xellipsespace{\xelipprebef}%
15    \xelipprechar%
16    \LWR@xellipsespace{\xelippreaft}%
17    \LWR@xellipsespace{\xelipbef}%
18    \xelipchar%
19    \xel@loopi = 1%
20    \loop\ifnum\xelipnum>\xel@loopi%
21       \advance\xel@loopi by1%
22       \LWR@xellipsespace{\xelipgap}%
23       \xelipchar%
24    \repeat%
25    \LWR@xellipsespace{\xelipaft}%
26    \LWR@xellipsespace{\xelippostbef}%
27    \xelippostchar%
28    \LWR@xellipsespace{\xelippostaft}%
29 }%
30 }%
```

File 572 **`lwarp-xetexko.sty`**

§ 684   Package   # xetexko

(*Emulates or patches code by* Dohyun Kim.)

xetexko (*Pkg*)   xetexko is patched for use by lwarp.

for HTML output:

```
1 \LWR@loadbefore{xetexko}
2
3 \LWR@ProvidesPackagePass{xetexko}[2021/09/06]
4
5 \protected\def\typesetvertical{}
6 \protected\def\typesethorizontal{}
7
8 \def\verticaltypesetting{\BlockClass{verticalrl}}
9 \def\beginverticaltypesetting{\BlockClass{verticalrl}}
10 \def\endverticaltypesetting{\endBlockClass}
11
12 \protected\def\vertical#1{\BlockClass{verticalrl}}
13 \protected\def\endvertical{\endBlockClass}
14 \protected\def\horizontal#1{\BlockClass{horizontaltb}}
15 \protected\def\endhorizontal{\endBlockClass}
16 \DeclareDocumentCommand{\vertlatin}{m}{#1}
```

File 573 **`lwarp-xevlna.sty`**

§ 685 Package **xevlna**

(*Emulates or patches code by* ZDENĚK WAGNER.)

xevlna (*Pkg*) xevlna is patched for use by lwarp.

Non-breakable spaces are inserted into HTML.

for HTML output: 1 `\LWR@ProvidesPackagePass{xevlna}[2016/09/05]`

2 `\def\ProcessCSpreposition{\ifx\next\xevlnaXeTeXspace\HTMLentity{nbsp}\fi}`
3
4 `\appto{\LWR@hook@processingtags}{\xevlnaDisable}%`

File 574 **`lwarp-xfakebold.sty`**

§ 686 Package **xfakebold**

(*Emulates or patches code by* HERBERT VOSS.)

xfakebold (*Pkg*) xfakebold is patched for use by lwarp, and additional underlying support is found in the lwarp core.

⚠ text mode xfakebold is only used in SVG math and lateximages. Text mode is not set bold, but `\setBold` in text will be applied to any following SVG math.

for HTML output: 1 `\LWR@ProvidesPackagePass{xfakebold}[2020/06/24]`

2 `\newcommand*{\LWR@HTML@setBold}{\booltrue{LWR@xfakebold}}`
3 `\LWR@formatted{setBold}`
4
5 `\newcommand*{\LWR@HTML@unsetBold}{\boolfalse{LWR@xfakebold}}`
6 `\LWR@formatted{unsetBold}`
7
8 `\renewcommand*{\LWR@applyxfakebold}{%`
9 `    \ifbool{LWR@xfakebold}{\LWR@print@setBold}{\LWR@print@unsetBold}%`
10 `}`

For MATHJAX, xfakebold is ignored.

11 `\begin{warpMathJax}`
12 `\CustomizeMathJax{\newcommand{\setBold}[1][]{}}`
13 `\CustomizeMathJax{\newcommand{\unsetBold}{}}`
14 `\end{warpMathJax}`

File 575 **`lwarp-xfrac.sty`**

§ 687   Package   **xfrac**

(*Emulates or patches code by* THE LATEX3 PROJECT.)

xfrac (*Pkg*)   Supported by adding xfrac instances, and emulated for MATHJAX.

for HTML output: 1 \LWR@ProvidesPackagePass{xfrac}[2018-08-23]

⚠   font size   In the user's document preamble, lwarp should be loaded after font-related setup. During HTML conversion, this font is used by lwarp to generate its initial PDF output containing HTML tags, later to be converted by *pdftotext* to a plain text file. While the text may be in any font which *pdftotext* can read, the math is directly converted into SVG images using this same user-selected font. xfrac below is set for the Latin Modern (lmr) font. If another font is used, it may be desirable to redefine \xfracHTMLfontsize with a different em size.

\sfrac   [⟨*instance*⟩] {⟨*num*⟩} [⟨*sep*⟩] {⟨*denom*⟩}

A text-mode instance for the default font is provided below. The numerator and denominator formats are adjusted to encase everything in HTML tags. \scalebox is made null inside the numerator and denominator, since the HTML tags should not be scaled, and we do not want to introduce additional HTML tags for scaling.

In math mode, which will appear inside a lateximage, no adjustments are necessary.

\xfracHTMLfontsize   User-redefinable macro which controls the font size of the fraction.

2 \newcommand*{\xfracHTMLfontsize}{.6em}

instances   Instances of xfrac for various font choices:

Produce CSS for a small raised numerator and a small denominator.

Scaling is turned off so that *pdftotext* correctly reads the result.

```
3 \DeclareInstance{xfrac}{default}{text}{
4     numerator-format = {%
5         \begingroup%
6         \RenewDocumentCommand{\scalebox}{m o m}{##3}%
7         \InlineClass{numerator}{#1}\,%
8         \endgroup%
9     },
10     denominator-format = {%
11        \begingroup%
12        \RenewDocumentCommand{\scalebox}{m o m}{##3}%
13        \InlineClass{denominator}{#1}%
14        \endgroup%
15     },
```

For *pdftotext*, do not scale the text:

```
16    scaling = false
17 }
18
19 \DeclareInstance{xfrac}{lmr}{text}{
20    numerator-format = {%
21        \begingroup%
22        \RenewDocumentCommand{\scalebox}{m o m}{##3}%
23        \InlineClass{numerator}{#1}\,%
24        \endgroup%
25    },
26    denominator-format = {%
27        \begingroup%
28        \RenewDocumentCommand{\scalebox}{m o m}{##3}%
29        \InlineClass{denominator}{#1}%
30        \endgroup%
31    },
```

For *pdftotext*, do not scale the text:

```
32    scaling = false
33 }
34
35 \DeclareInstance{xfrac}{lmss}{text}{
36    numerator-format = {%
37        \begingroup%
38        \RenewDocumentCommand{\scalebox}{m o m}{##3}%
39        \InlineClass{numerator}{#1}\,%
40        \endgroup%
41    },
42    denominator-format = {%
43        \begingroup%
44        \RenewDocumentCommand{\scalebox}{m o m}{##3}%
45        \InlineClass{denominator}{#1}%
46        \endgroup%
47    },
```

For *pdftotext*, do not scale the text:

```
48    scaling = false
49 }
50
51 \DeclareInstance{xfrac}{lmtt}{text}{
52    numerator-format = {%
53        \begingroup%
54        \RenewDocumentCommand{\scalebox}{m o m}{##3}%
55        \InlineClass{numerator}{#1}\,%
56        \endgroup%
57    },
58    denominator-format = {%
59        \begingroup%
60        \RenewDocumentCommand{\scalebox}{m o m}{##3}%
61        \InlineClass{denominator}{#1}%
62        \endgroup%
63    },
```

For *pdftotext*, do not scale the text:

```
64    scaling = false
65 }
```

For MATHJAX:

```
66 \begin{warpMathJax}
67 \CustomizeMathJax{\newcommand{\LWRsfrac}[2][/]{{}^\LWRsfracnumerator\!#1{}_{#2}}}
68 \CustomizeMathJax{\newcommand{\sfrac}[2][]{\def\LWRsfracnumerator{#2}\LWRsfrac}}
69 \end{warpMathJax}
```

File 576 **`lwarp-xltabular.sty`**

§ 688   Package   # xltabular

(*Emulates or patches code by* ROLF NIEPRASCHK, HERBERT VOSS.)

xltabular (*Pkg*)   xltabular is emulated by lwarp.

for HTML output: Relies on tabularx.

⚠ table numbering   At present, an `xltabular` without a caption or with only a `\caption*` may be misnumbered in HTML, so it may be necessary to place at the end of the table:

> `\warpHTMLonly{\addtocounter{table}{-1}}`

```
1 \RequirePackage{tabularx}
2 \RequirePackage{ltablex}
3
4 \LWR@ProvidesPackageDrop{xltabular}[2018/05/23]
5
6 \DeclareDocumentEnvironment{xltabular}{o m m}
7 {\longtable{#3}}
8 {\endlongtable}
```

File 577   **`lwarp-xltxtra.sty`**

§ 689   Package   # xltxtra

(*Emulates or patches code by* WILL ROBERTSON, JONATHAN KEW.)

xltxtra (*Pkg*)   xltxtra is emulated.

for HTML output: `1 \LWR@ProvidesPackageDrop{xltxtra}[2016/01/21]`

```
2 \RequirePackage{realscripts}
3 \RequirePackage{metalogo}
4 \newcommand*\TeX@logo@spacing[6]{}
5
6 \newcommand*{\vfrac}[2]{%
7 \textsuperscript{#1}/\textsubscript{#2}%
8 }
9
10 \newcommand\namedglyph[1]{%
11   \@tempcnta=\XeTeXglyphindex "#1"\relax
12   \ifnum\@tempcnta>0
13     \XeTeXglyph\@tempcnta
14   \else
15     \xxt@namedglyph@fallback{#1}%
```

```
16   \fi}
17
18 \newcommand\xxt@namedglyph@fallback[1]{[#1]}
19
20 \DeclareDocumentCommand{\showhyphens}{m}{}
```

File 578 **`lwarp-xmpincl.sty`**

§ 690 Package **xmpincl**

(*Emulates or patches code by* MAARTEN SNEEP.)

xmpincl (*Pkg*)　　xmpincl is ignored.

for HTML output:　Discard all options for lwarp-xmpincl:

```
1 \LWR@ProvidesPackageDrop{xmpincl}[2008/05/10]
```

```
2 \newcommand*{\includexmp}[1]{}
```

File 579 **`lwarp-xpiano.sty`**

§ 691 Package **xpiano**

(*Emulates or patches code by* ENRICO GREGORIO.)

xpiano (*Pkg*)　　xpiano is patched for use by lwarp.

for HTML output:　
```
1 \LWR@ProvidesPackagePass{xpiano}
```

```
2 \ExplSyntaxOn
3 \NewDocumentCommand{\LWR@print@keyboard}{ O{}m }
4 {
5 \xpiano_keyboard:nn { #1 } { #2 }
6 }
7
8 \NewDocumentCommand{\LWR@HTML@keyboard}{ O{}m }
9 {
10 \begin{lateximage}*
11     [%
12         -xpiano-~\PackageDiagramAltText{}: \detokenize\expandafter{#2}%
13     ]?%
14     [\detokenize\expandafter{#1}]
15 \xpiano_keyboard:nn { #1 } { #2 }
16 \end{lateximage}
17 }
18 \ExplSyntaxOff
19
20 \LWR@formatted{keyboard}
```

File 580 **`lwarp-xpinyin.sty`**

§ 692   Package   **xpinyin**

(*Emulates or patches code by* SOBEN LEE.)

xpinyin (*Pkg*)   xpinyin is supported.

Pinyin is disabled for file names, the sideтoc, and regular footnotes, but is left enabled for minipage footnotes, as per the print mode.

for HTML output: 1 `\LWR@ProvidesPackagePass{xpinyin}[2019-04-07]`

The original's boxes are not used, instead the contents are used with <ruby>, <rt>, and <rp> tags per modern HTML. Color is detected. ratio is ignored for *pdftotext* to work correctly. Extra spaces are placed inside the tags to allow line breaks in the HTML text.

```
2 \ExplSyntaxOn
3 \cs_new_protected_nopar:Npn \LWR@HTML@__xpinyin_make_pinyin_box:nnn #1#2#3
4 {
5     \color_group_begin: \color_ensure_current:
6     \l__xpinyin_pinyin_box_hook_tl
7     \renewcommand*{\l__xpinyin_ratio_tl}{1}% for pdftotext
8     \__xpinyin_select_font:
9     \clist_if_exist:cTF { c__xpinyin_multiple_ #1 _clist }
10        { \l__xpinyin_multiple_tl \l__xpinyin_format_tl }
11        { \l__xpinyin_format_tl }
12    \ifdefempty{\l__xpinyin_format_tl}
13        {#3}
14        {\LWR@textcurrentcolor{#3}}
15    \color_group_end:
16 }
17 \LWR@formatted{__xpinyin_make_pinyin_box:nnn}


18 \cs_new_protected_nopar:Npn \LWR@HTML@__xpinyin_CJKsymbol:nn #1#2
19  {
20     \__xpinyin_leavevmode:
21     \LWR@htmltagc{ruby}
22     \__xpinyin_save_CJKsymbol:n {#2}\null% \null removes extra space
23     \LWR@htmltagc{rp}(\LWR@htmltagc{/rp\space}
24     \LWR@htmltagc{rt}
25     \__xpinyin_make_pinyin_box:nnn {#1} {#2} { \use:c { c__xpinyin_ #1 _tl } }
26     \LWR@htmltagc{/rt\space}
27     \LWR@htmltagc{rp})\LWR@htmltagc{/rp\space}
28     \LWR@htmltagc{/ruby\space}\null
29  }
30 \LWR@formatted{__xpinyin_CJKsymbol:nn}


31 \cs_new_protected_nopar:Npn \LWR@HTML@__xpinyin_single_CJKsymbol:nn #1#2
32  {
33     \__xpinyin_leavevmode:
34     \LWR@htmltagc{ruby}
35     \__xpinyin_save_CJKsymbol:n {#1}\null% \null removes extra space
```

```
36        \LWR@htmltagc{rp}(\LWR@htmltagc{/rp\space}
37        \LWR@htmltagc{rt}
38        \__xpinyin_make_pinyin_box:xnn
39          { \__xpinyin_to_unicode:n {#1} } {#1} { \__xpinyin_pinyin:n {#2} }
40        \LWR@htmltagc{/rt\space}
41        \LWR@htmltagc{rp})\LWR@htmltagc{/rp\space}
42        \LWR@htmltagc{/ruby\space}\null
43    }
44 \LWR@formatted{__xpinyin_single_CJKsymbol:nn}
45
46 \ExplSyntaxOff
```

The lwarp core uses the following to disable CJK xpinyin for filenames, sideTOC, and footnotes.

```
47 \renewcommand*{\LWR@disablepinyin}{\disablepinyin}
48
49 \FilenameNullify{\LWR@disablepinyin}
```

File 581   **`lwarp-xr.sty`**

§ 693   Package   **xr**

(*Emulates or patches code by* JEAN-PIERRE DRUCBERT, DAVID CARLISLE.)

xr (*Pkg*)   xr is patched for use by lwarp. The `*_html.aux` file is used.

See section 5.18.

for HTML output: 1 \LWR@ProvidesPackagePass{xr}[2024-04-10]%


```
2 \VerifyCommand{\XR@}{56D2754CF8EDBB8863B603F708D65CE9}
3 %\def\XR@[#1][#2]#3{\@testopt{\XR@@{#1}{#2}{#3}}{#3.\XR@ext}}
4 \def\XR@[#1][#2]#3{\@testopt{\XR@@{#1}{#2}{#3_html}}{#3_html.\XR@ext}}
```

File 582   **`lwarp-xr-hyper.sty`**

§ 694   Package   **xr-hyper**

(*Emulates or patches code by* DAVID CARLISLE.)

xr-hyper (*Pkg*)   xr-hyper is replaced by xr, which is modified to accept the optional arguments for `\externaldocument`. So far, no hyperlinks are provided for citations.

See section 5.18.

for HTML output: 1 \LWR@ProvidesPackageDrop{xr-hyper}[2019/10/03]%
2
3 \LWR@origRequirePackage{lwarp-xr}

File 583 **lwarp-xtab.sty**

§ 695 Package **xtab**

(*Emulates or patches code by* PETER WILSON.)

xtab (*Pkg*) xtab is emulated.

for HTML output: 1 \LWR@ProvidesPackageDrop{xtab}[2011/07/31]

⚠ Misplaced alignment tab character & For \tablefirsthead, etc., enclose them as follows:

> \StartDefiningTabulars
> \tablefirsthead
> . . .
> \StopDefiningTabulars

See section 8.10.1.

⚠ lateximage supertabular and xtab are not supported inside a lateximage.

```
 2 \newcommand{\LWRXT@firsthead}{}
 3
 4 \newcommand{\tablefirsthead}[1]{%
 5    \long\gdef\LWRXT@firsthead{#1}%
 6 }
 7
 8 \newcommand{\tablehead}[1]{}
 9
10 \newcommand{\tablelasthead}[1]{}
11
12 \newcommand{\notablelasthead}{}
13
14 \newcommand{\tabletail}[1]{}
15
16 \newcommand{\LWRXT@lasttail}{}
17
18 \newcommand{\tablelasttail}[1]{%
19    \long\gdef\LWRXT@lasttail{#1}%
20 }
```

```
21 \newcommand{\tablecaption}[2][]{%
22    \long\gdef\LWRXT@caption{%
23        \ifblank{#1}%
24            {\caption{#2}}%
25            {\caption[#1]{#2}}%
26    }%
27 }
28
29 \let\topcaption\tablecaption
30 \let\bottomcaption\tablecaption
```

```
31 \newcommand*{\LWRXT@caption}{}
32
33 \newcommand*{\shrinkheight}[1]{}
```

```
34
35 \newcommand*{\xentrystretch}[1]{}
36
37 \NewDocumentEnvironment{xtabular}{s o m}
38 {%
39 \LWR@traceinfo{xtabular}%
40 \table%
41 \LWRXT@caption%
42 \begin{tabular}{#3}%
43 \TabularMacro\ifdefvoid{\LWRXT@firsthead}%
44 {\LWR@getmynexttoken}%
45 {\expandafter\LWR@getmynexttoken\LWRXT@firsthead}%
46 }%
47 {%
48 \ifdefvoid{\LWRXT@lasttail}%
49 {}%
50 {%
51 \TabularMacro\ResumeTabular%
52 \LWRXT@lasttail%
53 }%
54 \end{tabular}%
55 \endtable%

56 \gdef\LWRXT@caption{}%

57 \LWR@traceinfo{xtabular done}%
58 }
59
60 \NewDocumentEnvironment{mpxtabular}{s o m}
61 {\minipage{\linewidth}\xtabular{#3}}
62 {\endxtabular\endminipage}
```

File 584 **lwarp-xunicode.sty**

§ 696   Package   # xunicode

xunicode (*Pkg*)   Error if xunicode is loaded after lwarp.

Patch lwarp-xunicode, but also verify that is was loaded before lwarp:

for HTML output:
```
1 \LWR@loadbefore{xunicode}%
2
3 \LWR@ProvidesPackagePass{xunicode}[2011/09/09]
```

\textcircled becomes a span with a rounded border. \providecommand is used to avoid conflict with textcomp.

```
4 \providecommand*{\LWR@HTML@textcircled}[1]{%
5     \InlineClass[border: 1px solid \LWR@currenttextcolor]{textcircled}{#1}%
6 }
7
8 \LWR@formatted{textcircled}
```

Nullify xunicode macros when generating filenames:

```
9 \FilenameNullify{%
```

```
10    \renewcommand*{\textdegree}{}%
11    \renewcommand*{\textcelsius}{}%
12    \renewcommand*{\textohm}{}%
13    \renewcommand*{\textmu}{}%
14    \renewcommand*{\textlquill}{}%
15    \renewcommand*{\textrquill}{}%
16    \renewcommand*{\textcircledP}{}%
17    \renewcommand*{\texttwelveudash}{}%
18    \renewcommand*{\textthreequartersemdash}{}%
19    \renewcommand*{\textmho}{}%
20    \renewcommand*{\textnaira}{}%
21    \renewcommand*{\textpeso}{}%
22    \renewcommand*{\textrecipe}{}%
23    \renewcommand*{\textinterrobang}{}%
24    \renewcommand*{\textinterrobangdown}{}%
25    \renewcommand*{\textperthousand}{}%
26    \renewcommand*{\textpertenthousand}{}%
27    \renewcommand*{\textbaht}{}%
28    \renewcommand*{\textdiscount}{}%
29    \renewcommand*{\textservicemark}{}%
30    \renewcommand*{\textcircled}[1]{#1}%
31    \renewcommand*{\capitalcedilla}[1]{#1}%
32    \renewcommand*{\capitalogonek}[1]{#1}%
33    \renewcommand*{\capitalgrave}[1]{#1}%
34    \renewcommand*{\capitalacute}[1]{#1}%
35    \renewcommand*{\capitalcircumflex}[1]{#1}%
36    \renewcommand*{\capitaltilde}[1]{#1}%
37    \renewcommand*{\capitaldieresis}[1]{#1}%
38    \renewcommand*{\capitalhungarumlaut}[1]{#1}%
39    \renewcommand*{\capitalring}[1]{#1}%
40    \renewcommand*{\capitalcaron}[1]{#1}%
41    \renewcommand*{\capitalbreve}[1]{#1}%
42    \renewcommand*{\capitalmacron}[1]{#1}%
43    \renewcommand*{\capitaldotaccent}[1]{#1}%
44 }% FilenameNullify
```

File 585   **`lwarp-xurl.sty`**

§ 697   Package   **xurl**

xurl (*Pkg*)   xurl is ignored.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{xurl}[2020/01/14]
2
3 \def\useOriginalUrlSetting{}
```

File 586   **`lwarp-xy.sty`**

§ 698   Package   **xy**

(*Emulates or patches code by* Kristoffer H. Rose, Ross Moore.)

xy (*Pkg*)   xy is patched for use by lwarp.

for HTML output: `1 \LWR@ProvidesPackagePass{xy}[2013/10/06]`

After xy modules have been loaded:

```
2 \AtBeginDocument{
```

The original definitions without a lateximage:

```
3 \LetLtxMacro\LWR@orig@xy\xy
4 \LetLtxMacro\LWR@orig@endxy\endxy
```

The outer-most xy environment is placed in a lateximage, but not more than one
level deep, which would conflict with xy:

```
5 \renewcommand*{\xy}{%
6     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
7         {\addtocounter{LWR@lateximagedepth}{1}}%
8         {\begin{lateximage}[-xy-~\PackageDiagramAltText]?}%
9     \LWR@orig@xy%
10 }
11
12 \renewcommand*{\endxy}{%
13     \LWR@orig@endxy%
14     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{1}%
15         {\addtocounter{LWR@lateximagedepth}{-1}}%
16         {\end{lateximage}}%
17 }
```

The \xybox must use the original definitions of \xy, \endxy:

```
18 \def\xybox#1{%
19     \LWR@orig@xy#1\LWR@orig@endxy%
20     \Edge@c={\rectangleEdge}\computeLeftUpness@%
21 }
```

If \xygraph is used, it is placed inside a lateximage:

```
22 \@ifundefined{xygraph}{}{
23
24 \LetLtxMacro\LWR@origxygraph\xygraph
25
26 \renewcommand{\xygraph}[1]{%
27     \begin{lateximage}[-xy- xygraph \PackageDiagramAltText]?%
28     \LWR@origxygraph{#1}
29     \end{lateximage}
30 }
31
32 }% xygraph defined
33
34 }% AtBeginDocument
```

File 587   **`lwarp-zhlineskip.sty`**

§ 699   Package   # zhlineskip

zhlineskip (*Pkg*)   zhlineskip is ignored.

1 \LWR@ProvidesPackageDrop{zhlineskip}[2019/05/15]

```
2 \newcommand*\SetTextEnvironmentSinglespace[1]{}
3 \newcommand*\RestoreTextEnvironmentLeading[1]{}
4 \newcommand*\SetMathEnvironmentSinglespace[1]{}
5 \newcommand*\RestoreMathEnvironmentLeading[1]{}
```

File 588   **`lwarp-zwpagelayout.sty`**

§ 700   Package   # zwpagelayout

(*Emulates or patches code by* ZDENĚK WAGNER.)

zwpagelayout (*Pkg*)   zwpagelayout is ignored.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{zwpagelayout}[2013/01/13]

2 \def\noBboxes{}
3 \@onlypreamble\noBboxes
4
5 \expandafter\ifx\csname definecolor\endcsname\relax \else
6   \definecolor{cmykblack}{cmyk}{0,0,0,1}
7   \definecolor{grblack}{gray}{0}
8 %   \ifzwpl@redefineblack
9 %     \definecolor{black}{cmyk}{0,0,0,1}\color{black}
10 %   \fi
11  \definecolor{cmykred}{cmyk}{0,1,1,0}
12  \definecolor{cmykgreen}{cmyk}{1,0,1,0}
13  \definecolor{cmykblue}{cmyk}{1,1,0,0}
14  \definecolor{rgbred}{rgb}{1,0,0}
15  \definecolor{rgbgreen}{rgb}{0,1,0}
16  \definecolor{rgbblue}{rgb}{0,0,1}
17 %   \ifzwpl@redefinetocmyk
18 %     \definecolor{red}{cmyk}{0,1,1,0}
19 %     \definecolor{green}{cmyk}{1,0,1,0}
20 %     \definecolor{blue}{cmyk}{1,1,0,0}
21 %   \fi
22 \fi
23
24 \let\OverprintXeTeXExtGState\relax
25
26 \DeclareRobustCommand\SetOverprint{\ignorespaces}
27 \DeclareRobustCommand\SetKnockout{\ignorespaces}
28 \DeclareRobustCommand\textoverprint[1]{{\SetOverprint#1}}
29 \DeclareRobustCommand\textknockout[1]{{\SetKnockout#1}}
30
31 \def\SetPDFminorversion#1{}
32 \@onlypreamble\SetPDFminorversion
33
34 \newcommand*\Vcorr{}
35
36 \DeclareRobustCommand\vb[1][]{}
37 \NewDocumentCommand{\NewOddPage}{* o}{}
38 \NewDocumentCommand{\NewEvenPage}{* o}{}
39 \def\SetOddPageMessage#{\gdef\ZW@oddwarning}
40 \def\SetEvenPageMessage#{\gdef\Z@@evenwarning}
41 \def\ZW@oddwarning{Empty page inserted}\let\ZW@evenwarning\ZW@oddwarning
42
43 \def\clap#1{#1}
```

```
44
45 \def\CropFlap{2in}
46 \def\CropSpine{1in}
47 \def\CropXSpine{1in}
48 \def\CropXtrim{.25in}
49 \def\CropYtrim{.25in}
50 \def\UserWidth{5in}
51 \def\UserLeftMargin{1in}
52 \def\UserRightMargin{1in}
53 \def\UserTopMargin{1in}
54 \def\UserBotMargin{1in}
55 \def\thePageNumber{\LWR@origpound\,\arabic{page}}
56 \ifXeTeX
57 \def\ifcaseZWdriver{\ifcase2}
58 \else
59 \def\ifcaseZWdriver{\ifcase1}
60 \fi
61 \DeclareRobustCommand\ZWifdriver[2]{}
```

File 589   **`lwarp-patch-komascript.sty`**

§ 701   Package   # patch-komascript

lwarp-patch-komascript (*Pkg*)   Patches for komascript classes.

lwarp loads this package when scrbook, scrartcl, or scrreprt classes are detected.

Many features are ignored during the HTML conversion. The goal is source-level compatibility.

\captionformat, \figureformat, and \tableformat are not yet emulated.

⚠   Not fully tested!   Please send bug reports!

Some features have not yet been tested. Please contact the author with any bug reports.

for HTML output: `1 \ProvidesPackage{lwarp-patch-komascript}`

typearea is emulated.

`2 \RequirePackage{lwarp-typearea}`

tocbasic is emulated.

`3 \RequirePackage{lwarp-tocbasic}`

scrextend patches most of the new macros.

`4 \RequirePackage{lwarp-scrextend}`

Indexing macros, simplified for lwarp:

```
5 \AtBeginDocument{
6
7 \renewcommand*{\idx@heading}{%
```

```
 8    \idx@@heading{\indexname}%
 9 }
10
11 \renewenvironment{theindex}{%
12    \idx@heading%
13    \index@preamble\par\nobreak
14       \LetLtxMacro\item\LWR@indexitem%
15       \LetLtxMacro\subitem\LWR@indexsubitem%
16       \LetLtxMacro\subsubitem\LWR@indexsubsubitem%
17 }
18 {}
19
20 \renewcommand*\indexspace{}
21
22 }% AtBeginDocument
```

The \minisec is placed inside a <div> of class minisec.

```
23 \renewcommand*{\minisec}[1]{
24       \begin{BlockClass}{minisec}
25       #1
26       \end{BlockClass}
27 }
```

The part and chapter preambles are placed as plain text just after each heading.

```
28 \@ifundefined{setpartpreamble}{}{
29 \RenewDocumentCommand{\setpartpreamble}{o o +m}{%
30       \renewcommand{\part@preamble}{#3}%
31 }
32 }
33
34 \@ifundefined{setchapterpreamble}{}{
35 \RenewDocumentCommand{\setchapterpreamble}{o o +m}{%
36       \renewcommand{\chapter@preamble}{#3}%
37 }
38 }
```

Do not use \chaptername:

```
39 \renewcommand*{\LWR@printchaptername}{}
```

Simple captions are used in all cases.

```
40 \AtBeginDocument{
41 \AtBeginDocument{
42       \LetLtxMacro\captionbelow\caption
43       \LetLtxMacro\captionabove\caption
44
45       \LetLtxMacro\captionofbelow\captionof
46       \LetLtxMacro\captionofabove\captionof
47 }
48 }
49
50 \RenewDocumentEnvironment{captionbeside}{o m o o o s}
51 {}
52 {%
53       \IfValueTF{#1}%
54             {\caption[#1]{#2}}%
```

```
55          {\caption{#2}}%
56 }
57
58 \RenewDocumentEnvironment{captionofbeside}{m o m o o o s}
59 {}
60 {%
61      \IfValueTF{#2}%
62          {\captionof{#1}[#2]{#3}}%
63          {\captionof{#1}{#3}}%
64 }
65
66 \RenewDocumentCommand{\setcapindent}{s m}{}
67 \renewcommand*{\setcaphanging}{}
68 \renewcommand*{\setcapwidth}[2][]{}
69 \renewcommand*{\setcapdynwidth}[2][]{}
70 \RenewDocumentCommand{\setcapmargin}{s o m}{}
```

File 590   **`lwarp-patch-memoir.sty`**

§ 702   Package   # patch-memoir

(*Emulates or patches code by* PETER WILSON.)

lwarp-patch-memoir (*Pkg*)   Patches for memoir class.

⚠ Not fully tested!   Please send bug reports!

lwarp loads this package when the memoir class is detected.

⚠ captions   lwarp uses caption, which causes a warning from memoir. This is normal. Adjust captions via caption, instead of memoir.

While emulating memoir, lwarp pre-loads a number of packages (section 702.1). This can cause an options clash when the user's document later loads the same ⚠ options clash   packages with options. To fix this problem, specify the options before loading lwarp:

> \documentclass{memoir}
>
> . . .
>
> \PassOptionsToPackage{options_list}{package_name}
>
> . . .
>
> \usepackage{lwarp}
>
> . . .
>
> \usepackage{package_name}

memoir emulates a number of packages, and declares a version date for each ⚠ version numbers   which often does not match the date of the corresponding freestanding package. This can cause warnings about incorrect version numbers. Since lwarp is intended to support the freestanding packages, which are often newer than the date declared by memoir, it is hoped that memoir will update and change its emulated version numbers to match.

\label(bookmark){tag}   \label accepts an optional (bookmark) argument, but this is ignored in HTML.

⚠ comment   The comment environment is from the comment package, and thus requires that the \begin and \end each be on its own line:

```
\begin{comment}
This is a comment.
\end{comment}
```

\newcomment   Comments defined with \newcomment use memoir's defintions, and behave as expected, where the \begin and \end do have to each be on its own line.

⚠ verbatim footnotes   \verbfootnote is not supported.

⚠ \newfootnoteseries   \newfootnoteseries, etc. are not supported.

⚠ page notes   lwarp loads pagenote to perform memoir's pagenote functions, but there are minor differences in \pagenotesubhead and related macros.

page notes with cleveref   To add support for pagenotes with cleveref, add:

```
\crefname{pagenote}{page note}{page notes}
\Crefname{pagenote}{Page note}{Page notes}
```

page note \nameref   Note that for print mode, \nameref print the section name where the page notes are declared in the text, but for HTML it prints the name where the page notes are printed.

⚠ poems   Poem numbering is not supported.

⚠ verbatim   The verbatim environment does not yet support the memoir enhancements. It is currently recommended to load and use fancyvrb instead.

⚠ glossaries   The memoir glossary system is not yet supported by *lwarpmk*. The glossaries package may be used instead, but does require the glossary entries be changed from the memoir syntax to the glossaries syntax.

for HTML output: 1 \ProvidesPackage{lwarp-patch-memoir}

§ 702.1 Packages

These are pre-loaded to provide emulation for many of memoir's functions. memoir pretends that abstract, etc. are already loaded, via its "emulated" package mechanism, but lwarp is directly loading the "lwarp-" version of each, which happens to avoid memoir's emulation system.

```
2 \RequirePackage{lwarp-abstract}% req'd
3 % \RequirePackage{lwarp-array}% no longer req'd
4 \RequirePackage{lwarp-booktabs}% req'd
5 % \RequirePackage{lwarp-ccaption}% emualated below
6 \RequirePackage{lwarp-changepage}% req'd
7 \RequirePackage{lwarp-crop}
8 % \RequirePackage{lwarp-dcolumn}% no longer req'd
9 \RequirePackage{lwarp-enumerate}% req'd
10 \RequirePackage{lwarp-epigraph}% req'd
11 \RequirePackage{lwarp-fancyvrb}% req'd
12 \RequirePackage{lwarp-footmisc}% req'd


13 \let\framed\relax \let\endframed\relax
14 \let\shaded\relax \let\endshaded\relax
15 \let\leftbar\relax \let\endleftbar\relax
16 \let\snugshade\relax \let\endsnugshade\relax
17 \RequirePackage{lwarp-framed}% req'd
18
19 \RequirePackage{lwarp-hanging}% req'd
```

```
20 \RequirePackage{lwarp-makeidx}% req'd
21 \DisemulatePackage{moreverb}
22 \RequirePackage{lwarp-moreverb}
23 \RequirePackage{lwarp-mparhack}
24 \RequirePackage{lwarp-needspace}% req'd
25 \RequirePackage{lwarp-nextpage}% req'd
26 \RequirePackage{lwarp-pagenote}% req'd
27 \RequirePackage{lwarp-parskip}
28 \RequirePackage{lwarp-setspace}% req'd
29 \RequirePackage{lwarp-showidx}
```

```
30 \makeindex
```

```
31 % \RequirePackage{lwarp-tabularx}% no longer req'd
32 \RequirePackage{lwarp-titling}% req'd
33 % \RequirePackage{lwarp-tocbibind}% not emulated by memoir
34 \RequirePackage{lwarp-tocloft}% req'd
35 \RequirePackage{lwarp-verse}% req'd
```

§ 702.2 **Label handling**

Insert the lwarp label mechanism into the memoir package mechanism:

- \LWR@orig@label is the kernel version, or nameref version if loaded before lwarp.

- lwarp's \LWR@new@label uses \LWR@orig@label.

- cleveref then encapsulates all the above with \cref@old@label.

- For a subcaption, cleveref modifies memoir's \sf@memsub@label, but that change is undone by lwarp.

- memoir uses the final \label for subcaptions.

Patches for subfloats to support additional lwarp labels. This is the non-hyperref version from memoir.

```
36 \AtBeginDocument{
37     \renewcommand*{\sf@@memsub@label}[1]{%
38         \@bsphack
39 %         \@mem@kernel@label{#1}%
40         \cref@label{#1}%                          lwarp
41         \LWR@label@createtag{sub@#1}%             lwarp
42         \protected@write\@auxout{}{%
43             \string\newlabel{sub@#1}%
44             {%
45                 {\@nameuse{@@thesub\@captype}}%
46                 {\thepage}%
47                 {\detokenize\expandafter{\@currentlabelname}}% name
48                 {#1}% Href
49                 {}% reserved
50             }%
51         }%
52         \LWR@write@lwarplabel{sub@#1}%            lwarp
53         \@esphack
54     }
55 }
```

§ 702.3 **Page layout**

memoir already set the page size to a default, so it must be forced large for lwarp's use, to avoid tag overflows off the page.

```
56 \setstocksize{190in}{20in}
57 \setlrmarginsandblock{2in}{2in}{*}
58 \setulmarginsandblock{1in}{1in}{*}

59 \renewcommand*{\stockavi}{}
60 \renewcommand*{\stockav}{}
61 \renewcommand*{\stockaiv}{}
62 \renewcommand*{\stockaiii}{}
63 \renewcommand*{\stockavii}{}
64 \renewcommand*{\stockbvi}{}
65 \renewcommand*{\stockbv}{}
66 \renewcommand*{\stockbiv}{}
67 \renewcommand*{\stockbiii}{}
68 \renewcommand*{\stockbvii}{}
69 % \renewcommand*{\stockmetriccrownvo}{}% in docs but not in the package
70 \renewcommand*{\stockmlargecrownvo}{}
71 \renewcommand*{\stockmdemyvo}{}
72 \renewcommand*{\stockmsmallroyalvo}{}
73 \renewcommand*{\pageavi}{}
74 \renewcommand*{\pageavii}{}
75 \renewcommand*{\pageav}{}
76 \renewcommand*{\pageaiv}{}
77 \renewcommand*{\pageaiii}{}
78 \renewcommand*{\pagebvi}{}
79 \renewcommand*{\pagebvii}{}
80 \renewcommand*{\pagebv}{}
81 \renewcommand*{\pagebiv}{}
82 \renewcommand*{\pagebiii}{}
83 % \renewcommand*{\pagemetriccrownvo}{}% in docs but not in the package
84 \renewcommand*{\pagemlargecrownvo}{}
85 \renewcommand*{\pagemdemyvo}{}
86 \renewcommand*{\pagemsmallroyalvo}{}
87
88 \renewcommand*{\stockdbill}{}
89 \renewcommand*{\stockstatement}{}
90 \renewcommand*{\stockexecutive}{}
91 \renewcommand*{\stockletter}{}
92 \renewcommand*{\stockold}{}
93 \renewcommand*{\stocklegal}{}
94 \renewcommand*{\stockledger}{}
95 \renewcommand*{\stockbroadsheet}{}
96 \renewcommand*{\pagedbill}{}
97 \renewcommand*{\pagestatement}{}
98 \renewcommand*{\pageexecutive}{}
99 \renewcommand*{\pageletter}{}
100 \renewcommand*{\pageold}{}
101 \renewcommand*{\pagelegal}{}
102 \renewcommand*{\pageledger}{}
103 \renewcommand*{\pagebroadsheet}{}
104
105 \renewcommand*{\stockpottvo}{}
106 \renewcommand*{\stockfoolscapvo}{}
107 \renewcommand*{\stockcrownvo}{}
108 \renewcommand*{\stockpostvo}{}
109 \renewcommand*{\stocklargecrownvo}{}
```

```
110 \renewcommand*{\stocklargepostvo}{}
111 \renewcommand*{\stocksmalldemyvo}{}
112 \renewcommand*{\stockdemyvo}{}
113 \renewcommand*{\stockmediumvo}{}
114 \renewcommand*{\stocksmallroyalvo}{}
115 \renewcommand*{\stockroyalvo}{}
116 \renewcommand*{\stocksuperroyalvo}{}
117 \renewcommand*{\stockimperialvo}{}
118 \renewcommand*{\pagepottvo}{}
119 \renewcommand*{\pagefoolscapvo}{}
120 \renewcommand*{\pagecrownvo}{}
121 \renewcommand*{\pagepostvo}{}
122 \renewcommand*{\pagelargecrownvo}{}
123 \renewcommand*{\pagelargepostvo}{}
124 \renewcommand*{\pagesmalldemyvo}{}
125 \renewcommand*{\pagedemyvo}{}
126 \renewcommand*{\pagemediumvo}{}
127 \renewcommand*{\pagesmallroyalvo}{}
128 \renewcommand*{\pageroyalvo}{}
129 \renewcommand*{\pagesuperroyalvo}{}
130 \renewcommand*{\pageimperialvo}{}
131
132 \renewcommand*{\memfontfamily}{}
133 \renewcommand*{\memfontenc}{}
134 \renewcommand*{\memfontpack}{}
135
136 \renewcommand*{\anyptfilebase}{}
137 \renewcommand*{\anyptsize}{10}
138
139 \renewcommand*{\setstocksize}[2]{}
140 \renewcommand*{\settrimmedsize}[3]{}
141 \renewcommand*{\settrims}[2]{}
142
143 % \newlength{\lxvchars}
144 % \setlength{\lxvchars}{305pt}
145 % \newlength{\xlvchars}
146 % \setlength{\xlvchars}{190pt}
147 \renewcommand*{\setxlvchars}[1]{}
148 \renewcommand*{\setlxvchars}[1]{}
149
150 \renewcommand*{\settypeblocksize}[3]{}
151 \renewcommand*{\setlrmargins}[3]{}
152 \renewcommand*{\setlrmarginsandblock}[3]{}
153 \renewcommand*{\setbinding}[1]{}
154 \renewcommand*{\setulmargins}[3]{}
155 \renewcommand*{\setulmarginsandblock}[3]{}
156 \renewcommand*{\setcolsepandrule}[2]{}
157
158 \renewcommand*{\setheadfoot}[2]{}
159 \renewcommand*{\setheaderspaces}[3]{}
160 \renewcommand*{\setmarginnotes}[3]{}
161 \renewcommand*{\setfootins}[2]{}
162 \renewcommand*{\checkandfixthelayout}[1][]{}
163 \renewcommand*{\checkthelayout}[1]{}
164 \renewcommand*{\fixthelayout}{}
165 %
166 % \newlength{\stockheight}
167 % \newlength{\trimtop}
168 % \newlength{\trimedge}
169 % \newlength{\stockwidth}
```

```
170 % \newlength{\spinemargin}
171 % \newlength{\foremargin}
172 % \newlength{\uppermargin}
173 % \newlength{\headmargin}
174 %
175 \renewcommand*{\typeoutlayout}{}
176 \renewcommand*{\typeoutstandardlayout}{}
177 \renewcommand*{\settypeoutlayoutunit}[1]{}
178 \renewcommand*{\fixpdflayout}{}
179 \renewcommand*{\fixdvipslayout}{}
180
181 \renewcommand*{\medievalpage}[1][]{}
182 \renewcommand*{\isopage}[1][]{}
183 \renewcommand*{\semiisopage}[1][]{}
184
185 \renewcommand{\setpagebl}[3]{}
186 \renewcommand{\setpageml}[3]{}
187 \renewcommand{\setpagetl}[3]{}
188 \renewcommand{\setpagetm}[3]{}
189 \renewcommand{\setpagetr}[3]{}
190 \renewcommand{\setpagemr}[3]{}
191 \renewcommand{\setpagebr}[3]{}
192 \renewcommand{\setpagebm}[3]{}
193 \renewcommand{\setpagecc}[3]{}
```

§ 702.4 **Text and fonts**

```
194 \let\miniscule\tiny
195 \let\HUGE\Huge
196
197 \renewcommand*{\abnormalparskip}[1]{}
198 \renewcommand*{\nonzeroparskip}{}
199 \renewcommand*{\traditionalparskip}{}
200
201 \let\onelineskip\baselineskip
202
203 \let\OnehalfSpacing\onehalfspacing
204 \let\DoubleSpacing\doublespacing
205 \renewcommand*{\setPagenoteSpacing}[1]{}
206 \renewcommand*{\setFloatSpacing}[1]{}


207 \renewcommand{\SingleSpacing}{\@ifstar\singlespacing\singlespacing}


208 \let\setSingleSpace\SetSinglespace
209 \let\SingleSpace\singlespace
210 \let\endSingleSpace\endsinglespace
211 \let\Spacing\spacing
212 \let\endSpacing\endspacing
213 \let\OnehalfSpace\onehalfspace
214 \let\endOnehalfSpace\endonehalfspace
215 \csletcs{OnehalfSpace*}{onehalfspace}
216 \csletcs{endOnehalfSpace*}{endonehalfspace}
217 \let\DoubleSpace\doublespace
218 \let\endDoubleSpace\enddoublespace
219 \csletcs{DoubleSpace*}{doublespace}
220 \csletcs{endDoubleSpace*}{enddoublespace}
221 \renewcommand*{\setDisplayskipStretch}[1]{}
222 \renewcommand*{\memdskipstretch}{}
223 \renewcommand*{\noDisplayskipStretch}{}
224 \renewcommand*{\memdskips}{}
```

```
225
226 \renewcommand*{\midsloppy}{}
227 \renewenvironment*{midsloppypar}{}{}
228
229 \renewcommand*{\sloppybottom}{}
```

§702.5 Titles

```
230 \csletcs{titlingpage*}{titlingpage}
231 \csletcs{endtitlingpage*}{endtitlingpage}
232 \let\titlingpageend\relax
233 \newcommand{\titlingpageend}[2]{}
234 \let\andnext\and
235 \renewcommand*{\thanksmarkstyle}[1]{}
236
237 \renewcommand{\thanksfootmark}{%
238     \thanksscript{\tamark}%
239 }
240
241 % \newlength{\thanksmarksep}% already provided by memoir

242 \renewcommand\titlingpageend[2]{}
```

§702.6 Abstracts

```
243 % \newlength{\absindent}
244 % \newlength{\absparsep}
245 \renewcommand*{\abstractcol}{}
246 \renewcommand*{\abstractintoc}{}
247 \renewcommand*{\abstractnum}{}
248 \renewcommand*{\abstractrunin}{}
```

§702.7 Docment divisions

\book   * (⟨*2:PDF name*⟩) [⟨*3:TOC name*⟩] [⟨*4:PDF name*⟩] (⟨*5:PDF name*⟩) {⟨*6:name*⟩}

```
249 \DeclareDocumentCommand{\book}{s d() o o d() m}{%
250     \LWR@section{#1}{#3}{#6}{book}%
251 }

252 \def\@apppage{%
253     \part*{\appendixpagename}
254 }
255 \renewcommand\mempreaddapppagetotochook{}
256 \renewcommand\mempostaddapppagetotochook{}
257
258 \def\@sapppage{%
259     \part*{\appendixpagename}
260 }

261 \DeclareDocumentCommand{\mainmatter}{s}{%
262     \booltrue{LWR@mainmatter}%
263 }
264
265 \DeclareDocumentCommand{\frontmatter}{s}{%
266     \boolfalse{LWR@mainmatter}%
267 }
```

```
268 \renewcommand*{\raggedbottomsection}{}
269 \renewcommand*{\normalbottomsection}{}
270 \renewcommand*{\bottomsectionskip}{}
271 \renewcommand*{\bottomsectionpenalty}{}
272 \csletcs{appendixpage*}{appendixpage}
273 \renewcommand*{\namedsubappendices}{}
274 \renewcommand*{\unnamedsubappendices}{}
275 \renewcommand*{\beforebookskip}{}
276 \renewcommand*{\afterbookskip}{}
277 \renewcommand*{\beforepartskip}{}
278 \renewcommand*{\afterpartskip}{}
279 \renewcommand*{\midbookskip}{}
280 \renewcommand*{\midpartskip}{}
281 \renewcommand*{\printbookname}{}
282 \renewcommand*{\booknamefont}{}
283 \renewcommand*{\booknamenum}{}
284 \renewcommand*{\printbooknum}{}
285 \renewcommand*{\booknumfont}{}
286 \renewcommand*{\printpartname}{}
287 \renewcommand*{\partnamefont}{}
288 \renewcommand*{\partnamenum}{}
289 \renewcommand*{\printpartnum}{}
290 \renewcommand*{\partnumfont}{}
291 \renewcommand*{\printbooktitle}[1]{}
292 \renewcommand*{\booktitlefont}{}
293 \renewcommand{\printparttitle}[1]{}
294 \renewcommand*{\parttitlefont}{}
295 \renewcommand*{\bookpageend}{}
296 \renewcommand*{\bookblankpage}{}
297 \renewcommand*{\nobookblankpage}{}
298 \renewcommand*{\partpageend}{}
299 \renewcommand*{\partblankpage}{}
300 \renewcommand*{\nopartblankpage}{}
301 \RenewDocumentCommand{\newleadpage}{s o m m}{}% todo
302 \RenewDocumentCommand{\renewleadpage}{s o m m}{}% todo
303 \renewcommand*{\leadpagetoclevel}{chapter}
304
305 \renewcommand*{\openright}{}
306 \renewcommand*{\openleft}{}
307 \renewcommand*{\openany}{}
308 \renewcommand*{\clearforchapter}{}
309 \renewcommand*{\memendofchapterhook}{}
310 \renewcommand*{\chapterheadstart}{}
311 % \newlength{\beforechapskip}
312 \renewcommand*{\afterchapternum}{}
313 % \newlength{\midchapskip}
314 \renewcommand*{\afterchaptertitle}{}
315 % \newlength{\afterchapskip}
316 \renewcommand*{\printchaptername}{}
317 \renewcommand*{\chapnamefont}{}
318 \renewcommand*{\chapternamenum}{}
319 \renewcommand*{\printchapternum}{}
320 \renewcommand*{\chapnumfont}{}
321 \renewcommand{\printchaptertitle}[1]{}
322 \renewcommand*{\chaptitlefont}{}
323 \renewcommand*{\printchapternonum}{}
324 \renewcommand*{\indentafterchapter}{}
325 \renewcommand*{\noindentafterchapter}{}
326 \renewcommand*{\insertchapterspace}{}
327
```

```
328 \renewcommand*{\chapterstyle}[1]{}
329 \renewcommand{\makechapterstyle}[2]{}
330 \renewcommand*{\chapindent}{}
331 \let\chapterprecis\cftchapterprecis
332 \let\chapterprecishere\cftchapterprecishere
333 \let\chapterprecistoc\cftchapterprecistoc
334 \renewcommand*{\precisfont}{}
335 \renewcommand*{\prechapterprecis}{}
336 \renewcommand*{\postchapterprecis}{}
337 \renewcommand{\precistoctext}[1]{}
338 \renewcommand*{\precistocfont}{}
339 \renewcommand*{\precistocformat}{}
340 % \newlength{\prechapterprecisshift}
341
342 \renewcommand*{\setbeforesecskip}[1]{}
343 \renewcommand*{\setaftersecskip}[1]{}
344 \renewcommand*{\setsecindent}[1]{}
345 \renewcommand*{\setsecheadstyle}[1]{}
346 \renewcommand*{\setbeforesubsecskip}[1]{}
347 \renewcommand*{\setaftersubsecskip}[1]{}
348 \renewcommand*{\setsubsecindent}[1]{}
349 \renewcommand*{\setsubsecheadstyle}[1]{}
350 \renewcommand*{\setbeforesubsubsecskip}[1]{}
351 \renewcommand*{\setaftersubsubsecskip}[1]{}
352 \renewcommand*{\setsubsubsecindent}[1]{}
353 \renewcommand*{\setsubsubsecheadstyle}[1]{}
354 \renewcommand*{\setbeforeparaskip}[1]{}
355 \renewcommand*{\setafterparaskip}[1]{}
356 \renewcommand*{\setparaindent}[1]{}
357 \renewcommand*{\setparaheadstyle}[1]{}
358 \renewcommand*{\setbeforesubparaskip}[1]{}
359 \renewcommand*{\setaftersubparaskip}[1]{}
360 \renewcommand*{\setsubparaindent}[1]{}
361 \renewcommand*{\setsubparaheadstyle}[1]{}
362 \renewcommand{\@hangfrom}[1]{#1}
363 \renewcommand{\sethangfrom}[1]{}
364 \renewcommand{\setsecnumformat}[1]{}
365
366 \renewcommand*{\hangsecnum}{}
367 \renewcommand*{\defaultsecnum}{}
368
369 \renewcommand*{\sechook}{}
370 \renewcommand{\setsechook}[1]{}
371 \renewcommand*{\subsechook}{}
372 \renewcommand{\setsubsechook}[1]{}
373 \renewcommand*{\subsubsechook}{}
374 \renewcommand{\setsubsubsechook}[1]{}
375 \renewcommand*{\parahook}{}
376 \renewcommand{\setparahook}[1]{}
377 \renewcommand*{\subparahook}{}
378 \renewcommand{\setsubparahook}[1]{}
379
380 \RenewDocumentCommand{\plainbreak}{s m}{\begin{center}~\end{center}}
381
382 \RenewDocumentCommand{\fancybreak}{s +m}{%
383     \begin{center}#2\end{center}%
384 }
385
386 \RenewDocumentCommand{\plainfancybreak}{s m m +m}{%
387     \begin{center}#4\end{center}%
```

```
388 }
389
390 \RenewDocumentCommand{\pfbreak}{s}{%
391     \begin{center}
392     \pfbreakdisplay
393     \end{center}
394 }
395
396 % \newlength{\pfbreakskip}
397 \renewcommand{\pfbreakdisplay}{*\quad*\quad*}
398
399 \renewcommand{\makeheadstyles}[2]{}
400 \renewcommand*{\headstyles}[1]{}
```

§ 702.8 **Pagination and headers**

```
401 \renewcommand*{\savepagenumber}{}
402 \renewcommand*{\restorepagenumber}{}
403 \renewcommand*{\uppercaseheads}{}
404 \renewcommand*{\nouppercaseheads}{}
405
406 \renewcommand*{\bookpagemark}[1]{}
407 \renewcommand*{\partmark}[1]{}
408 \renewcommand*{\bibmark}{}
409 \renewcommand*{\indexmark}{}
410 \renewcommand*{\glossarymark}{}
411
412 \LWR@origpagestyle{empty}
413 \renewcommand*{\ps@empty}{}
414 \renewcommand*{\makepagestyle}[1]{}
415 \renewcommand*{\emptypshook}{}%
416 % \renewcommand*{\empty@oddhead}{}
417 % \renewcommand*{\empty@oddfoot}{}
418 % \renewcommand*{\empty@evenhead}{}
419 % \renewcommand*{\empty@evenfoot}{}
420 \renewcommand*{\@oddhead}{}
421 \renewcommand*{\@oddfoot}{}
422 \renewcommand*{\@evenhead}{}
423 \renewcommand*{\@evenfoot}{}
424 \renewcommand*{\aliaspagestyle}[2]{}
425 \renewcommand*{\copypagestyle}[2]{}
426
427 \renewcommand*{\makeevenhead}[4]{}
428 \renewcommand*{\makeoddhead}[4]{}
429 \renewcommand*{\makeevenfoot}[4]{}
430 \renewcommand*{\makeoddfoot}[4]{}
431 \renewcommand*{\makerunningwidth}[3]{}
432 % \newlength{\headwidth}
433 \renewcommand*{\makeheadrule}[3]{}
434 \renewcommand*{\makefootrule}[3]{}
435 \renewcommand*{\makeheadfootruleprefix}[3]{}
436 % \newlength{\normalrulethickness}
437 % \setlength{\normalrulethickness}{.4pt}
438 % \newlength{\footruleheight}
439 % \newlength{\footruleskip}
440 \renewcommand*{\makeheadposition}[5]{}
441 \renewcommand{\makepsmarks}[2]{}
442 \renewcommand*{\makeheadfootstrut}[3]{}
443 \renewcommand{\createmark}[5]{\csdef{#1mark}[1]{}}
```

```
444 \renewcommand{\createplainmark}[3]{\csdef{#1mark}{}}

445 \renewcommand{\memUChead}[1]{}
446 \renewcommand*{\clearplainmark}[1]{}
447 \renewcommand*{\clearmark}[1]{}
448 \renewcommand{\addtopsmarks}[3]{}
449 \renewcommand{\ifonlyfloats}[2]{#2}
450 \renewcommand*{\mergepagefloatstyle}[3]{}
451
452 \renewcommand*{\framepichead}{}
453 \renewcommand*{\framepictextfoot}{}
454 \renewcommand*{\framepichook}{}
455 \renewcommand*{\showheadfootlocoff}{}
456 \renewcommand*{\showtextblocklocoff}{}
```

§ 702.9 Paragraphs and lists

```
457 \renewcommand{\hangfrom}[1]{#1}
458 \let\centerfloat\centering
459 \renewcommand*{\raggedyright}[1][]{}
460 % \newlength{\ragrparindent}
461 \renewcommand{\sourceatright}[2][]{\attribution{#2}}
462 \let\memorigdbs\LWR@endofline

463 \renewcommand*{\memorigpar}{\par}

464 \let\atcentercr\LWR@endofline
465
466 \renewcommand*{\linenottooshort}[1][]{}
467 \renewcommand*{\russianpar}{}
468 \renewcommand*{\lastlinerulefill}{}
469 \renewcommand*{\lastlineparrule}{}
470 \renewcommand*{\justlastraggedleft}{}
471 \renewcommand*{\raggedrightthenleft}{}
472 \renewcommand*{\leftcenterright}{}
473
474 \renewcommand{\leftspringright}[4]{%
475     \begin{minipage}{#1\linewidth}#3\end{minipage}\qquad%
476   \begin{minipage}{#2\linewidth}\begin{flushright}#4\end{flushright}\end{minipage}%
477 }
478
479 \renewenvironment*{blockdescription}
480 {\LWR@descriptionstart\LWR@origdescription}
481 {\enddescription}
482
483 \renewcommand*{\blockdescriptionlabel}[1]{\textbf{#1}}
484 \renewenvironment*{labelled}[1]{\begin{description}}{\end{description}}
485 \renewenvironment*{flexlabelled}[6]{\begin{description}}{\end{description}}
486 \renewcommand*{\tightlists}{}
487 \renewcommand*{\defaultlists}{}
488 \RenewDocumentCommand{\firmlists}{s}{}
489 \renewcommand*{\firmlist}{}
490 \renewcommand*{\tightlist}{}
491 \renewcommand*{\zerotrivseps}{}
492 \renewcommand*{\savetrivseps}{}
493 \renewcommand*{\restoretrivseps}{}
```

§ 702.10 Contents lists

```
494 \csletcs{tableofcontents*}{tableofcontents}
```

```
495 \csletcs{listoffigures*}{listoffigures}
496 \csletcs{listoftables*}{listoftables}
497 \renewenvironment{KeepFromToc}{}{}
498 \renewcommand*{\onecoltocetc}{}
499 \renewcommand*{\twocoltocetc}{}
500 \renewcommand*{\ensureonecol}{}
501 \renewcommand*{\restorefromonecol}{}
502 \renewcommand*{\doccoltocetc}{}
503
504 \renewcommand{\tocheadstart}{}
505 \renewcommand{\printtoctitle}[1]{}
506 \renewcommand{\tocmark}{}
507 \renewcommand{\aftertoctitle}{}
508 \renewcommand{\lofheadstart}{}
509 \renewcommand{\printloftitle}[1]{}
510 \renewcommand{\lofmark}{}
511 \renewcommand{\afterloftitle}{}
512 \renewcommand{\lotheadstart}{}
513 \renewcommand{\printlottitle}[1]{}
514 \renewcommand{\lotmark}{}
515 \renewcommand{\afterlottitle}{}
516
517 \renewcommand*{\setpnumwidth}[1]{}
518 \renewcommand*{\setrmarg}[1]{}
519 \renewcommand*{\cftbookbreak}{}
520 \renewcommand*{\cftpartbreak}{}
521 \renewcommand*{\cftchapterbreak}{}

522 % \newlength{\cftbeforebookskip}
523 % \newlength{\cftbookindent}
524 % \newlength{\cftbooknumwidth}
525 \renewcommand*{\cftbookfont}{}
526 \renewcommand*{\cftbookname}{}
527 \renewcommand*{\cftbookpresnum}{}
528 \renewcommand*{\cftbookaftersnum}{}
529 \renewcommand*{\cftbookaftersnumb}{}
530 \renewcommand*{\cftbookleader}{}
531 \renewcommand*{\cftbookdotsep}{1}
532 \renewcommand*{\cftbookpagefont}{}
533 \renewcommand*{\cftbookafterpnum}{}
534 \renewcommand*{\cftbookformatpnum}[1]{}
535 \renewcommand*{\cftbookformatpnumhook}[1]{}
```

Part is already defined by tocloft.

```
536 % \newlength{\cftbeforechapterskip}
537 % \newlength{\cftchapterindent}
538 % \newlength{\cftchapternumwidth}
539 \renewcommand*{\cftchapterfont}{}
540 \renewcommand*{\cftchaptername}{}
541 \renewcommand*{\cftchapterpresnum}{}
542 \renewcommand*{\cftchapteraftersnum}{}
543 \renewcommand*{\cftchapteraftersnumb}{}
544 \renewcommand*{\cftchapterleader}{}
545 \renewcommand*{\cftchapterdotsep}{1}
546 \renewcommand*{\cftchapterpagefont}{}
547 \renewcommand*{\cftchapterafterpnum}{}
548 \renewcommand*{\cftchapterformatpnum}[1]{}
549 \renewcommand*{\cftchapterformatpnumhook}[1]{}
```

```
550 % \newlength{\cftbeforesectionskip}
551 % \newlength{\cftsectionindent}
552 % \newlength{\cftsectionnumwidth}
553 \renewcommand*{\cftsectionfont}{}
554 \renewcommand*{\cftsectionname}{}
555 \renewcommand*{\cftsectionpresnum}{}
556 \renewcommand*{\cftsectionaftersnum}{}
557 \renewcommand*{\cftsectionaftersnumb}{}
558 \renewcommand*{\cftsectionleader}{}
559 \renewcommand*{\cftsectiondotsep}{1}
560 \renewcommand*{\cftsectionpagefont}{}
561 \renewcommand*{\cftsectionafterpnum}{}
562 \renewcommand*{\cftsectionformatpnum}[1]{}
563 \renewcommand*{\cftsectionformatpnumhook}[1]{}

564 % \newlength{\cftbeforesubsectionskip}
565 % \newlength{\cftsubsectionindent}
566 % \newlength{\cftsubsectionnumwidth}
567 \renewcommand*{\cftsubsectionfont}{}
568 \renewcommand*{\cftsubsectionname}{}
569 \renewcommand*{\cftsubsectionpresnum}{}
570 \renewcommand*{\cftsubsectionaftersnum}{}
571 \renewcommand*{\cftsubsectionaftersnumb}{}
572 \renewcommand*{\cftsubsectionleader}{}
573 \renewcommand*{\cftsubsectiondotsep}{1}
574 \renewcommand*{\cftsubsectionpagefont}{}
575 \renewcommand*{\cftsubsectionafterpnum}{}
576 \renewcommand*{\cftsubsectionformatpnum}[1]{}
577 \renewcommand*{\cftsubsectionformatpnumhook}[1]{}

578 % \newlength{\cftbeforesubsubsectionskip}
579 % \newlength{\cftsubsubsectionindent}
580 % \newlength{\cftsubsubsectionnumwidth}
581 \renewcommand*{\cftsubsubsectionfont}{}
582 \renewcommand*{\cftsubsubsectionname}{}
583 \renewcommand*{\cftsubsubsectionpresnum}{}
584 \renewcommand*{\cftsubsubsectionaftersnum}{}
585 \renewcommand*{\cftsubsubsectionaftersnumb}{}
586 \renewcommand*{\cftsubsubsectionleader}{}
587 \renewcommand*{\cftsubsubsectiondotsep}{1}
588 \renewcommand*{\cftsubsubsectionpagefont}{}
589 \renewcommand*{\cftsubsubsectionafterpnum}{}
590 \renewcommand*{\cftsubsubsectionformatpnum}[1]{}
591 \renewcommand*{\cftsubsubsectionformatpnumhook}[1]{}

592 % \newlength{\cftbeforeparagraphskip}
593 % \newlength{\cftparagraphindent}
594 % \newlength{\cftparagraphnumwidth}
595 \renewcommand*{\cftparagraphfont}{}
596 \renewcommand*{\cftparagraphname}{}
597 \renewcommand*{\cftparagraphpresnum}{}
598 \renewcommand*{\cftparagraphaftersnum}{}
599 \renewcommand*{\cftparagraphaftersnumb}{}
600 \renewcommand*{\cftparagraphleader}{}
601 \renewcommand*{\cftparagraphdotsep}{1}
602 \renewcommand*{\cftparagraphpagefont}{}
603 \renewcommand*{\cftparagraphafterpnum}{}
604 \renewcommand*{\cftparagraphformatpnum}[1]{}
605 \renewcommand*{\cftparagraphformatpnumhook}[1]{}

606 % \newlength{\cftbeforesubparagraphskip}
```

```
607 % \newlength{\cftsubparagraphindent}
608 % \newlength{\cftsubparagraphnumwidth}
609 \renewcommand*{\cftsubparagraphfont}{}
610 \renewcommand*{\cftsubparagraphname}{}
611 \renewcommand*{\cftsubparagraphpresnum}{}
612 \renewcommand*{\cftsubparagraphaftersnum}{}
613 \renewcommand*{\cftsubparagraphaftersnumb}{}
614 \renewcommand*{\cftsubparagraphleader}{}
615 \renewcommand*{\cftsubparagraphdotsep}{1}
616 \renewcommand*{\cftsubparagraphpagefont}{}
617 \renewcommand*{\cftsubparagraphafterpnum}{}
618 \renewcommand*{\cftsubparagraphformatpnum}[1]{}
619 \renewcommand*{\cftsubparagraphformatpnumhook}[1]{}

620 % \newlength{\cftbeforefigureskip}
621 % \newlength{\cftfigureindent}
622 % \newlength{\cftfigurenumwidth}
623 \renewcommand*{\cftfigurefont}{}
624 \renewcommand*{\cftfigurename}{}
625 \renewcommand*{\cftfigurepresnum}{}
626 \renewcommand*{\cftfigureaftersnum}{}
627 \renewcommand*{\cftfigureaftersnumb}{}
628 \renewcommand*{\cftfigureleader}{}
629 \renewcommand*{\cftfiguredotsep}{1}
630 \renewcommand*{\cftfigurepagefont}{}
631 \renewcommand*{\cftfigureafterpnum}{}
632 \renewcommand*{\cftfigureformatpnum}[1]{}
633 \renewcommand*{\cftfigureformatpnumhook}[1]{}

634 % \newlength{\cftbeforesubfigureskip}
635 % \newlength{\cftsubfigureindent}
636 % \newlength{\cftsubfigurenumwidth}
637 \newcommand*{\cftsubfigurefont}{}
638 \newcommand*{\cftsubfigurename}{}
639 \newcommand*{\cftsubfigurepresnum}{}
640 \newcommand*{\cftsubfigureaftersnum}{}
641 \newcommand*{\cftsubfigureaftersnumb}{}
642 \newcommand*{\cftsubfigureleader}{}
643 \newcommand*{\cftsubfiguredotsep}{1}
644 \newcommand*{\cftsubfigurepagefont}{}
645 \newcommand*{\cftsubfigureafterpnum}{}
646 \newcommand*{\cftsubfigureformatpnum}[1]{}
647 \newcommand*{\cftsubfigureformatpnumhook}[1]{}

648 % \newlength{\cftbeforetableskip}
649 % \newlength{\cfttableindent}
650 % \newlength{\cfttablenumwidth}
651 \renewcommand*{\cfttablefont}{}
652 \renewcommand*{\cfttablename}{}
653 \renewcommand*{\cfttablepresnum}{}
654 \renewcommand*{\cfttableaftersnum}{}
655 \renewcommand*{\cfttableaftersnumb}{}
656 \renewcommand*{\cfttableleader}{}
657 \renewcommand*{\cfttabledotsep}{1}
658 \renewcommand*{\cfttablepagefont}{}
659 \renewcommand*{\cfttableafterpnum}{}
660 \renewcommand*{\cfttableformatpnum}[1]{}
661 \renewcommand*{\cfttableformatpnumhook}[1]{}

662 % \newlength{\cftbeforesubtableskip}
663 % \newlength{\cftsubtableindent}
```

```
664 % \newlength{\cftsubtablenumwidth}
665 \newcommand*{\cftsubtablefont}{}
666 \newcommand*{\cftsubtablename}{}
667 \newcommand*{\cftsubtablepresnum}{}
668 \newcommand*{\cftsubtableaftersnum}{}
669 \newcommand*{\cftsubtableaftersnumb}{}
670 \newcommand*{\cftsubtableleader}{}
671 \newcommand*{\cftsubtabledotsep}{1}
672 \newcommand*{\cftsubtablepagefont}{}
673 \newcommand*{\cftsubtableafterpnum}{}
674 \newcommand*{\cftsubtableformatpnum}[1]{}
675 \newcommand*{\cftsubtableformatpnumhook}[1]{}

676 \renewcommand*{\booknumberline}[1]{}
677 \renewcommand*{\partnumberline}[1]{}
678 \renewcommand*{\chapternumberline}[1]{}
679 \renewcommand*{\numberlinehook}[1]{}
680 % \renewcommand*{\cftwhatismyname}{}%
681 \renewcommand*{\booknumberlinehook}[1]{}
682 \renewcommand*{\partnumberlinehook}[1]{}
683 \renewcommand*{\chapternumberlinehook}[1]{}
684 \renewcommand{\numberlinebox}[2]{}
685 \renewcommand{\booknumberlinebox}[2]{}
686 \renewcommand{\partnumberlinebox}[2]{}
687 \renewcommand{\chapternumberlinebox}[2]{}
688 %
689 % \newlength{\cftparfillskip}
690 \renewcommand*{\cftpagenumbersoff}[1]{}
691 \renewcommand*{\cftpagenumberson}[1]{}
692 \renewcommand*{\cftlocalchange}[3]{}
693 \renewcommand*{\cftaddtitleline}[4]{}
694 \renewcommand*{\cftaddnumtitleline}[4]{}
695 \renewcommand{\cftinsertcode}[2]{}
696 \renewcommand{\cftinserthook}[2]{}
697 \renewcommand{\settocpreprocessor}[2]{}
698 \DeclareRobustCommand{\cftpagenumbersoff}[1]{}
699 \DeclareRobustCommand{\cftpagenumberson}[1]{}
```

§ 702.11 **Floats and captions**

\@xfloat

\@xdblfloat   Reestablish lwarp's takeover the float handing, which memoir tried to grab:

```
700 \AtBeginDocument{
701 \def\@xfloat #1[#2]{%
702     \LWR@floatbegin{#1}[#2]
703     \normalsize
704     \@nameuse{#1adjustment}%
705     \LWR@futurenonspacelet\LWR@mynexttoken\LWR@floatalignment%
706 }
707 \def\@xdblfloat #1[#2]{%
708     \LWR@floatbegin{#1}[#2]
709     \normalsize
710     \@nameuse{#1adjustment}%
711     \LWR@futurenonspacelet\LWR@mynexttoken\LWR@floatalignment%
712 }
713 }
```

\newfloat   [⟨*1: within*⟩] {⟨*2: type*⟩} {⟨*3: ext*⟩} {⟨*4: capname*⟩}

```
714 \RenewDocumentCommand{\newfloat}{o m m m}{%
715     \def\LWR@tempone{#4}%
716     \def\LWR@temptwo{\@nameuse{#2name}}%
717     \ifdefequal{\LWR@tempone}{\LWR@temptwo}{% recursive name, already defined
718         \IfValueTF{#1}%
719             {\DeclareFloatingEnvironment[fileext=#3,within=#1]{#2}}%
720             {\DeclareFloatingEnvironment[fileext=#3]{#2}}%
721     }{% not recursive name
722         \IfValueTF{#1}%
723          {\DeclareFloatingEnvironment[fileext=#3,within=#1,name={#4}]{#2}}%
724             {\DeclareFloatingEnvironment[fileext=#3,name={#4}]{#2}}%
725     }%
```

newfloat package automatically creates the \listof command for new floats, but **float** does not, so remove \listof here in case it is manually created later.

```
726     \cslet{listof#2s}\relax%
727     \cslet{listof#2es}\relax%
728 }
```

\newlistof   [⟨*within*⟩] {⟨*type*⟩} {⟨*ext*⟩} {⟨*listofname*⟩}

Emulated through the \newfloat mechanism. Note that **memoir** uses a different syntax than **tocloft** for the name.

```
729 \RenewDocumentCommand{\newlistof}{o m m m}
730 {%
731     \IfValueTF{#1}%
732         {\newlistentry[#1]{#2}{#3}{0}}%
733         {\newlistentry{#2}{#3}{0}}%
734     \@namedef{ext@#2}{#3}%
735     \@ifundefined{c@#3depth}{\newcounter{#3depth}}{}%
736     \setcounter{#3depth}{1}%
737     \@namedef{#3mark}{}%
738     \@namedef{#2}{\LWR@listof{#2}{#4}}%
739     \@namedef{@cftmake#3title}{}%
740     \@ifundefined{cftbefore#3titleskip}{%
741         \expandafter\newlength\csname cftbefore#3titleskip\endcsname%
742         \expandafter\newlength\csname cftafter#3titleskip\endcsname%
743     }{}%
744     \@namedef{cft#3titlefont}{}%
745     \@namedef{cftafter#3title}{}%
746     \@namedef{cft#3prehook}{}%
747     \@namedef{cft#3posthook}{}%
748 }
```

```
749 \renewcommand{\setfloatadjustment}[2]{}
```

Borrowed from the **lwarp** version of **keyfloat**:

```
750 \NewDocumentEnvironment{KFLTmemoir@marginfloat}{O{-1.2ex} m}
751 {% start
752     \LWR@BlockClassWP{float:right; width:2in; margin:10pt}{}(note){marginblock}%
753         \renewcommand*{\@captype}{#2}%
754 }
755 {%
756     \endLWR@BlockClassWP%
757 }
758
759 \DeclareDocumentEnvironment{marginfigure}{o}
760  {\begin{KFLTmemoir@marginfloat}{figure}}
```

```
761    {\end{KFLTmemoir@marginfloat}}
762
763 \DeclareDocumentEnvironment{margintable}{o}
764    {\begin{KFLTmemoir@marginfloat}{table}}
765    {\end{KFLTmemoir@marginfloat}}

766 \renewcommand{\setmarginfloatcaptionadjustment}[2]{}
767 \renewcommand{\setmpjustification}[2]{}
768 \renewcommand*{\mpjustification}{}
769 \renewcommand*{\setfloatlocations}[2]{}
770 \DeclareDocumentCommand{\suppressfloats}{o}{}
771 \renewcommand*{\FloatBlock}{}
772 \renewcommand*{\FloatBlockAllowAbove}{}
773 \renewcommand*{\FloatBlockAllowBelow}{}
774 \renewcommand*{\setFloatBlockFor}{}
775
776 \renewcommand{\captiontitlefinal}[1]{}
```

\flegtable, \flegfigure, \flegtoctable, \flegtocfigure are defined by memoir
using \newfloat. These are defined with an @ in ccaption.

```
777 \renewcommand{\flegtable}{\tablename}
778 \renewcommand{\flegfigure}{\figurename}
779 \renewcommand{\flegtoctable}{}
780 \renewcommand{\flegtocfigure}{}


781 \renewcommand{\@makesubfloatcaption}[2]{%
782    \minipagefullwidth
783    \begin{minipage}{\linewidth}%
784    #1 \ignorespaces #2 \unskip%
785    \end{minipage}
786 }
787
788 \renewcommand*{\tightsubcaptions}{}
789 \renewcommand*{\loosesubcaptions}{}
790
791 \renewcommand*{\subcaptionsize}[1]{}
792 \renewcommand*{\subcaptionlabelfont}[1]{}
793 \renewcommand*{\subcaptionfont}[1]{}
794 \renewcommand*{\subcaptionstyle}[1]{}
795
796 \renewcommand*{\hangsubcaption}{}
797 \renewcommand*{\shortsubcaption}{}
798 \renewcommand*{\normalsubcaption}{}
```

\AfterEndPreamble now required for sidecaption.

```
799 \AfterEndPreamble{%
800 \RenewDocumentEnvironment{sidecaption}{o m o}
801 {}
802 {%
803    \IfValueTF{#1}{\caption[#1]{#2}}{\caption{#2}}%
804    \IfValueT{#3}{\label{#3}}%
805 }
806 }
807
808 % \newlength{\sidecapwidth}
809 % \newlength{\sidecapsep}
```

```
810 \renewcommand*{\setsidecaps}[2]{}
811 \renewcommand*{\sidecapmargin}[1]{}
812 % \newif\ifscapmargleft
813 \scapmargleftfalse
814 \renewcommand*{\setsidecappos}[1]{}
```

Env sidecontcaption

```
815 \RenewDocumentEnvironment{sidecontcaption}{m o}
816 {}
817 {%
818     \ifdef{\ContinuedFloat}%
819         {\ContinuedFloat}%
820         {\addtocounter{\@captype}{-1}}%
821     \caption{#1}%
```

Without \@captype, the section is referred to instead.

```
822     \IfValueT{#2}{\label[\@captype]{#2}}%
823 }
```

\sidenamedlegend does not appear to use the toc argument.

```
824 \renewenvironment{sidenamedlegend}[2][]{
825     \begin{center}
826     \@nameuse{\@captype name}\CaptionSeparator#2
827     \end{center}
828 }
829 {}
830
831 \renewenvironment{sidelegend}[1]
832 {\begin{center}
833     #1
834
835 }
836 {\end{center}}
837
838 \renewcommand*{\sidecapstyle}{}
839 \renewcommand*{\overridescapmargin}[1]{}
840 % \newlength{\sidecapraise}
841 \renewcommand*{\sidecapfloatwidth}{\linewidth}
842
843 \LetLtxMacro\ctabular\tabular
844 \LetLtxMacro\endctabular\endtabular
845
846 \renewcommand{\autorows}[5][]{%
847     #5%
848 }
849
850 \renewcommand{\autocols}[5][]{%
851     #5%
852 }
```

§ 702.12 Footnotes and page notes

```
853 \renewcommand*{\feetabovefloat}{}
854 \renewcommand*{\feetbelowfloat}{}
855 \renewcommand*{\feetatbottom}{}
856
857 \renewcommand*{\verbfootnote}[2][]{%
858     \PackageError{lwarp,memoir}%
859     {Verbatim footnotes are not yet supported by lwarp}%
860     {This may be improved some day.}%
861 }
862
863 \renewcommand*{\plainfootnotes}{}
864 \renewcommand*{\twocolumnfootnotes}{}
865 \renewcommand*{\threecolumnfootnotes}{}
866 \renewcommand*{\paragraphfootnotes}{}
867 \renewcommand*{\footfudgefiddle}{}
868
869 \renewcommand*{\newfootnoteseries}[1]{%
870     \PackageError{lwarp,memoir}%
871     {Memoir footnote series are not yet supported by lwarp}%
872     {This may be improved some day.}%
873 }
874
875 \renewcommand*{\plainfootstyle}[1]{}
876 \renewcommand*{\twocolumnfootstyle}[1]{}
877 \renewcommand*{\threecolumnfootstyle}[1]{}
878 \renewcommand*{\paragraphfootstyle}[1]{}
879
880 \renewcommand*{\footfootmark}{}
881 \renewcommand*{\footmarkstyle}[1]{}
882
883 % \newlength{\footmarkwidth}
884 % \newlength{\footmarksep}
885 % \newlength{\footparindent}
886
887 \renewcommand*{\foottextfont}{}
888
889 \renewcommand*{\marginparmargin}[1]{}
890 \renewcommand*{\sideparmargin}[1]{}
891
892 \LetLtxMacro\sidepar\marginpar
893 \renewcommand*{\sideparfont}{}
894 \renewcommand*{\sideparform}{}
895 \LWR@providelength{\sideparvshift}
896
897 \renewcommand*{\parnopar}{}
898
899 \renewcommand{\sidebar}[1]{\begin{quote}#1\end{quote}}
900 \renewcommand*{\sidebarmargin}[1]{}
901 \renewcommand*{\sidebarfont}{}
902 \renewcommand*{\sidebarform}{}
903 % \newlength{\sidebarhsep}
904 % \newlength{\sidebarvsep}
905 % \newlength{\sidebarwidth}
906 % \newlength{\sidebartopsep}
907 \renewcommand{\setsidebarheight}[1]{}
908 \renewcommand*{\setsidebars}[6]{}
909 \renewcommand*{\footnotesatfoot}{}
910 \renewcommand*{\footnotesinmargin}{}
```

```
911
912 \LetLtxMacro\sidefootnote\footnote
913 \LetLtxMacro\sidefootnotemark\footnotemark
914 \LetLtxMacro\sidefootnotetext\footnotetext
915
916 \renewcommand*{\sidefootmargin}[1]{}
917 % \newlength{\sidefoothsep}
918 % \newlength{\sidefootvsep}
919 % \newlength{\sidefootwidth}
920 % \newlength{\sidefootadjust}
921 % \newlength{\sidefootheight}
922 \renewcommand*{\setsidefootheight}[1]{}
923 % \renewcommand*{\sidefootfont}{}% in docs but not in the package
924 \renewcommand*{\setsidefeet}[6]{}
925 \renewcommand*{\sidefootmarkstyle}[1]{}
926 \renewcommand*{\sidefoottextfont}{}
927 \renewcommand*{\sidefootform}{}

928 \renewcommand*{\continuousnotenums}{\pncontopttrue}% from pagenote
929 \renewcommand*{\notepageref}{}
930 \renewcommand*{\prenotetext}{}
931 \renewcommand*{\postnotetext}{}
932 \LetLtxMacro\printpageinnoteshyperref\printpageinnotes
933 \renewcommand*{\foottopagenote}{}
934 \renewcommand*{\pagetofootnote}{}
```

\m@m@wrpnote

\startnoteentrystart   To have **cleveref** work with page note labels, the following patch writes \thepagenote
                          and also adds \arabic{pagenote} to the first argument written to the .ent file:

> \startnoteentry{{\thepagenote}{\arabic{pagenote}}} . . .

The arabic value is required for **cleveref**. \thepagenote becomes \@firstoftwo#1
and the arabic value becomes \@secondoftwo#1.

⚠ \nameref   Note that for print mode,\nameref print the section name where the page notes
            are declared in the text, but for HTML it prints the name where the page notes are
            printed.

```
935 \VerifyCommand[lwarp][lwarp-patch-memoir]{\m@m@wrpnote}{D2AE41FE9A265B639F7074AB2AF29976}
936
937 \xpatchcmd{\m@m@wrpnote}
938     {\string\startnoteentry{\thepagenote}}
939     {\string\startnoteentry{{\thepagenote}{\arabic{pagenote}}}}
940     {}
941     {\LWR@patcherror{memoir}{m@m@wrpnote}}
942
943 \VerifyCommand[lwarp][lwarp-patch-memoir]{\startnoteentrystart}{2A595EA1DC483451337C33072604EDD6}
944
945 \renewcommand\startnoteentrystart[4]{%
946   \prenoteinnotes%
947   \noteidinnotes{\@firstoftwo#1}{#2}%
948   \@ifmtarg{#2}{%
949 %       \phantomsection\def\@currentlabel{#1}%          original
950       \def\@currentlabel{\@firstoftwo#1}%              lwarp
951       \def\cref@currentlabel{%                         lwarp
952           [pagenote][\@secondoftwo#1][]\@firstoftwo#1% lwarp
953       }%                                               lwarp
954   }{}%
955   \pagenoteanchor{#4}%
956   \pageinnotes{#3}%
```

```
957   \prenotetext%
958 }
```

§ 702.13   **Decorative text**

```
959 \renewcommand*{\epigraphposition}[1]{}
960 \renewcommand*{\epigraphtextposition}[1]{}
961 \renewcommand*{\epigraphsourceposition}[1]{}
962 \renewcommand*{\epigraphfontsize}[1]{}
963 \renewcommand*{\epigraphforheader}[2][]{}
964 \renewcommand*{\epigraphpicture}{}
```

§ 702.14   **Poetry**

```
965 \renewcommand*{\vinphantom}{}
966 \renewcommand*{\vleftofline}[1]{#1}
967 % \let\linenumberfrequency\poemlines
968 % \renewcommand*{\linenumberfont}[1]{}
969
970 \DeclareDocumentCommand{\PoemTitle}{s o o m}{%
971     \IfValueTF{#2}%
972         {\poemtitle[#2]{#4}}%
973         {\poemtitle{#4}}%
974 }
975
976 \renewcommand*{\NumberPoemTitle}{}
977 \renewcommand*{\PlainPoemTitle}{}
978 \renewcommand*{\poemtitlepstyle}{}
979 \renewcommand*{\poemtitlestarmark}[1]{}
980 \renewcommand*{\poemtitlestarpstyle}{}
981 \renewcommand*{\PoemTitleheadstart}{}
982 \renewcommand*{\printPoemTitlenonum}{}
983 \renewcommand*{\printPoemTitlenum}{}
984 \renewcommand*{\afterPoemTitlenum}{}
985 \renewcommand*{\printPoemTitletitle}[1]{}
986 \renewcommand*{\afterPoemTitle}{}
987 \newlength{\midpoemtitleskip}
988 \renewcommand*{\PoemTitlenumfont}{}
989 \renewcommand*{\PoemTitlefont}{}
```

§ 702.15   **Boxes, verbatims and files**

```
990 \renewenvironment{qframe}{\framed}{\endframed}
991 \renewenvironment{qshade}{\shaded}{\endshaded}

992 \renewcommand*{\setverbatimfont}[1]{}
993 \renewcommand*{\tabson}[1]{}% disabled as of 3.8.2
994 \renewcommand*{\tabsoff}{}% disabled as of 3.8.2
995 \renewcommand*{\wrappingon}{}% disabled as of 3.8.2
996 \renewcommand*{\wrappingoff}{}% disabled as of 3.8.2
997 \renewcommand*{\verbatimindent}{}% no longer used as of 3.8.2
998 \renewcommand*{\verbatimbreakchar}[1]{}% no longer used as of 3.8.2

999 \DefineVerbatimEnvironment{fboxverbatim}{Verbatim}{frame=single}
```

boxedverbatim is already defined by moreverb. boxedverbatim* does not appear to work at all, even in a minimal print memoir document.

```
1000 \renewcommand*{\bvbox}{}
```

```
1001 \renewcommand*{\bvtopandtail}{}
1002 \renewcommand*{\bvsides}{}
1003 \renewcommand*{\nobvbox}{}
1004 % \newlength\bvboxsep
1005 \renewcommand*{\bvtoprulehook}{}
1006 \renewcommand*{\bvtopmidhook}{}
1007 \renewcommand*{\bvendrulehook}{}
1008 \renewcommand*{\bvleftsidehook}{}
1009 \renewcommand*{\bvrightsidehook}{}
1010 \renewcommand*{\bvperpagetrue}{}
1011 \renewcommand*{\bvperpagefalse}{}
1012 \renewcommand{\bvtopofpage}[1]{}
1013 \renewcommand{\bvendofpage}[1]{}
1014 \renewcommand*{\linenumberfrequency}[1]{}
1015 \renewcommand*{\resetbvlinenumber}{}
1016 \renewcommand*{\setbvlinenums}[2]{}
1017 \renewcommand*{\linenumberfont}[1]{}
1018 \renewcommand*{\bvnumbersinside}{}
1019 \renewcommand*{\bvnumbersoutside}{}
```

§ 702.16 Cross referencing

```
1020 \renewcommand*{\fref}[1]{\cref{#1}}
1021 \renewcommand*{\tref}[1]{\cref{#1}}
1022 \renewcommand*{\pref}[1]{\cpageref{#1}}
1023 \renewcommand*{\Aref}[1]{\cref{#1}}
1024 \renewcommand*{\Bref}[1]{\cref{#1}}
1025 \renewcommand*{\Pref}[1]{\cref{#1}}
1026 \renewcommand*{\Sref}[1]{\cref{#1}}
1027 \renewcommand*{\figurerefname}{Figure}
1028 \renewcommand*{\tablerefname}{Table}
1029 \renewcommand*{\pagerefname}{page}
1030 \renewcommand*{\bookrefname}{Book~}
1031 \renewcommand*{\partrefname}{Part~}
1032 \renewcommand*{\chapterrefname}{Chapter~}
1033 \renewcommand*{\sectionrefname}{\S}
1034 \renewcommand*{\appendixrefname}{Appendix~}
1035 \LetLtxMacro\titleref\nameref
1036 \renewcommand*{\headnameref}{}
1037 \renewcommand*{\tocnameref}{}
```

\currenttitle has been removed from memoir.

```
1038 \renewcommand*{\theTitleReference}[2]{}
1039 \renewcommand*{\namerefon}{}
1040 \renewcommand*{\namerefoff}{}
```

§ 702.17 Back matter

\@@wrindexhyp   Redefined to write the LWR@autoindex counter instead of page. Note that memoir has two versions, depending on the use of hyperref.

```
1041 \AtBeginDocument{
1042
1043 \VerifyCommand[lwarp][lwarp-patch-memoir]{\@@wrindexhyp}{8DA7E3C8BE7A830442D98EA033147F63}
1044
1045 \def\@@wrindexhyp#1||\\{%
1046     \addtocounter{LWR@autoindex}{1}%              lwarp
1047 %     \ifshowindexmark\@showidx{#1}\fi
1048     \protected@write\@auxout{}%
```

```
1049 %            {\string\@@wrindexm@m{\@idxfile}{#1}{\thepage}}%
1050            {\string\@@wrindexm@m{\@idxfile}{#1}{\arabic{LWR@autoindex}}}}% lwarp
```

The label is assigned after the file write to avoid conflict with cleveref.

```
1051     \label{LWRindex-\arabic{LWR@autoindex}}%      lwarp
1052     \endgroup
1053     \@esphack}%
```

\@@wrspindexhyp   \specialindex behaves like a regular \index, pointing to where \specialindex
                         is used. If \specialindex is used inside a figure or table after the \caption, then
                         the hyperlink will be given the name of that particular figure or table.

```
1054 \def\@@wrspindexhyp#1||\\{%
1055     \addtocounter{LWR@autoindex}{1}%
1056 %      \ifshowindexmark\@showidx{#1}\fi
1057     \protected@write\@auxout{}%
1058 %          {\string\@@wrindexm@m{\@idxfile}{#1}{\@nameuse{the\@sptheidx}}}%
1059          {\string\@@wrindexm@m{\@idxfile}{#1}{\arabic{LWR@autoindex}}}%
```

The label is assigned after the file write to avoid conflict with cleveref.

```
1060     \label{LWRindex-\arabic{LWR@autoindex}}%
1061     \endgroup
1062     \@esphack}%
1063
1064 }% \AtBeginDocument
```

\@spindex   Patched to append _html to the file:

```
1065 \renewcommand{\@spindex}[2]{%
1066   \@ifundefined{#1@idxfile}%
1067   {\ifreportnoidxfile
1068      \@memwarn{Undefined index file #1}%
1069     \fi
1070     \begingroup
1071     \@sanitize
1072     \@nowrindex}%
1073   {\def\@idxfile{#1_html}%
1074   \def\@sptheidx{#2}%
1075   \begingroup
1076   \@sanitize
1077   \@wrspindex}}
```

\makeindex   Patched to use _html filename and \BaseJobname:

```
1078 \catcode`\_=12%
1079 \renewcommand*{\makeindex}[1][\BaseJobname]{%
1080   \if@filesw
1081     \def\gindex{\@bsphack%
1082       \@ifnextchar [{\@index}{\@index[\BaseJobname]}}
1083     \def\specialindex{\@bsphack\@spindex}%
1084     \makememindexhook
1085     \expandafter\newwrite\csname #1@idxfile\endcsname
1086   \expandafter\immediate\openout \csname #1@idxfile\endcsname #1_html.idx\relax
1087     \typeout{Writing index file #1_html.idx }%
1088   \fi}
1089 \catcode`\_=8%
```

\printindex   Patched to use _html filename and \BaseJobname. This will later be patched by

the lwarp core.

```
1090 \catcode`\_=12%
1091 \renewcommand{\printindex}[1][\BaseJobname]{\@input@{#1_html.ind}}
1092 \catcode`\_=8%

1093 \DeclareDocumentCommand{\newblock}{}{}
1094 %
1095 \renewcommand*{\showindexmarks}{}
1096 \renewcommand*{\hideindexmarks}{}
1097
1098 \renewcommand*{\xindyindex}{}
```

§ 702.18 **Miscellaneous**

```
1099 \renewcommand*{\changemarks}{}
1100 \renewcommand*{\nochangemarks}{}
1101 \renewcommand*{\added}[1]{}
1102 \renewcommand*{\deleted}[1]{}
1103 \renewcommand*{\changed}[1]{}
1104
1105 \renewcommand*{\showtrimsoff}{}
1106 \renewcommand*{\showtrimson}{}
1107 \renewcommand*{\trimXmarks}{}
1108 \renewcommand*{\trimLmarks}{}
1109 \renewcommand*{\trimFrame}{}
1110 \renewcommand*{\trimNone}{}
1111 \renewcommand*\trimmarkscolor{}
1112 \renewcommand*{\trimmarks}{}
1113 \renewcommand*{\tmarktl}{}
1114 \renewcommand*{\tmarktr}{}
1115 \renewcommand*{\tmarkbr}{}
1116 \renewcommand*{\tmarkbl}{}
1117 \renewcommand*{\tmarktm}{}
1118 \renewcommand*{\tmarkmr}{}
1119 \renewcommand*{\tmarkbm}{}
1120 \renewcommand*{\tmarkml}{}
1121 \renewcommand*{\trimmark}{}
1122 \renewcommand*{\quarkmarks}{}
1123 \renewcommand*{\registrationColour}[1]{}
1124
1125 \renewcommand*{\leavespergathering}[1]{}
1126
1127 \renewcommand*{\noprelistbreak}{}
1128
1129 \renewcommand*{\cleartorecto}{}
1130 \renewcommand*{\cleartoverso}{}
1131
1132 \renewenvironment{vplace}[1][]{}{}
```

§ 702.19 **ccaption emulation**

```
1133 \renewcommand*{\captiondelim}[1]{\renewcommand*{\CaptionSeparator}{#1}}
1134 \renewcommand*{\captionnamefont}[1]{}
1135 \renewcommand*{\captiontitlefont}[1]{}
1136 \renewcommand*{\flushleftright}{}
1137 \renewcommand*{\centerlastline}{}
1138 \renewcommand*{\captionstyle}[2][]{}
1139 \DeclareDocumentCommand{\captionwidth}{m}{}
1140 \renewcommand*{\changecaptionwidth}{}
```

```
1141 \renewcommand*{\normalcaptionwidth}{}
1142 \renewcommand*{\hangcaption}{}
1143 \renewcommand*{\indentcaption}[1]{}
1144 \renewcommand*{\normalcaption}{}
1145 \renewcommand{\precaption}[1]{}
1146 \renewcommand{\postcaption}[1]{}
1147 \renewcommand{\midbicaption}[1]{}
1148 \renewcommand{\contcaption}[1]{%
1149 %    \ContinuedFloat%
1150 %    \caption{#1}%
1151     \begin{LWR@figcaption}% later becomes \caption*
1152     \LWR@isolate{\@nameuse{\@captype name}}~%
1153     \thechapter.\the\value{\@captype}\CaptionSeparator\LWR@isolate{#1}%
1154     \end{LWR@figcaption}%
1155 }

1156 \newlength{\abovelegendskip}
1157 \setlength{\abovelegendskip}{0.5\baselineskip}
1158 \newlength{\belowlegendskip}
1159 \setlength{\belowlegendskip}{\abovelegendskip}
```

The extra \\ here forces a <br> in HTML when \legend is used in a \marginpar.

```
1160 \renewcommand{\legend}[1]{\begin{center}#1\\\end{center}}
1161
1162 \renewcommand{\namedlegend}[2][]{%
1163     \begin{center}
1164     \@nameuse{fleg\@captype}\CaptionSeparator#2\\
1165     \end{center}
1166     \@nameuse{flegtoc\@captype}{#1}
1167 }
```

\flegtable, \flegfigure, \flegtoctable, \flegtocfigure are defined by memoir using \newfloat. These are defined with an @ in ccaption.

```
1168 \renewcommand{\newfixedcaption}[3][\caption]{%
1169   \renewcommand{#2}{\def\@captype{#3}#1}}
1170 \renewcommand{\renewfixedcaption}[3][\caption]{%
1171   \renewcommand{#2}{\def\@captype{#3}#1}}
1172 \renewcommand{\providefixedcaption}[3][\caption]{%
1173   \providecommand{#2}{\def\@captype{#3}#1}}
1174
1175 \renewcommand{\bitwonumcaption}[6][]{%
1176     \ifblank{#2}{\caption{#3}}{\caption[#2]{#3}}%
1177     \addtocounter{\@captype}{-1}%
1178     \begingroup%
1179     \csdef{\@captype name}{#4}%
1180     \ifblank{#5}{\caption{#6}}{\caption[#5]{#6}}%
1181     \endgroup%
1182     \ifblank{#1}{}{\label{#1}}%
1183 }
1184
1185 \LetLtxMacro\bionenumcaption\bitwonumcaption% todo
1186
1187 \renewcommand{\bicaption}[5][]{%
1188     \ifblank{#2}{\caption{#3}}{\caption[#2]{#3}}%
1189     \begin{LWR@figcaption}% later becomes \caption*
1190     \LWR@isolate{#4} % space
1191     \thechapter.\the\value{\@captype}\CaptionSeparator\LWR@isolate{#5}%
1192     \end{LWR@figcaption}%
```

```
1193    \ifblank{#1}{}{\label{#1}}%
1194 }
1195
1196 \renewcommand{\bicontcaption}[3]{%
1197    \contcaption{#1}%
1198    \begingroup%
1199    \csdef{\@captype name}{#2}%
1200    \contcaption{#3}%
1201    \endgroup%
1202 }
```

Only in ccaption, not in memoir:

```
1203 % \LetLtxMacro\longbitwonumcaption\bitwonumcaption%
1204 % \LetLtxMacro\longbionenumcaption\bitwonumcaption%
1205 % \LetLtxMacro\longbicaption\bicaption%
```

Patches for subfloats to support additional lwarp labels:

```
1206 \renewcommand{\@memsubbody}{%
1207    \bgroup
1208    \let\label=\memsub@label
1209    \ifdonemaincaption\else
1210      \advance\csname c@\@captype\endcsname\@ne
1211    \fi
1212 % \refstepcounter{sub\@captype}\@contkeep%
1213 %    \leavevmode%                    lwarp
1214    \@ifnextchar [%
1215      {\@memsubfig}%
1216      {\@memsubfig[\@empty]}}
1217
1218 \renewcommand{\@memcontsubbody}{%
1219    \bgroup
1220    \let\label=\memsub@label
1221    \@contset
1222 % \refstepcounter{sub\@captype}\@contkeep%
1223 %    \leavevmode%                    lwarp
1224    \@ifnextchar [%
1225      {\@memsubfig}%
1226      {\@memsubfig[\@empty]}}
1227
1228
1229 \long\def\@memsubfloat#1[#2][#3]#4{%
1230 %    \@tempcnta=\@ne
1231 %    \if@tightsubcap
1232 %      \if@minipage
1233 %        \@tempcnta=\z@
1234 %      \else
1235 %      \ifdim\lastskip=\z@
1236 %        \@tempcnta=\@ne
1237 %      \else
1238 %        \@tempcnta=\tw@
1239 %      \fi
1240 %    \fi
1241 %    \fi
1242 %    \if@contbotsub
1243 %      \def\subfig@top{\subfloattopskip}%
1244 %      \def\subfig@bottom{\subfloatbottomskip}%
1245 %    \else
1246 %      \def\subfig@top{\subfloatbottomskip}%
```

```
1247 %     \def\subfig@bottom{\subfloattopskip}%
1248 %     \fi
1249 %     \setbox\@tempboxa \hbox{#4}%
1250 %     \@tempdima=\wd\@tempboxa
1251 %     \vbox
1252   \bgroup%
1253     \mem@step@subcounter%
1254 %     \vbox
1255     \LWR@stoppars%
1256     \minipagefullwidth%              lwarp
1257     \begin{minipage}{\linewidth}%    lwarp
1258     \bgroup
1259 %     \ifcase\@tempcnta
1260 %       \@minipagefalse
1261 %     \or
1262 %       \vspace{\subfig@top}
1263 %     \or
1264 %       \ifdim \lastskip=\z@ \else
1265 %         \@tempskipb\subfig@top\@xaddvskip
1266 %       \fi
1267 %     \fi
1268     \if@contbotsub
1269       #4% \box\@tempboxa
1270       \egroup
1271       \ifx \@empty#3\relax \else
1272 %         \vskip\subfloatcapskip
1273       \@memsubcaption{#1}{#2}{#3}%
1274       \fi
1275     \else
1276       \ifx \@empty#3\relax \else
1277         \@memsubcaption{#1}{#2}{#3}%
1278 %         \vskip\subfloatcapskip
1279 %         \vskip\subfloatcaptopadj
1280       \fi\egroup
1281       #4% \box\@tempboxa
1282     \fi
1283 %     \vspace{\subfig@bottom}
1284     \end{minipage}%                  lwarp
1285     \LWR@startpars%                  lwarp
1286   \egroup
1287 \egroup
1288 }
```

§ 702.20 **Final patchwork**

```
1289 \newlistof{tableofcontents}{toc}{\contentsname}
1290 \newlistof{listoffigures}{lof}{\listfigurename}
1291 \newlistof{listoftables}{lot}{\listtablename}
```

File 591 **`lwarp-common-multimedia.sty`**

§ 703 Package **common-multimedia**

lwarp-common-multimedia (*Pkg*) Common code for multimedia, movie15, and media9.

The packages multimedia, movie15, and media9 are supported.

HTML5 <audio> and <video> objects are created for .mp3 and .mp4 files.

HTML5 <embed> objects are created for http and ftp links.

\href links are created for other media types. (Unfortunately, there is not much overlap between the file types supported for print output and the file types supported by HTML5.)

For media9, a multimedia object is inserted for each addresource=, as well as each flashvars source= and src=. This may result in duplicate objects.

Undesired objects may be nullified by placing them inside \warpprintonly or the warpprint environment.

Each HTML multimedia object includes the poster text, except for <embed> objects. For movie15, the text option is supported to specify the poster text.

The width, height, and totalheight options are supported. The HTML object is scaled according to the display width, correctly compensating for either tall or wide viewports.

Other options are ignored.

media9 \addmediapath is supported. It is assumed that the same path structure will exist for the HTML document.

HTML5 media controls are always specified for each <audio> and <video> object.

media9 slideshows are not supported.

\hyperlinkmovie, \movieref, and \mediabutton are not supported.

3D objects are not supported.

If using a YouTube™ video, use an "embedded" URL with .../embed/... instead of .../v/...

for HTML output:

```
1 \ProvidesPackage{lwarp-common-multimedia}[2019/04/22]
```

```
2 \RequirePackage{xkeyval}
3
4 \define@key{LWR@multimedia}{width}{\setlength{\LWR@multimedia@width}{#1}}
5 \define@key{LWR@multimedia}{height}{\setlength{\LWR@multimedia@height}{#1}}
6 \define@key{LWR@multimedia}{totalheight}{\setlength{\LWR@multimedia@height}{#1}}
```

```
7 \newlength{\LWR@multimedia@width}
8 \newlength{\LWR@multimedia@height}
9 \newlength{\LWR@multimedia@maxdimension}
```

\LWR@multimedia@printsize   Proportional to \linewidth and the viewport's smaller dimension. This scales each object such that it will always fit on the screen, even if a tall or wide object inside a tall or wide viewport.

```
10 \newcommand*{\LWR@multimedia@printsize}{%
11     \setlength{\LWR@multimedia@maxdimension}{%
12         \maxof%
13             {\linewidth}%
14             {\maxof{\LWR@multimedia@width}{\LWR@multimedia@height}}%
15     }%
```

```
16    \setlength{\LWR@multimedia@maxdimension}{1.1\LWR@multimedia@maxdimension}%
17    \ifdimgreater{\LWR@multimedia@width}{0pt}{%
18        width:%
19            \LWR@printpercentlength%
20                {\LWR@multimedia@width}%
21                {\LWR@multimedia@maxdimension}vmin ; % space
22    }{}%
23    \ifdimgreater{\LWR@multimedia@height}{0pt}{%
24        height:%
25            \LWR@printpercentlength%
26                {\LWR@multimedia@height}%
27                {\LWR@multimedia@maxdimension}vmin ; % space
28    }{}%
29 }
```

\LWR@multimedia@fileAV   {⟨*poster text*⟩} {⟨*filename*⟩} {⟨*audio/video*⟩} {⟨*mimetype*⟩}

Creates a video or audio from a file. The 2019/10 update of the LATEX kernel may cause extra quotes to be added in the filenames. They are removed here.

```
30 \newcommand*{\LWR@multimedia@fileAV}[4]{%
31 \IfFileExists{#2}{% also sets \@filef@und
32 \StrSubstitute[100]{\@filef@und}{"}{}[\LWR@parsedfilename]%
```

The container <div> is sized as desired.

```
33    \ifstrequal{#3}{audio}{%
34        \begin{BlockClass}{AVviewport}
35    }{%
36      \begin{BlockClass}[\LWR@multimedia@printsize\ margin:auto]{AVviewport}
37    }
```

Paragraph tags are unnecessary for the A / V tags.

```
38    \LWR@stoppars
```

The A / V element is 100% of the container.

```
39    \LWR@htmltag{%
40        #3\ % space
41        \ifstrequal{#3}{audio}{}{%
42            width=\textquotedbl{}100\%\textquotedbl\ % space
43            height=\textquotedbl{}100\%\textquotedbl\ % space
44        }%
45        controls%
46    }\LWR@orignewline
```

The file source and type:

```
47    \LWR@htmltag{%
48        source % space
49        src=\textquotedbl%
50        \LWR@parsedfilename\unskip\textquotedbl\ % space
51        type=\textquotedbl{}#4\textquotedbl}
```

The poster text inside paragraph tags, along with a reference to the file.

```
52    \LWR@startpars
53    \LWR@href{\LWR@parsedfilename}{#1}
54    \LWR@stoppars
```

Finish.

```
55    \LWR@htmltag{/#3}\LWR@orignewline
56    \end{BlockClass}
```

```
57 }{%
58     \PackageError{lwarp-common-multimedia}
59         {File '#2' not found}
60         {Perhaps an incorrect path?}
61 }%
62 }
```

\LWR@multimedia@httpAV   {⟨*poster text*⟩} {⟨*filename*⟩} {⟨*audio/video*⟩} {⟨*mimetype*⟩}

Creates a video or audio from a URL link.

```
63 \newcommand*{\LWR@multimedia@httpAV}[4]{%
```

The container <div> is sized as desired.

```
64     \ifstrequal{#3}{audio}{%
65         \begin{BlockClass}{AVviewport}
66     }{%
67       \begin{BlockClass}[\LWR@multimedia@printsize\ margin:auto]{AVviewport}
68     }
```

Paragraph tags are unnecessary for the A / V tags.

```
69     \LWR@stoppars
```

The A / V element is 100% of the container.

```
70     \LWR@htmltag{%
71         #3\ % space
72         \ifstrequal{#3}{audio}{}{%
73             width=\textquotedbl{}100\%\textquotedbl\ % space
74             height=\textquotedbl{}100\%\textquotedbl\ controls%
75         }%
76     }\LWR@orignewline
```

The file source and type:

```
77     \LWR@htmltag{%
78         source % space
79         src=\textquotedbl#2\textquotedbl\ % space
80         type=\textquotedbl#4\textquotedbl}
```

The poster text inside paragraph tags, along with a reference to the URL.

```
81     \LWR@startpars
82     \LWR@href{#2}{#1}
83     \LWR@stoppars
```

Finish.

```
84     \LWR@htmltag{/#3}\LWR@orignewline
85     \end{BlockClass}
86 }
```

\LWR@multimedia@AV   {⟨*poster text*⟩} {⟨*filename*⟩} {⟨*audio/video*⟩} {⟨*mimetype*⟩}

Creates an audio or video from a file or a URL.

```
87 \newcommand*{\LWR@multimedia@AV}[4]{%
88     \IfBeginWith{#2}{http}%
89         {\LWR@multimedia@httpAV{#1}{#2}{#3}{#4}}%
90         {%
91             \IfBeginWith{#2}{HTTP}%
92                 {\LWR@multimedia@httpAV{#1}{#2}{#3}{#4}}%
93                 {\LWR@multimedia@fileAV{#1}{#2}{#3}{#4}}%
94         }%
```

```
95 }
```

\LWR@multimedia@embed   {⟨*poster text*⟩} {⟨*URL or filename*⟩} {⟨*mime type*⟩}

Embeds multimedia of an arbitrary type. The poster text is not used, as it would appear along with the video if the <embed> element is supported.

```
 96 \newcommand*{\LWR@multimedia@embed}[3]{%
 97     \begin{BlockClass}[width:100\%]{AVviewport}%
 98     \LWR@stoppars
 99         \LWR@htmltag{%
100             embed % space
101             \ifblank{#3}{}{type=\textquotedbl#3\textquotedbl\ }%
102         style=\textquotedbl\LWR@multimedia@printsize\ margin:auto\textquotedbl\ % space
103             src=\textquotedbl#2\textquotedbl\ % space
104         }%
105     \LWR@startpars
106     \end{BlockClass}
107 }
```

Error message if the comment character is used among the arguments of \LWR@multimedia@percenterror   \LWR@multimediab.

```
108 \newcommand*{\LWR@multimedia@percenterror}{%
109     \PackageError{lwarp-media9}
110     {%
111         Do not use a percent comment between\MessageBreak
112         \protect\includemedia\space arguments%
113     }
114     {%
115         Percent is changed to a regular character\MessageBreak
116         to allow its use inside a URL.%
117     }
118 }
```

\LWR@multimediab   [⟨*options*⟩] {⟨*poster text*⟩} {⟨*filename*⟩}

Creates multimedia. Examines the file extension to determine the type. If not a supported type, creates an embedded object if it has a URL. If neither, create a link to the unsupported object.

```
119 \newcommand*{\LWR@multimediab}[3][]{%
```

Error if the percent character appears among the arguments. This could happen since the comment character has been temporarily disabled, for use in a URL.

```
120     \if#1\@percentchar\LWR@multimedia@percenterror\fi%
121     \if#2\@percentchar\LWR@multimedia@percenterror\fi%
122     \if#3\@percentchar\LWR@multimedia@percenterror\fi%
```

Paragraph handling:

```
123     \LWR@stoppars%
```

Record the desired size.

```
124     \setlength{\LWR@multimedia@width}{0pt}%
125     \setlength{\LWR@multimedia@height}{0pt}%
126     \setkeys*{LWR@multimedia}{#1}%
```

If a known A/V type, create an HTML5 <video> or <audio>.

```
127     \IfEndWith{#3}{.mp4}{\LWR@multimedia@AV{#2}{#3}{video}{video/mp4}}{%
```

```
128     \IfEndWith{#3}{.MP4}{\LWR@multimedia@AV{#2}{#3}{video}{video/mp4}}{%
129     \IfEndWith{#3}{.mp3}{\LWR@multimedia@AV{#2}{#3}{audio}{audio/mpeg}}{%
130     \IfEndWith{#3}{.MP3}{\LWR@multimedia@AV{#2}{#3}{audio}{audio/mpeg}}{%
```

If an arbitrary URL, embed it.

```
131     \IfBeginWith{#3}{http}{\LWR@multimedia@embed{#2}{#3}{}}{%
132     \IfBeginWith{#3}{HTTP}{\LWR@multimedia@embed{#2}{#3}{}}{%
133     \IfBeginWith{#3}{ftp}{\LWR@multimedia@embed{#2}{#3}{}}{%
134     \IfBeginWith{#3}{FTP}{\LWR@multimedia@embed{#2}{#3}{}}{%
```

If unknown, create a link to it.

```
135         \LWR@href{#3}{#2}% unknown format
136     }}}}}}}%
```

Paragraph handling:

```
137     \LWR@startpars%
138     \endgroup%
139 }
```

Catcodes which may apper in a URL.

```
140 \newrobustcmd*\LWR@multimedia}{%
141     \begingroup%
142     \LWR@linkmediacatcodes%
143     \LWR@multimediab%
144 }
```

File 592 **`lwarp-common-mathjax-letters.sty`**

§ 704   Package   # common-mathjax-letters

lwarp-common-mathjax-letters Common code used by a number of packages to generate Greek math characters
(*Pkg*) for MATHJAX.

for HTML output: 1 \ProvidesPackage{lwarp-common-mathjax-letters}[2020/08/10]

\LWR@mathjax@addletter   * {⟨2: capitalize name?⟩} {⟨3: prefix⟩} {⟨4: postfix⟩} {⟨5: name⟩} {⟨6: unicode⟩}

Star to italicize the result, used when the unicode character does not exist.

```
2 \begin{warpMathJax}
3
4 \NewDocumentCommand{\LWR@mathjax@addletter}{s m m m m m}{
5     \IfBooleanTF{#2}%
6         {\edef\LWR@tempone{\LWRtexttitlecase{#5}}}%
7         {\edef\LWR@tempone{#5}}%
8     \xdef\LWR@customizedMathJax{%
9         \LWR@customizedMathJax%
10        \LWRbackslash(%
11        \LWRbackslash def\LWRbackslash%
12        #3% prefix
13        \LWR@tempone%name
14        #4% postfix
15        \LWRleftbrace%
16    }%
17    \IfBooleanTF{#1}{%
18        \xdef\LWR@customizedMathJax{%
```

```
19              \LWR@customizedMathJax%
20              \LWRbackslash mathit\LWRleftbrace%
21              \LWRbackslash unicode\LWRleftbrace x#6\LWRrightbrace%
22              \LWRrightbrace%
23          }%
24      }{%
25          \xdef\LWR@customizedMathJax{%
26              \LWR@customizedMathJax%
27              \LWRbackslash unicode\LWRleftbrace x#6\LWRrightbrace%
28          }%
29      }%
30      \xdef\LWR@customizedMathJax{%
31          \LWR@customizedMathJax%
32          \LWRrightbrace\LWRbackslash)\par%
33      }%
34 }
```

       * {⟨*2: prefix*⟩} {⟨*3: postfix*⟩}

\LWR@mathjax@addgreek@l@up   Star to capitalize the macro names.

Adds \CustomizeMathjax expressions to define a set of macros for Greek letters, lowercase upright.

```
35 \NewDocumentCommand{\LWR@mathjax@addgreek@l@up}{s m m}{
36      \LWR@mathjax@addletter{#1}{#2}{#3}{alpha}{03B1}
37      \LWR@mathjax@addletter{#1}{#2}{#3}{beta}{03B2}
38      \LWR@mathjax@addletter{#1}{#2}{#3}{varbeta}{03D0}
39      \LWR@mathjax@addletter{#1}{#2}{#3}{gamma}{03B3}
40      \LWR@mathjax@addletter{#1}{#2}{#3}{digamma}{03DD}
41      \LWR@mathjax@addletter{#1}{#2}{#3}{delta}{03B4}
42      \LWR@mathjax@addletter{#1}{#2}{#3}{epsilon}{03F5}
43      \LWR@mathjax@addletter{#1}{#2}{#3}{varepsilon}{03B5}
44      \LWR@mathjax@addletter{#1}{#2}{#3}{zeta}{03B6}
45      \LWR@mathjax@addletter{#1}{#2}{#3}{eta}{03B7}
46      \LWR@mathjax@addletter{#1}{#2}{#3}{theta}{03B8}
47      \LWR@mathjax@addletter{#1}{#2}{#3}{vartheta}{03D1}
48      \LWR@mathjax@addletter{#1}{#2}{#3}{iota}{03B9}
49      \LWR@mathjax@addletter{#1}{#2}{#3}{kappa}{03BA}
50      \LWR@mathjax@addletter{#1}{#2}{#3}{varkappa}{03F0}
51      \LWR@mathjax@addletter{#1}{#2}{#3}{lambda}{03BB}
52      \LWR@mathjax@addletter{#1}{#2}{#3}{mu}{03BC}
53      \LWR@mathjax@addletter{#1}{#2}{#3}{nu}{03BD}
54      \LWR@mathjax@addletter{#1}{#2}{#3}{xi}{03BE}
55      \LWR@mathjax@addletter{#1}{#2}{#3}{omicron}{03BF}
56      \LWR@mathjax@addletter{#1}{#2}{#3}{pi}{03C0}
57      \LWR@mathjax@addletter{#1}{#2}{#3}{varpi}{03D6}
58      \LWR@mathjax@addletter{#1}{#2}{#3}{rho}{03C1}
59      \LWR@mathjax@addletter{#1}{#2}{#3}{varrho}{03F1}
60      \LWR@mathjax@addletter{#1}{#2}{#3}{sigma}{03C3}
61      \LWR@mathjax@addletter{#1}{#2}{#3}{varsigma}{03C2}
62      \LWR@mathjax@addletter{#1}{#2}{#3}{tau}{03C4}
63      \LWR@mathjax@addletter{#1}{#2}{#3}{upsilon}{03C5}
64      \LWR@mathjax@addletter{#1}{#2}{#3}{phi}{03D5}
65      \LWR@mathjax@addletter{#1}{#2}{#3}{varphi}{03C6}
66      \LWR@mathjax@addletter{#1}{#2}{#3}{chi}{03C7}
67      \LWR@mathjax@addletter{#1}{#2}{#3}{psi}{03C8}
68      \LWR@mathjax@addletter{#1}{#2}{#3}{omega}{03C9}
69 }
```

* {⟨*2: prefix*⟩} {⟨*3: postfix*⟩}

`\LWR@mathjax@addgreek@u@up`

Star to capitalize the macro names.

Adds `\CustomizeMathjax` expressions to define a set of macros for Greek letters, uppercase upright.

```
70 \NewDocumentCommand{\LWR@mathjax@addgreek@u@up}{s m m}{
71     \LWR@mathjax@addletter{#1}{#2}{#3}{alpha}{0391}
72     \LWR@mathjax@addletter{#1}{#2}{#3}{beta}{0392}
73     \LWR@mathjax@addletter{#1}{#2}{#3}{gamma}{0393}
74     \LWR@mathjax@addletter{#1}{#2}{#3}{digamma}{03DC}
75     \LWR@mathjax@addletter{#1}{#2}{#3}{delta}{0394}
76     \LWR@mathjax@addletter{#1}{#2}{#3}{epsilon}{0395}
77     \LWR@mathjax@addletter{#1}{#2}{#3}{zeta}{0396}
78     \LWR@mathjax@addletter{#1}{#2}{#3}{eta}{0397}
79     \LWR@mathjax@addletter{#1}{#2}{#3}{theta}{0398}
80     \LWR@mathjax@addletter{#1}{#2}{#3}{vartheta}{03F4}
81     \LWR@mathjax@addletter{#1}{#2}{#3}{iota}{0399}
82     \LWR@mathjax@addletter{#1}{#2}{#3}{kappa}{039A}
83     \LWR@mathjax@addletter{#1}{#2}{#3}{lambda}{039B}
84     \LWR@mathjax@addletter{#1}{#2}{#3}{mu}{039C}
85     \LWR@mathjax@addletter{#1}{#2}{#3}{nu}{039D}
86     \LWR@mathjax@addletter{#1}{#2}{#3}{xi}{039E}
87     \LWR@mathjax@addletter{#1}{#2}{#3}{omicron}{039F}
88     \LWR@mathjax@addletter{#1}{#2}{#3}{pi}{03A0}
89     \LWR@mathjax@addletter{#1}{#2}{#3}{varpi}{03D6}
90     \LWR@mathjax@addletter{#1}{#2}{#3}{rho}{03A1}
91     \LWR@mathjax@addletter{#1}{#2}{#3}{sigma}{03A3}
92     \LWR@mathjax@addletter{#1}{#2}{#3}{tau}{03A4}
93     \LWR@mathjax@addletter{#1}{#2}{#3}{upsilon}{03A5}
94     \LWR@mathjax@addletter{#1}{#2}{#3}{phi}{03A6}
95     \LWR@mathjax@addletter{#1}{#2}{#3}{chi}{03A7}
96     \LWR@mathjax@addletter{#1}{#2}{#3}{psi}{03A8}
97     \LWR@mathjax@addletter{#1}{#2}{#3}{omega}{03A9}
98 }
```

* {⟨*2: prefix*⟩} {⟨*3: postfix*⟩}

`\LWR@mathjax@addgreek@l@it`

Star to capitalize the macro names.

Adds `\CustomizeMathjax` expressions to define a set of macros for Greek letters, lowercase italic.

```
99 \NewDocumentCommand{\LWR@mathjax@addgreek@l@it}{s m m}{
100     \LWR@mathjax@addletter{#1}{#2}{#3}{alpha}{1D6FC}
101     \LWR@mathjax@addletter{#1}{#2}{#3}{beta}{1D6FD}
102     \LWR@mathjax@addletter{#1}{#2}{#3}{varbeta}{03D0}
103     \LWR@mathjax@addletter{#1}{#2}{#3}{gamma}{1D6FE}
104     \LWR@mathjax@addletter*{#1}{#2}{#3}{digamma}{03DD}
105     \LWR@mathjax@addletter{#1}{#2}{#3}{delta}{1D6FF}
106     \LWR@mathjax@addletter{#1}{#2}{#3}{epsilon}{1D716}
107     \LWR@mathjax@addletter{#1}{#2}{#3}{varepsilon}{1D700}
108     \LWR@mathjax@addletter{#1}{#2}{#3}{zeta}{1D701}
109     \LWR@mathjax@addletter{#1}{#2}{#3}{eta}{1D702}
110     \LWR@mathjax@addletter{#1}{#2}{#3}{theta}{1D703}
111     \LWR@mathjax@addletter{#1}{#2}{#3}{vartheta}{1D717}
112     \LWR@mathjax@addletter{#1}{#2}{#3}{iota}{1D704}
113     \LWR@mathjax@addletter{#1}{#2}{#3}{kappa}{1D705}
114     \LWR@mathjax@addletter{#1}{#2}{#3}{varkappa}{1D718}
115     \LWR@mathjax@addletter{#1}{#2}{#3}{lambda}{1D706}
116     \LWR@mathjax@addletter{#1}{#2}{#3}{mu}{1D707}
```

```
117        \LWR@mathjax@addletter{#1}{#2}{#3}{nu}{1D708}
118        \LWR@mathjax@addletter{#1}{#2}{#3}{xi}{1D709}
119        \LWR@mathjax@addletter{#1}{#2}{#3}{omicron}{1D70A}
120        \LWR@mathjax@addletter{#1}{#2}{#3}{pi}{1D70B}
121        \LWR@mathjax@addletter{#1}{#2}{#3}{varpi}{1D71B}
122        \LWR@mathjax@addletter{#1}{#2}{#3}{rho}{1D70C}
123        \LWR@mathjax@addletter{#1}{#2}{#3}{varrho}{1D71A}
124        \LWR@mathjax@addletter{#1}{#2}{#3}{sigma}{1D70E}
125        \LWR@mathjax@addletter{#1}{#2}{#3}{varsigma}{1D70D}
126        \LWR@mathjax@addletter{#1}{#2}{#3}{tau}{1D70F}
127        \LWR@mathjax@addletter{#1}{#2}{#3}{upsilon}{1D710}
128        \LWR@mathjax@addletter{#1}{#2}{#3}{phi}{1D719}
129        \LWR@mathjax@addletter{#1}{#2}{#3}{varphi}{1D711}
130        \LWR@mathjax@addletter{#1}{#2}{#3}{chi}{1D712}
131        \LWR@mathjax@addletter{#1}{#2}{#3}{psi}{1D713}
132        \LWR@mathjax@addletter{#1}{#2}{#3}{omega}{1D714}
133 }
```

\LWR@mathjax@addgreek@u@it

* {⟨*2: prefix*⟩} {⟨*3: postfix*⟩}

Star to capitalize the macro names.

Adds \CustomizeMathjax expressions to define a set of macros for Greek letters, uppercase italic.

```
134 \NewDocumentCommand{\LWR@mathjax@addgreek@u@it}{s m m}{
135        \LWR@mathjax@addletter{#1}{#2}{#3}{alpha}{1D6E2}
136        \LWR@mathjax@addletter{#1}{#2}{#3}{beta}{1D6E3}
137        \LWR@mathjax@addletter{#1}{#2}{#3}{gamma}{1D6E4}
138        \LWR@mathjax@addletter*{#1}{#2}{#3}{digamma}{03DC}
139        \LWR@mathjax@addletter{#1}{#2}{#3}{delta}{1D6E5}
140        \LWR@mathjax@addletter{#1}{#2}{#3}{epsilon}{1D6E6}
141        \LWR@mathjax@addletter{#1}{#2}{#3}{zeta}{1D6E7}
142        \LWR@mathjax@addletter{#1}{#2}{#3}{eta}{1D6E8}
143        \LWR@mathjax@addletter{#1}{#2}{#3}{theta}{1D6E9}
144        \LWR@mathjax@addletter{#1}{#2}{#3}{vartheta}{1D6F3}
145        \LWR@mathjax@addletter{#1}{#2}{#3}{iota}{1D6EA}
146        \LWR@mathjax@addletter{#1}{#2}{#3}{kappa}{1D6EB}
147        \LWR@mathjax@addletter{#1}{#2}{#3}{lambda}{1D6EC}
148        \LWR@mathjax@addletter{#1}{#2}{#3}{mu}{1D6ED}
149        \LWR@mathjax@addletter{#1}{#2}{#3}{nu}{1D6EE}
150        \LWR@mathjax@addletter{#1}{#2}{#3}{xi}{1D6EF}
151        \LWR@mathjax@addletter{#1}{#2}{#3}{omicron}{1D6F0}
152        \LWR@mathjax@addletter{#1}{#2}{#3}{pi}{1D6F1}
153        \LWR@mathjax@addletter{#1}{#2}{#3}{rho}{1D6F2}
154        \LWR@mathjax@addletter{#1}{#2}{#3}{sigma}{1D6F4}
155        \LWR@mathjax@addletter{#1}{#2}{#3}{tau}{1D6F5}
156        \LWR@mathjax@addletter{#1}{#2}{#3}{upsilon}{1D6F6}
157        \LWR@mathjax@addletter{#1}{#2}{#3}{phi}{1D6F7}
158        \LWR@mathjax@addletter{#1}{#2}{#3}{chi}{1D6F8}
159        \LWR@mathjax@addletter{#1}{#2}{#3}{psi}{1D6F9}
160        \LWR@mathjax@addletter{#1}{#2}{#3}{omega}{1D6FA}
161 }
```

\LWR@mathjax@addgreek@l@bfit

* {⟨*2: prefix*⟩} {⟨*3: postfix*⟩}

Star to capitalize the macro names.

Adds \CustomizeMathjax expressions to define a set of macros for Greek letters, lowercase boldface italic.

```
162 \NewDocumentCommand{\LWR@mathjax@addgreek@l@bfit}{s m m}{
```

```
163        \LWR@mathjax@addletter{#1}{#2}{#3}{alpha}{1D736}
164        \LWR@mathjax@addletter{#1}{#2}{#3}{beta}{1D737}
165        \LWR@mathjax@addletter{#1}{#2}{#3}{varbeta}{03D0}
166        \LWR@mathjax@addletter{#1}{#2}{#3}{gamma}{1D738}
167        \LWR@mathjax@addletter*{#1}{#2}{#3}{digamma}{03DD}
168        \LWR@mathjax@addletter{#1}{#2}{#3}{delta}{1D739}
169        \LWR@mathjax@addletter{#1}{#2}{#3}{epsilon}{1D750}
170        \LWR@mathjax@addletter{#1}{#2}{#3}{varepsilon}{1D73A}
171        \LWR@mathjax@addletter{#1}{#2}{#3}{zeta}{1D73B}
172        \LWR@mathjax@addletter{#1}{#2}{#3}{eta}{1D73C}
173        \LWR@mathjax@addletter{#1}{#2}{#3}{theta}{1D73D}
174        \LWR@mathjax@addletter{#1}{#2}{#3}{vartheta}{1D751}
175        \LWR@mathjax@addletter{#1}{#2}{#3}{iota}{1D73E}
176        \LWR@mathjax@addletter{#1}{#2}{#3}{kappa}{1D73F}
177        \LWR@mathjax@addletter{#1}{#2}{#3}{varkappa}{1D752}
178        \LWR@mathjax@addletter{#1}{#2}{#3}{lambda}{1D740}
179        \LWR@mathjax@addletter{#1}{#2}{#3}{mu}{1D741}
180        \LWR@mathjax@addletter{#1}{#2}{#3}{nu}{1D742}
181        \LWR@mathjax@addletter{#1}{#2}{#3}{xi}{1D743}
182        \LWR@mathjax@addletter{#1}{#2}{#3}{omicron}{1D744}
183        \LWR@mathjax@addletter{#1}{#2}{#3}{pi}{1D745}
184        \LWR@mathjax@addletter{#1}{#2}{#3}{varpi}{1D755}
185        \LWR@mathjax@addletter{#1}{#2}{#3}{rho}{1D746}
186        \LWR@mathjax@addletter{#1}{#2}{#3}{varrho}{1D754}
187        \LWR@mathjax@addletter{#1}{#2}{#3}{sigma}{1D748}
188        \LWR@mathjax@addletter{#1}{#2}{#3}{varsigma}{1D747}
189        \LWR@mathjax@addletter{#1}{#2}{#3}{tau}{1D749}
190        \LWR@mathjax@addletter{#1}{#2}{#3}{upsilon}{1D74A}
191        \LWR@mathjax@addletter{#1}{#2}{#3}{phi}{1D753}
192        \LWR@mathjax@addletter{#1}{#2}{#3}{varphi}{1D74B}
193        \LWR@mathjax@addletter{#1}{#2}{#3}{chi}{1D74C}
194        \LWR@mathjax@addletter{#1}{#2}{#3}{psi}{1D74D}
195        \LWR@mathjax@addletter{#1}{#2}{#3}{omega}{1D74E}
196 }
```

* {⟨*2: prefix*⟩} {⟨*3: postfix*⟩}

\LWR@mathjax@addgreek@u@bfit   Star to capitalize the macro names.

Adds \CustomizeMathjax expressions to define a set of macros for Greek letters, uppercase boldface italic.

```
197 \NewDocumentCommand{\LWR@mathjax@addgreek@u@bfit}{s m m}{
198        \LWR@mathjax@addletter{#1}{#2}{#3}{alpha}{1D71C}
199        \LWR@mathjax@addletter{#1}{#2}{#3}{beta}{1D71D}
200        \LWR@mathjax@addletter{#1}{#2}{#3}{gamma}{1D71E}
201        \LWR@mathjax@addletter*{#1}{#2}{#3}{digamma}{03DC}
202        \LWR@mathjax@addletter{#1}{#2}{#3}{delta}{1D71F}
203        \LWR@mathjax@addletter{#1}{#2}{#3}{epsilon}{1D720}
204        \LWR@mathjax@addletter{#1}{#2}{#3}{zeta}{1D721}
205        \LWR@mathjax@addletter{#1}{#2}{#3}{eta}{1D722}
206        \LWR@mathjax@addletter{#1}{#2}{#3}{theta}{1D723}
207        \LWR@mathjax@addletter{#1}{#2}{#3}{vartheta}{1D72D}
208        \LWR@mathjax@addletter{#1}{#2}{#3}{iota}{1D724}
209        \LWR@mathjax@addletter{#1}{#2}{#3}{kappa}{1D725}
210        \LWR@mathjax@addletter{#1}{#2}{#3}{lambda}{1D726}
211        \LWR@mathjax@addletter{#1}{#2}{#3}{mu}{1D727}
212        \LWR@mathjax@addletter{#1}{#2}{#3}{nu}{1D728}
213        \LWR@mathjax@addletter{#1}{#2}{#3}{xi}{1D729}
214        \LWR@mathjax@addletter{#1}{#2}{#3}{omicron}{1D72A}
215        \LWR@mathjax@addletter{#1}{#2}{#3}{pi}{1D72B}
```

```
216    \LWR@mathjax@addletter{#1}{#2}{#3}{rho}{1D72C}
217    \LWR@mathjax@addletter{#1}{#2}{#3}{sigma}{1D72E}
218    \LWR@mathjax@addletter{#1}{#2}{#3}{tau}{1D72F}
219    \LWR@mathjax@addletter{#1}{#2}{#3}{upsilon}{1D730}
220    \LWR@mathjax@addletter{#1}{#2}{#3}{phi}{1D731}
221    \LWR@mathjax@addletter{#1}{#2}{#3}{chi}{1D732}
222    \LWR@mathjax@addletter{#1}{#2}{#3}{psi}{1D733}
223    \LWR@mathjax@addletter{#1}{#2}{#3}{omega}{1D734}
224 }
```

\LWR@mathjax@addgreek@u@bfup is not needed.

\LWR@mathjax@addgreek@u@bfup

* {⟨*2: prefix*⟩} {⟨*3: postfix*⟩}

Star to capitalize the macro names.

Adds \CustomizeMathjax expressions to define a set of macros for Greek letters, uppercase boldface upright.

```
225 \NewDocumentCommand{\LWR@mathjax@addgreek@u@bfup}{s m m}{
226    \LWR@mathjax@addletter{#1}{#2}{#3}{alpha}{1D6A8}
227    \LWR@mathjax@addletter{#1}{#2}{#3}{beta}{1D6A9}
228    \LWR@mathjax@addletter{#1}{#2}{#3}{gamma}{1D6AA}
229    \LWR@mathjax@addletter*{#1}{#2}{#3}{digamma}{03DC}
230    \LWR@mathjax@addletter{#1}{#2}{#3}{delta}{1D6AB}
231    \LWR@mathjax@addletter{#1}{#2}{#3}{epsilon}{1D6AC}
232    \LWR@mathjax@addletter{#1}{#2}{#3}{zeta}{1D6AD}
233    \LWR@mathjax@addletter{#1}{#2}{#3}{eta}{1D6AE}
234    \LWR@mathjax@addletter{#1}{#2}{#3}{theta}{1D6AF}
235    \LWR@mathjax@addletter{#1}{#2}{#3}{vartheta}{1D6B9}
236    \LWR@mathjax@addletter{#1}{#2}{#3}{iota}{1D6B0}
237    \LWR@mathjax@addletter{#1}{#2}{#3}{kappa}{1D6B1}
238    \LWR@mathjax@addletter{#1}{#2}{#3}{lambda}{1D6B2}
239    \LWR@mathjax@addletter{#1}{#2}{#3}{mu}{1D6B3}
240    \LWR@mathjax@addletter{#1}{#2}{#3}{nu}{1D6B4}
241    \LWR@mathjax@addletter{#1}{#2}{#3}{xi}{1D6B5}
242    \LWR@mathjax@addletter{#1}{#2}{#3}{omicron}{1D6B6}
243    \LWR@mathjax@addletter{#1}{#2}{#3}{pi}{1D6B7}
244    \LWR@mathjax@addletter{#1}{#2}{#3}{rho}{1D6B8}
245    \LWR@mathjax@addletter{#1}{#2}{#3}{sigma}{1D6BA}
246    \LWR@mathjax@addletter{#1}{#2}{#3}{tau}{1D6BB}
247    \LWR@mathjax@addletter{#1}{#2}{#3}{upsilon}{1D6BC}
248    \LWR@mathjax@addletter{#1}{#2}{#3}{phi}{1D6BD}
249    \LWR@mathjax@addletter{#1}{#2}{#3}{chi}{1D6BE}
250    \LWR@mathjax@addletter{#1}{#2}{#3}{psi}{1D6BF}
251    \LWR@mathjax@addletter{#1}{#2}{#3}{omega}{1D6C0}
252 }
```

{⟨*prefix*⟩}

\LWR@mathjax@addlatin@u@bfit

Adds \CustomizeMathjax expressions to define a set of macros for bold-face italic Latin letters, uppercase and lowercase.

```
253 \NewDocumentCommand{\LWR@mathjax@addlatin@u@bfit}{m}{
254    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{A}{1D468}
255    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{B}{1D469}
256    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{C}{1D46A}
257    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{D}{1D46B}
258    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{E}{1D46C}
259    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{F}{1D46D}
260    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{G}{1D46E}
```

```
261    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{H}{1D46F}
262    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{I}{1D470}
263    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{J}{1D471}
264    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{K}{1D472}
265    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{L}{1D473}
266    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{M}{1D474}
267    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{N}{1D475}
268    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{O}{1D476}
269    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{P}{1D477}
270    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{Q}{1D478}
271    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{R}{1D479}
272    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{S}{1D47A}
273    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{T}{1D47B}
274    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{U}{1D47C}
275    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{V}{1D47D}
276    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{W}{1D47E}
277    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{X}{1D47F}
278    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{Y}{1D480}
279    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{Z}{1D481}
280 }
```

{⟨*prefix*⟩}

\LWR@mathjax@addlatin@l@bfit Adds \CustomizeMathjax expressions to define a set of macros for bold-face italic Latin letters, uppercase and lowercase.

```
281 \NewDocumentCommand{\LWR@mathjax@addlatin@l@bfit}{m}{
282    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{a}{1D482}
283    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{b}{1D483}
284    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{c}{1D484}
285    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{d}{1D485}
286    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{e}{1D486}
287    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{f}{1D487}
288    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{g}{1D488}
289    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{h}{1D489}
290    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{i}{1D48A}
291    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{j}{1D48B}
292    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{k}{1D48C}
293    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{l}{1D48D}
294    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{m}{1D48E}
295    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{n}{1D48F}
296    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{o}{1D490}
297    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{p}{1D491}
298    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{q}{1D492}
299    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{r}{1D493}
300    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{s}{1D494}
301    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{t}{1D495}
302    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{u}{1D496}
303    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{v}{1D497}
304    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{w}{1D498}
305    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{x}{1D499}
306    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{y}{1D49A}
307    \LWR@mathjax@addletter{\BooleanFalse}{#1}{}{z}{1D49B}
308 }
```

```
309 \end{warpMathJax}
```

File 593   **`lwarp-common-mathjax-newpxtxmath.sty`**

§ 705   Package   # common-mathjax-newpxtxmath

(*Emulates or patches code by* Michael Sharpe.)

lwarp-common-mathjax-newpxtxmath   Common code used by newpxmath, newtxmath, and newtxsf for MathJax.
(*Pkg*)
for HTML output: 1 \ProvidesPackage{lwarp-common-mathjax-newpxtxmath}[2020/09/20]

For MathJax:

```
2 \LWR@origRequirePackage{lwarp-common-mathjax-nonunicode}
3 \LWR@origRequirePackage{lwarp-common-mathjax-overlaysymbols}
4
5 \begin{warpMathJax}
6 \CustomizeMathJax{\newcommand{\fAlt}{f}}
7 \CustomizeMathJax{\newcommand{\rhoAlt}{\rho}}
8
9 \CustomizeMathJax{\newcommand{\imathscr}{\mathord{\mathscr{i}}}}
10 \CustomizeMathJax{\newcommand{\jmathscr}{\mathord{\mathscr{j}}}}
```

lwarp_mathjax.txt adds \left/\right support for delimiters.

```
11 \CustomizeMathJax{\let\llbracket\lBrack}
12 \CustomizeMathJax{\let\rrbracket\rBrack}
13
14 \CustomizeMathJax{\let\smlbrace\{}
15 \CustomizeMathJax{\let\smrbrace\}}
16 \CustomizeMathJax{\newcommand{\Perp}{\mathrel{\unicode{x02AEB}}}}
17 \CustomizeMathJax{\newcommand{\nPerp}{\mathrel{\not{\!\unicode{x02AEB}}}}}
18 \CustomizeMathJax{\newcommand{\Zbar}{\mathord{\unicode{x01B5}}}}
19 \CustomizeMathJax{\newcommand{\Angstrom}{\mathord{\unicode{x212B}}}}
20 \CustomizeMathJax{\newcommand{\Euler}{\mathord{\unicode{x2107}}}}
21 \CustomizeMathJax{\newcommand{\transp}{\mathord{\unicode{xFF34}}}}
22 \CustomizeMathJax{\newcommand{\hermtransp}{\mathord{\unicode{xFF28}}}}
23 \CustomizeMathJax{\let\htransp=\hermtransp}
24 \CustomizeMathJax{\newcommand{\circledplus}{\mathbin{\unicode{x2295}}}}
25 \CustomizeMathJax{\newcommand{\circledminus}{\mathbin{\unicode{x2296}}}}
26 \CustomizeMathJax{\newcommand{\circledtimes}{\mathbin{\unicode{x2297}}}}
27 \CustomizeMathJax{\newcommand{\circledslash}{\mathbin{\unicode{x2298}}}}
28 %
29 \CustomizeMathJax{\newcommand{\circleddot}{\mathbin{\unicode{x2299}}}}
30 \CustomizeMathJax{\let\overgroup\overparen}
31 \CustomizeMathJax{\let\overgroupra\overrightarrow}
32 \CustomizeMathJax{\let\undergroup\underparen}
33 \CustomizeMathJax{\let\undergroupla\underleftarrow}
34 \CustomizeMathJax{\newcommand{\widering}[1]{%
35     \stackrel{\unicode{x2218}}{\overgroup{#1}}%
36 }}
37 \CustomizeMathJax{\let\widearc\overparen}
38 \CustomizeMathJax{\let\wideOarc\overrightarrow}
39 \CustomizeMathJax{\newcommand{\LWRvvstar}[2]{{\overrightarrow{#1}_{#2}}}}
40 \CustomizeMathJax{\newcommand{\vv}{\ifstar\LWRvvstar\overrightarrow}}
```

```
41 %
42 \CustomizeMathJax{\let\smallintsl\smallint}
43 \CustomizeMathJax{\newcommand{\smalliintsl}{\mathop{\unicode{x222C}}\limits}}
44 \CustomizeMathJax{\newcommand{\smalliiintsl}{\mathop{\unicode{x222D}}\limits}}
45 \CustomizeMathJax{\newcommand{\smalliiiintsl}{\mathop{\unicode{x2A0C}}\limits}}
46 \CustomizeMathJax{\newcommand{\smallointsl}{\mathop{\unicode{x222E}}\limits}}
47 \CustomizeMathJax{\newcommand{\smalloiintsl}{\mathop{\unicode{x222F}}\limits}}
48 \CustomizeMathJax{\newcommand{\smalloiiintsl}{\mathop{\unicode{x2230}}\limits}}
49 \CustomizeMathJax{\newcommand{\smallvarointclockwisesl}{%
50     \mathop{\unicode{x2232}}\limits%
51 }}
52 \CustomizeMathJax{\newcommand{\smallointctrclockwisesl}{%
53     \mathop{\unicode{x2233}}\limits%
54 }}
55 \CustomizeMathJax{\newcommand{\smallsumintsl}{\mathop{\unicode{x2A0B}}\limits}}
56 \CustomizeMathJax{\newcommand{\smallfintsl}{\mathop{\unicode{x2A0F}}\limits}}
57 \CustomizeMathJax{\newcommand{\smallsqintsl}{\mathop{\unicode{x2A16}}\limits}}
58 %
59 \CustomizeMathJax{\let\smallintup\smallint}
60 \CustomizeMathJax{\newcommand{\smalliintup}{\mathop{\unicode{x222C}}\limits}}
61 \CustomizeMathJax{\newcommand{\smalliiintup}{\mathop{\unicode{x222D}}\limits}}
62 \CustomizeMathJax{\newcommand{\smalliiiintup}{\mathop{\unicode{x2A0C}}\limits}}
63 \CustomizeMathJax{\newcommand{\smallointup}{\mathop{\unicode{x222E}}\limits}}
64 \CustomizeMathJax{\newcommand{\smalloiintup}{\mathop{\unicode{x222F}}\limits}}
65 \CustomizeMathJax{\newcommand{\smalloiiintup}{\mathop{\unicode{x2230}}\limits}}
66 \CustomizeMathJax{\newcommand{\smallvarointclockwiseup}{%
67     \mathop{\unicode{x2232}}\limits%
68 }}
69 \CustomizeMathJax{\newcommand{\smallointctrclockwiseup}{%
70     \mathop{\unicode{x2233}}\limits%
71 }}
72 \CustomizeMathJax{\newcommand{\smallsumintup}{\mathop{\unicode{x2A0B}}\limits}}
73 \CustomizeMathJax{\newcommand{\smallfintup}{\mathop{\unicode{x2A0F}}\limits}}
74 \CustomizeMathJax{\newcommand{\smallsqintup}{\mathop{\unicode{x2A16}}\limits}}
75 %
76 \CustomizeMathJax{\newcommand{\iint}{\mathop{\unicode{x222C}}\limits}}
77 \CustomizeMathJax{\newcommand{\iiint}{\mathop{\unicode{x222D}}\limits}}
78 \CustomizeMathJax{\newcommand{\iiiint}{\mathop{\unicode{x2A0C}}\limits}}
79 \CustomizeMathJax{\newcommand{\oiint}{\mathop{\unicode{x222F}}\limits}}
80 \CustomizeMathJax{\newcommand{\oiiint}{\mathop{\unicode{x2230}}\limits}}
81 \CustomizeMathJax{\newcommand{\varointclockwise}{\mathop{\unicode{x2232}}\limits}}
82 \CustomizeMathJax{\newcommand{\ointctrclockwise}{\mathop{\unicode{x2233}}\limits}}
83 \CustomizeMathJax{\newcommand{\sumint}{\mathop{\unicode{x2A0B}}\limits}}
84 \CustomizeMathJax{\newcommand{\fint}{\mathop{\unicode{x2A0F}}\limits}}
85 \CustomizeMathJax{\newcommand{\sqint}{\mathop{\unicode{x2A16}}\limits}}
86 %
87 \CustomizeMathJax{\let\intsl\int}
88 \CustomizeMathJax{\newcommand{\iintsl}{\mathop{\unicode{x222C}}\limits}}
89 \CustomizeMathJax{\newcommand{\iiintsl}{\mathop{\unicode{x222D}}\limits}}
90 \CustomizeMathJax{\newcommand{\iiiintsl}{\mathop{\unicode{x2A0C}}\limits}}
91 \CustomizeMathJax{\let\ointsl\oint}
92 \CustomizeMathJax{\newcommand{\oiintsl}{\mathop{\unicode{x222F}}\limits}}
93 \CustomizeMathJax{\newcommand{\oiiintsl}{\mathop{\unicode{x2230}}\limits}}
94 \CustomizeMathJax{\newcommand{\varointclockwisesl}{\mathop{\unicode{x2232}}\limits}}
95 \CustomizeMathJax{\newcommand{\ointctrclockwisesl}{\mathop{\unicode{x2233}}\limits}}
96 \CustomizeMathJax{\newcommand{\sumintsl}{\mathop{\unicode{x2A0B}}\limits}}
97 \CustomizeMathJax{\newcommand{\fintsl}{\mathop{\unicode{x2A0F}}\limits}}
98 \CustomizeMathJax{\newcommand{\sqintsl}{\mathop{\unicode{x2A16}}\limits}}
99 %
100 \CustomizeMathJax{\let\intup\int}
```

```
101 \CustomizeMathJax{\newcommand{\iintup}{\mathop{\unicode{x222C}}\limits}}
102 \CustomizeMathJax{\newcommand{\iiintup}{\mathop{\unicode{x222D}}\limits}}
103 \CustomizeMathJax{\newcommand{\iiiintup}{\mathop{\unicode{x2A0C}}\limits}}
104 \CustomizeMathJax{\let\ointup\oint}
105 \CustomizeMathJax{\newcommand{\oiintup}{\mathop{\unicode{x222F}}\limits}}
106 \CustomizeMathJax{\newcommand{\oiiintup}{\mathop{\unicode{x2230}}\limits}}
107 \CustomizeMathJax{\newcommand{\varointclockwiseup}{%
108     \mathop{\unicode{x2232}}\limits%
109 }}
110 \CustomizeMathJax{\newcommand{\ointctrclockwiseup}{%
111     \mathop{\unicode{x2233}}\limits%
112 }}
113 \CustomizeMathJax{\newcommand{\sumintup}{\mathop{\unicode{x2A0B}}\limits}}
114 \CustomizeMathJax{\newcommand{\fintup}{\mathop{\unicode{x2A0F}}\limits}}
115 \CustomizeMathJax{\newcommand{\sqintup}{\mathop{\unicode{x2A16}}\limits}}
116 %
117 \CustomizeMathJax{\newcommand{\bigcupdot}{\mathop{\unicode{x2A03}}}}
118 \CustomizeMathJax{\newcommand{\bigcupplus}{\mathop{\unicode{x2A04}}}}
119 \CustomizeMathJax{\newcommand{\bigsqcap}{\mathop{\unicode{x2A05}}}}
120 %


121 %
122 \CustomizeMathJax{\newcommand{\bigtimes}{\mathop{\unicode{x2A09}}}}
123 \CustomizeMathJax{\let\varprod\bigtimes}
124 %


125 \CustomizeMathJax{\newcommand{\mappedfrom}{\mathrel{\unicode{x021A4}}}}
126 \CustomizeMathJax{\let\mappedfromchar\mappedfrom}
127 \CustomizeMathJax{\newcommand{\mapsfrom}{\mathrel{\unicode{x021A4}}}}
128 \CustomizeMathJax{\newcommand{\longmappedfrom}{\mathrel{\unicode{x027FB}}}}
129 %
130 \CustomizeMathJax{\newcommand{\Mapsto}{\mathrel{\unicode{x02907}}}}
131 \CustomizeMathJax{\let\Mapstochar\Mapsto}
132 \CustomizeMathJax{\newcommand{\Longmapsto}{\mathrel{\unicode{x027FE}}}}
133 \CustomizeMathJax{\newcommand{\Mappedfrom}{\mathrel{\unicode{x02906}}}}
134 \CustomizeMathJax{\let\Mappedfromchar\Mappedfrom}
135 \CustomizeMathJax{\newcommand{\Mapsfrom}{\mathrel{\unicode{x02906}}}}
136 \CustomizeMathJax{\newcommand{\Longmappedfrom}{\mathrel{\unicode{x27FD}}}}
137 %


138 \CustomizeMathJax{\newcommand{\medcirc}{\mathbin{\unicode{x025CB}}}}
139 \CustomizeMathJax{\newcommand{\medbullet}{\mathbin{\unicode{x025CF}}}}
140 \CustomizeMathJax{\newcommand{\varparallel}{\mathrel{\unicode{x02AFD}}}}
141 \CustomizeMathJax{\newcommand{\varparallelinv}{\mathrel{\unicode{x244A}}}}
142 \CustomizeMathJax{\newcommand{\nvarparallel}{%
143     \mathrel{\LWRoverlaysymbols{-}{\unicode{x02AFD}}}%
144 }}
145 \CustomizeMathJax{\newcommand{\nvarparallelinv}{%
146     \mathrel{\LWRoverlaysymbols{-}{\unicode{x244A}}}%
147 }}
148 %


149 \CustomizeMathJax{\newcommand{\coloneq}{\mathrel{\unicode{x02254}}}}
150 \CustomizeMathJax{\newcommand{\eqcolon}{\mathrel{\unicode{x02255}}}}
151 %
152 \CustomizeMathJax{\newcommand{\VDash}{\mathrel{\unicode{x22AB}}}}
```

```
153 %
154 \CustomizeMathJax{\newcommand{\preceqq}{\mathrel{\unicode{x02AB3}}}}
155 \CustomizeMathJax{\newcommand{\succeqq}{\mathrel{\unicode{x02AB4}}}}
156 %
157
158 \CustomizeMathJax{\newcommand{\nprecsim}{%
159     \mathrel{\LWRoverlaysymbols{/}{\unicode{x0227E}}}%
160 }}
161 \CustomizeMathJax{\newcommand{\nsuccsim}{%
162     \mathrel{\LWRoverlaysymbols{/}{\unicode{x0227F}}}%
163 }}
164 \CustomizeMathJax{\newcommand{\nlesssim}{\mathrel{\unicode{x02274}}}}
165 \CustomizeMathJax{\newcommand{\ngtrsim}{\mathrel{\unicode{x02275}}}}
166 %

167 \CustomizeMathJax{\newcommand{\nsubset}{\mathrel{\unicode{x02284}}}}
168 \CustomizeMathJax{\newcommand{\nsupset}{\mathrel{\unicode{x02285}}}}
169 \CustomizeMathJax{\newcommand{\notni}{\mathrel{\unicode{x220C}}}}
170 \CustomizeMathJax{\let\notowns\notni}
171 %
172 \CustomizeMathJax{\newcommand{\nlessapprox}{%
173     \mathrel{\LWRoverlaysymbols{/}{\unicode{x02A85}}}%
174 }}
175 \CustomizeMathJax{\newcommand{\ngtrapprox}{%
176     \mathrel{\LWRoverlaysymbols{/}{\unicode{x02A86}}}%
177 }}
178 %
179 \CustomizeMathJax{\newcommand{\npreccurlyeq}{%
180     \mathrel{\LWRoverlaysymbols{/}{\unicode{x0227C}}}%
181 }}
182 \CustomizeMathJax{\newcommand{\nsucccurlyeq}{%
183     \mathrel{\LWRoverlaysymbols{/}{\unicode{x0227D}}}%
184 }}
185 \CustomizeMathJax{\newcommand{\ngtrless}{\mathrel{\unicode{x02279}}}}
186 \CustomizeMathJax{\newcommand{\nlessgtr}{\mathrel{\unicode{x2278}}}}
187 \CustomizeMathJax{\newcommand{\nbumpeq}{%
188     \mathrel{\LWRoverlaysymbols{/}{\unicode{x0224F}}}%
189 }}
190 \CustomizeMathJax{\newcommand{\nBumpeq}{%
191     \mathrel{\LWRoverlaysymbols{/}{\unicode{x0224E}}}%
192 }}
193 %
194 \CustomizeMathJax{\newcommand{\nbacksim}{%
195     \mathrel{\LWRoverlaysymbols{/}{\unicode{x0223D}}}%
196 }}
197 \CustomizeMathJax{\newcommand{\nbacksimeq}{%
198     \mathrel{\LWRoverlaysymbols{/}{\unicode{x022CD}}}%
199 }}
200 \CustomizeMathJax{\newcommand{\nasymp}{\mathrel{\unicode{x226D}}}}
201 \CustomizeMathJax{\newcommand{\nequiv}{\mathrel{\unicode{x2262}}}}
202 \CustomizeMathJax{\newcommand{\napprox}{\mathrel{\unicode{x2249}}}}
203 %
204 \CustomizeMathJax{\newcommand{\nll}{%
205     \mathrel{\LWRoverlaysymbols{/}{\unicode{x0226A}}}%
206 }}
207 \CustomizeMathJax{\newcommand{\ngg}{%
208     \mathrel{\LWRoverlaysymbols{/}{\unicode{x0226B}}}%
209 }}
210 \CustomizeMathJax{\newcommand{\nthickapprox}{%
211     \mathrel{\LWRoverlaysymbols{/}{{\mathbf{\unicode{x02248}}}}}%
```

```
212 }}
213 \CustomizeMathJax{\newcommand{\napproxeq}{%
214     \mathrel{\LWRoverlaysymbols{/}{\unicode{x0224A}}}}%
215 }}
216 \CustomizeMathJax{\newcommand{\nprecapprox}{%
217     \mathrel{\LWRoverlaysymbols{/}{\unicode{x02AB7}}}}%
218 }}
219 \CustomizeMathJax{\newcommand{\nsuccapprox}{%
220     \mathrel{\LWRoverlaysymbols{/}{\unicode{x02AB8}}}}%
221 }}
222 \CustomizeMathJax{\newcommand{\npreceqq}{%
223     \mathrel{\LWRoverlaysymbols{/}{\unicode{x02AB3}}}}%
224 }}
225 \CustomizeMathJax{\newcommand{\nsucceqq}{%
226     \mathrel{\LWRoverlaysymbols{/}{\unicode{x02AB4}}}}%
227 }}
228 \CustomizeMathJax{\newcommand{\nsimeq}{\mathrel{\unicode{x02244}}}}
229 %
230 \CustomizeMathJax{\newcommand{\nSubset}{%
231     \mathrel{\LWRoverlaysymbols{/}{\unicode{x022D0}}}}%
232 }}
233 \CustomizeMathJax{\newcommand{\nSupset}{%
234     \mathrel{\LWRoverlaysymbols{/}{\unicode{x022D1}}}}%
235 }}
236 \CustomizeMathJax{\newcommand{\nsqsubseteq}{\mathrel{\unicode{x022E2}}}}
237 \CustomizeMathJax{\newcommand{\nsqsupseteq}{\mathrel{\unicode{x022E3}}}}
238 %
239 \CustomizeMathJax{\newcommand{\coloneqq}{\mathrel{\unicode{x02254}}}}
240 \CustomizeMathJax{\newcommand{\eqqcolon}{\mathrel{\unicode{x02255}}}}
241 \CustomizeMathJax{\newcommand{\Coloneqq}{\mathrel{\unicode{x02A74}}}}
242 \CustomizeMathJax{\newcommand{\Coloneq}{\mathrel{\unicode{x2237}-}}}
243 \CustomizeMathJax{\newcommand{\Eqcolon}{\mathrel{-\unicode{x2237}}}}
244 %
245 \CustomizeMathJax{\newcommand{\lvec}[1]{%
246     \mathord{\overset{\unicode{x02190}}{#1}}%
247 }}
248 \CustomizeMathJax{\newcommand{\lrvec}[1]{%
249     \mathord{\overset{\unicode{x2194}}{#1}}%
250 }}
251 \CustomizeMathJax{\newcommand{\harpoonacc}[1]{%
252     \mathord{\overset{\unicode{x021C0}}{#1}}%
253 }}
254 \CustomizeMathJax{\newcommand{\lharpoonacc}[1]{%
255     \mathord{\overset{\unicode{x021BC}}{#1}}%
256 }}
257 \CustomizeMathJax{\newcommand{\lrharpoonacc}[1]{%
258     \mathord{\overset{\unicode{x0294E}}{#1}}%
259 }}
260 \CustomizeMathJax{\newcommand{\barbar}[1]{\mathord{\overset{=}{#1}}}}
261 \CustomizeMathJax{\newcommand{\bartilde}[1]{\mathord{\overset{\simeq}{#1}}}}
262 \CustomizeMathJax{\newcommand{\barhat}[1]{\mathord{\hat{\bar{#1}}}}}
263 \CustomizeMathJax{\newcommand{\tildebar}[1]{\mathord{\overset{\eqsim}{#1}}}}
264 \CustomizeMathJax{\newcommand{\tildetilde}[1]{\mathord{\overset{\approx}{#1}}}}
265 \CustomizeMathJax{\newcommand{\tildehat}[1]{\mathord{\hat{\tilde{#1}}}}}
266 \CustomizeMathJax{\newcommand{\hatbar}[1]{\mathord{\bar{\hat{#1}}}}}
267 \CustomizeMathJax{\newcommand{\hattilde}[1]{\mathord{\tilde{\hat{#1}}}}}
268 \CustomizeMathJax{\newcommand{\hathat}[1]{\mathord{\hat{\hat{#1}}}}}
269
270 \CustomizeMathJax{\newcommand{\cdotB}{\mathord{\boldsymbol{\cdot}}}}
271 \CustomizeMathJax{\newcommand{\cdotBB}{\mathord{\unicode{x2022}}}}
```

```
272 \CustomizeMathJax{\newcommand{\circS}{\boldsymbol{\circ}}}
273 \CustomizeMathJax{\newcommand{\bulletSSS}{\bullet}}
274 \CustomizeMathJax{\newcommand{\bulletSS}{\mathord{\unicode{x025CF}}}}
275 \CustomizeMathJax{\newcommand{\bulletS}{\mathord{\unicode{x02B24}}}}
276 \CustomizeMathJax{\newcommand{\primeS}{\prime}}
277
278 \CustomizeMathJax{\newcommand{\invamp}{\mathbin{\unicode{x0214B}}}}
```

lwarp_mathjax.txt adds \left/\right support for delimiters.

```
279 \CustomizeMathJax{\newcommand{\Lbag}{\mathopen{\large\unicode{x027C5}}}}
280 \CustomizeMathJax{\newcommand{\Rbag}{\mathclose{\large\unicode{x027C6}}}}
281 \CustomizeMathJax{\newcommand{\circledless}{\mathrel{\unicode{x029C0}}}}
282 \CustomizeMathJax{\newcommand{\circledgtr}{\mathrel{\unicode{x029C1}}}}
283 \CustomizeMathJax{\newcommand{\circledbslash}{\mathbin{\unicode{x029B8}}}}


284 \CustomizeMathJax{\newcommand{\lJoin}{\mathrel{\unicode{x22C9}}}}
285 \CustomizeMathJax{\newcommand{\rJoin}{\mathrel{\unicode{x22CA}}}}
286 \CustomizeMathJax{\newcommand{\lrJoin}{\mathrel{\unicode{x2A1D}}}}
287
288 \CustomizeMathJax{\newcommand{\lrtimes}{\mathrel{\unicode{x2A1D}}}}
289 \CustomizeMathJax{\newcommand{\Diamondblack}{\mathord{\unicode{x025C6}}}}
290 \CustomizeMathJax{\newcommand{\nplus}{%
291     \mathrel{\LWRoverlaysymbols{+}{\unicode{x02229}}}%
292 }}
293 \CustomizeMathJax{\newcommand{\nsqsubset}{%
294     \mathrel{\LWRoverlaysymbols{/}{\unicode{x0228F}}}%
295 }}
296 \CustomizeMathJax{\newcommand{\nsqsupset}{%
297     \mathrel{\LWRoverlaysymbols{/}{\unicode{x02290}}}%
298 }}
299 \CustomizeMathJax{\newcommand{\dasharrow}{\mathrel{\unicode{x021E2}}}}
300 \CustomizeMathJax{\newcommand{\leftsquigarrow}{\mathrel{\unicode{x021DC}}}}
301 \CustomizeMathJax{\newcommand{\ntwoheadrightarrow}{\mathrel{\unicode{x02900}}}}
302 \CustomizeMathJax{\newcommand{\ntwoheadleftarrow}{\mathrel{\unicode{x02B34}}}}
303 \CustomizeMathJax{\newcommand{\boxast}{\mathbin{\unicode{x029C6}}}}
304 \CustomizeMathJax{\newcommand{\boxbslash}{\mathbin{\unicode{x29C5}}}}
305 \CustomizeMathJax{\newcommand{\boxbar}{\mathbin{\unicode{x025EB}}}}
306 \CustomizeMathJax{\newcommand{\boxslash}{\mathbin{\unicode{x029C4}}}}
307
308 \CustomizeMathJax{\newcommand{\varclubsuit}{\mathord{\unicode{x02667}}}}
309 \CustomizeMathJax{\newcommand{\vardiamondsuit}{\mathord{\unicode{x02666}}}}
310 \CustomizeMathJax{\newcommand{\varheartsuit}{\mathord{\unicode{x02665}}}}
311 \CustomizeMathJax{\newcommand{\varspadesuit}{\mathord{\unicode{x02664}}}}
312
313 \CustomizeMathJax{\newcommand{\Nearrow}{\mathrel{\unicode{x021D7}}}}
314 \CustomizeMathJax{\newcommand{\Searrow}{\mathrel{\unicode{x021D8}}}}
315 \CustomizeMathJax{\newcommand{\Nwarrow}{\mathrel{\unicode{x021D6}}}}
316 \CustomizeMathJax{\newcommand{\Swarrow}{\mathrel{\unicode{x021D9}}}}
317 \CustomizeMathJax{\newcommand{\Top}{\mathord{\unicode{x02AEA}}}}
318 \CustomizeMathJax{\newcommand{\Bot}{\mathord{\unicode{x02AEB}}}}
319
320 \CustomizeMathJax{\newcommand{\leadstoext}{\mathrel{\unicode{xFF5E}}}}
321
322 \CustomizeMathJax{\newcommand{\sqcupplus}{%
323     \mathbin{\LWRoverlaysymbols{+}{\unicode{x02294}}}%
324 }}
325 \CustomizeMathJax{\newcommand{\sqcapplus}{%
326     \mathbin{\LWRoverlaysymbols{+}{\unicode{x02293}}}%
```

```
327 }}
328
329 \CustomizeMathJax{\newcommand{\dlb}{\mathopen{\unicode{x027E6}}}}
330 \CustomizeMathJax{\newcommand{\drb}{\mathopen{\unicode{x027E7}}}}
331
332 \CustomizeMathJax{\newcommand{\varg}{g}}
333 \CustomizeMathJax{\newcommand{\vary}{y}}
334 \CustomizeMathJax{\newcommand{\varv}{v}}
335 \CustomizeMathJax{\newcommand{\varw}{w}}
336
337 \CustomizeMathJax{\newcommand{\nexistsAlt}{\mathord{\unicode{x02204}}}}
338 \CustomizeMathJax{\newcommand{\existsAlt}{\mathord{\unicode{x02203}}}}
339 \CustomizeMathJax{\newcommand{\forallAlt}{\mathord{\unicode{x02200}}}}
340 \CustomizeMathJax{\newcommand{\emptysetAlt}{\mathord{\unicode{x02205}}}}
341
342 \CustomizeMathJax{\newcommand{\uppartial}{%
343     \mathord{\unicode{x02202}}%
344 }}% not upright
345
346 \CustomizeMathJax{\let\varmathbb\mathbb}
347 \CustomizeMathJax{\let\vmathbb\mathbb}
348 \CustomizeMathJax{\let\vvmathbb\mathbb}
349
350 \CustomizeMathJax{\let\smallprod\prod}
351 \CustomizeMathJax{\let\smallsum\sum}
352 \CustomizeMathJax{\let\smallcoprod\coprod}
353
354 \CustomizeMathJax{\newcommand{\openbox}{\mathord{\unicode{x25FD}}}}
355 \CustomizeMathJax{\let\textsquare\openbox}


356 \CustomizeMathJax{\let\varemptyset\emptyset}
357 %
358 % for newpxmath:
359 \CustomizeMathJax{\newcommand{\mathsterling}{\mathord{\unicode{x000A3}}}}
360 \CustomizeMathJax{\newcommand{\mathcent}{\mathord{\unicode{x000A2}}}}
361
362 \end{warpMathJax}
```

File 594   `lwarp-common-mathjax-nonunicode.sty`

§ 706   Package   # common-mathjax-nonunicode

(*Emulates or patches code by* Daniel Flipo, Michael Sharpe.)

lwarp-common-mathjax-nonunicode Common code used by newpxmath, newtxmath, newtxsf, kpfonts-otf for Math-
(*Pkg*)   Jax. These are symbols not found in Unicode.

Factored from lwarp-common-mathjax-newpxtxmath.

for HTML output: 1 \ProvidesPackage{lwarp-common-mathjax-nonunicode}[2020/09/20]

For MathJax:

```
2 \LWR@origRequirePackage{lwarp-common-mathjax-overlaysymbols}
3
4 \begin{warpMathJax}
```

```
5 \CustomizeMathJax{\newcommand{\mmapsto}{\mathrel{\unicode{x021A6}}}}
6 \CustomizeMathJax{\let\mmapstochar\mmapsto}
7 \CustomizeMathJax{\newcommand{\longmmapsto}{\mathrel{\unicode{x021A6}}}}
8 \CustomizeMathJax{\newcommand{\mmappedfrom}{\mathrel{\unicode{x021A4}}}}
9 \CustomizeMathJax{\let\mmappedfromchar\mmappedfrom}
10 \CustomizeMathJax{\newcommand{\longmmappedfrom}{\mathrel{\unicode{x021A4}}}}
11 \CustomizeMathJax{\let\mmapsfrom\mmappedfrom}% from kpfonts-otf
12 \CustomizeMathJax{\let\longmmapsfrom\longmmappedfrom}% from kpfonts-otf
13
14 \CustomizeMathJax{\newcommand{\Mmapsto}{\mathrel{\unicode{x02907}}}}
15 \CustomizeMathJax{\let\Mmapstochar\Mmapsto}
16 \CustomizeMathJax{\newcommand{\Longmmapsto}{\mathrel{\unicode{x027FE}}}}
17 \CustomizeMathJax{\newcommand{\Mmappedfrom}{\mathrel{\unicode{x02906}}}}
18 \CustomizeMathJax{\let\Mmappedfromchar\Mmappedfrom}
19 \CustomizeMathJax{\newcommand{\Longmmappedfrom}{\mathrel{\unicode{x027FD}}}}
20 \CustomizeMathJax{\let\Mmapsfrom\Mmappedfrom}% from kpfonts-otf
21 \CustomizeMathJax{\let\Longmmapsfrom\Longmmappedfrom}% from kpfonts-otf
22 %
23 \CustomizeMathJax{\newcommand{\boxright}{%
24     \mathrel{\unicode{x025A1}\!\unicode{x02192}}%
25 }}
26 \CustomizeMathJax{\newcommand{\boxleft}{%
27     \mathrel{\unicode{x02190}\!\unicode{x025A1}}%
28 }}
29 \CustomizeMathJax{\newcommand{\boxdotright}{%
30     \mathrel{\unicode{x022A1}\!\unicode{x02192}}%
31 }}
32 \CustomizeMathJax{\newcommand{\boxdotleft}{%
33     \mathrel{\unicode{x02190}\!\unicode{x022A1}}%
34 }}
35
36 \CustomizeMathJax{\newcommand{\Diamondright}{%
37     \mathrel{\unicode{x025C7}\!\unicode{x02192}}%
38 }}
39 \CustomizeMathJax{\newcommand{\Diamondleft}{%
40     \mathrel{\unicode{x02190}\!\unicode{x025C7}}%
41 }}
42 \CustomizeMathJax{\newcommand{\Diamonddotright}{%
43     \mathrel{\unicode{x027D0}\!\unicode{x02192}}%
44 }}
45 \CustomizeMathJax{\newcommand{\Diamonddotleft}{%
46     \mathrel{\unicode{x02190}\!\unicode{x027D0}}%
47 }}
48
49 \CustomizeMathJax{\newcommand{\boxRight}{%
50     \mathrel{\unicode{x025A1}\!\unicode{x021D2}}%
51 }}
52 \CustomizeMathJax{\newcommand{\boxLeft}{%
53     \mathrel{\unicode{x021D0}\!\unicode{x025A1}}%
54 }}
55 \CustomizeMathJax{\newcommand{\boxdotRight}{%
56     \mathrel{\unicode{x022A1}\!\unicode{x021D2}}%
57 }}
58 \CustomizeMathJax{\newcommand{\boxdotLeft}{%
59     \mathrel{\unicode{x021D0}\!\unicode{x022A1}}%
60 }}
61
62 \CustomizeMathJax{\newcommand{\DiamondRight}{%
63     \mathrel{\unicode{x025C7}\!\unicode{x021D2}}%
64 }}
```

```
65 \CustomizeMathJax{\newcommand{\DiamondLeft}{%
66     \mathrel{\unicode{x021D0}\!\unicode{x025C7}}%
67 }}
68 \CustomizeMathJax{\newcommand{\DiamonddotRight}{%
69     \mathrel{\unicode{x027D0}\!\unicode{x021D2}}%
70 }}
71 \CustomizeMathJax{\newcommand{\DiamonddotLeft}{%
72     \mathrel{\unicode{x021D0}\!\unicode{x027D0}}%
73 }}
74 \CustomizeMathJax{\newcommand{\Diamonddot}{\mathrel{\unicode{x027D0}}}}
75
76 \CustomizeMathJax{\newcommand{\circleright}{%
77     \mathrel{\unicode{x025CB}\!\unicode{x02192}}%
78 }}
79 \CustomizeMathJax{\newcommand{\circleleft}{%
80     \mathrel{\unicode{x02190}\!\unicode{x025CB}}%
81 }}
82 \CustomizeMathJax{\newcommand{\circledotright}{%
83     \mathrel{\unicode{x02299}\!\unicode{x02192}}%
84 }}
85 \CustomizeMathJax{\newcommand{\circledotleft}{%
86     \mathrel{\unicode{x02190}\!\unicode{x02299}}%
87 }}
88 \CustomizeMathJax{\let\circleddotright\circledotright}
89 \CustomizeMathJax{\let\circleddotleft\circledotleft}
90
91 \CustomizeMathJax{\newcommand{\multimapinv}{\mathrel{\unicode{x027DC}}}}
92 \CustomizeMathJax{\newcommand{\multimapboth}{\mathrel{\unicode{x029DF}}}}
93 \CustomizeMathJax{\newcommand{\multimapdot}{{\mathrel{-\!\bullet}}}}
94 \CustomizeMathJax{\newcommand{\multimapdotinv}{\mathrel{\bullet\!-}}}
95 \CustomizeMathJax{\newcommand{\multimapdotboth}{%
96     \mathrel{{\bullet\!\!-\!\!\bullet}}%
97 }}
98 \CustomizeMathJax{\newcommand{\multimapdotbothA}{\mathrel{\unicode{x022B6}}}}
99 \CustomizeMathJax{\newcommand{\multimapdotbothB}{\mathrel{\unicode{x22B7}}}}
100
101 \CustomizeMathJax{\newcommand{\multimapbothvert}{%
102     \mathrel{\overset{\unicode{x025CB}}{\underset{\unicode{x025CB}}{|}}}%
103 }}
104 \CustomizeMathJax{\newcommand{\multimapdotbothvert}{%
105     \mathrel{\overset{\unicode{x025CF}}{\underset{\unicode{x025CF}}{|}}}%
106 }}
107 \CustomizeMathJax{\newcommand{\multimapdotbothBvert}{% bug in kpfonts-otf
108     \mathrel{\overset{\unicode{x025CF}}{\underset{\unicode{x025CB}}{|}}}%
109 }}
110 \CustomizeMathJax{\newcommand{\multimapdotbothAvert}{% bug in kpfonts-otf
111     \mathrel{\overset{\unicode{x025CB}}{\underset{\unicode{x025CF}}{|}}}%
112 }}
113
114 \CustomizeMathJax{\newcommand{\bignplus}{%
115     \mathop{\LWRoverlaysymbols{\unicode{xFF0B}}{\unicode{x22C2}}}%
116 }}
117 \CustomizeMathJax{\let\bigcapplus\bignplus}
118 \CustomizeMathJax{\let\capplus\bignplus}% from kpfonts-otf
119
120 \CustomizeMathJax{\newcommand{\bigsqcapplus}{%
121     \mathop{\LWRoverlaysymbols{\unicode{xFF0B}}{\unicode{x2A05}}}%
122 }}
123 \CustomizeMathJax{\let\sqcapplus\bigsqcapplus}% from kpfonts-otf
124
```

```
125 \CustomizeMathJax{\newcommand{\bigsqcupplus}{%
126     \mathop{\LWRoverlaysymbols{\unicode{xFF0B}}{\unicode{x2A06}}}}
127 }}
128 \CustomizeMathJax{\let\sqcupplus\bigsqcupplus}% from kpfonts-otf
129
130 \CustomizeMathJax{\newcommand{\parallelslant}{\mathrel{\unicode{x02AFD}}}}
131 \CustomizeMathJax{\newcommand{\parallelbackslant}{%
132     \mathrel{\unicode{x0005C}}\!\!\unicode{x0005C}}%
133 }}
134
135 \CustomizeMathJax{\newcommand{\Eqqcolon}{\mathrel{=\!\unicode{x2237}}}}
136 \CustomizeMathJax{\let\eqqColon\Eqqcolon}% for kpfonts-otf
137 \CustomizeMathJax{\newcommand{\dashColon}{\mathrel{-\unicode{x2237}}}}
138 \CustomizeMathJax{\newcommand{\Colondash}{\mathrel{\unicode{x2237}-}}}
139
140 \CustomizeMathJax{\newcommand{\colonapprox}{\mathrel{:\approx}}}
141 \CustomizeMathJax{\newcommand{\colonsim}{\mathrel{:\sim}}}
142 \CustomizeMathJax{\newcommand{\Colonapprox}{%
143     \mathrel{\unicode{x2237}\!\approx}%
144 }}
145 \CustomizeMathJax{\newcommand{\Colonsim}{\mathrel{\unicode{x2237}\!\sim}}}
146
147 \CustomizeMathJax{\newcommand{\strictif}{%
148     \mathrel{\unicode{x0297D}}%
149 }}% right fish tail
150 \CustomizeMathJax{\newcommand{\strictfi}{%
151     \mathrel{\unicode{x0297C}}%
152 }}% left fish tail
153 \CustomizeMathJax{\newcommand{\strictiff}{%
154     \mathrel{\unicode{x0297C}\!\!\unicode{x0297D}}%
155 }}% left/right fish tails
156
157 \CustomizeMathJax{\newcommand{\circledwedge}{%
158     \mathbin{\LWRoverlaysymbols{\unicode{x025EF}}{\unicode{x02227}}}%
159 }}
160 \CustomizeMathJax{\newcommand{\circledvee}{%
161     \mathbin{\LWRoverlaysymbols{\unicode{x025EF}}{\unicode{0x02228}}}%
162 }}
163 \CustomizeMathJax{\newcommand{\circledbar}{\mathbin{\unicode{x029B6}}}}
164
165 \CustomizeMathJax{\newcommand{\openJoin}{%
166     \mathrel{\unicode{x2AA4}}%
167 }}% overlapping ><
168 \CustomizeMathJax{\newcommand{\opentimes}{%
169     \mathrel{\unicode{x2AA4}}%
170 }}% overlapping ><
171
172 \CustomizeMathJax{\newcommand{\VvDash}{\mathrel{\unicode{x22AA}}}}
173
174 \CustomizeMathJax{\newcommand{\lambdabar}{%
175     \mathord{\LWRoverlaysymbols{\raise{.5ex}{-}}{\lambda}}%
176 }}
177
178 \CustomizeMathJax{\newcommand{\lambdaslash}{\mathord{\unicode{x019B}}}}
179
180 \CustomizeMathJax{\newcommand{\Wr}{%
181     \mathbin{\unicode{x02240}\!\unicode{x02240}}%
182 }}
183
184 \CustomizeMathJax{\newcommand{\dashleftrightarrow}{%
```

```
185         \mathrel{\unicode{x021E0}\!\unicode{x021E2}}%
186 }}
187 \CustomizeMathJax{\let\leftrightdasharrow\dashleftrightarrow}% for kpfonts-otf
188
189 \end{warpMathJax}
```

File 595   **lwarp-common-mathjax-overlaysymbols.sty**

§ 707   Package   **common-mathjax-overlaysymbols**

lwarp-common-mathjax-overlaysymbols   Common code used by a number of packages to overlay two symbols for MATHJAX.
(*Pkg*)

for HTML output: 1 \ProvidesPackage{lwarp-common-mathjax-overlaysymbols}[2020/08/17]

\LWRoverlaysymbols   {⟨*symbol*⟩} {⟨*symbol*⟩}

Overlays one symbol over another.

```
2 \begin{warpMathJax}
3
4 \CustomizeMathJax{\newcommand{\LWRoverlaysymbols}[2]{%
5     \mathord{%
6         \smash{%
7             \mathop{#2\strut}%
8             \limits^{\smash{\lower3ex{#1}}}}%
9         }%
10        \strut%
11    }%
12 }}
13
14 \end{warpMathJax}
```

Change History

§ 708 **Chg Hist**

For the most recent changes, see page .

1337

1341

1353

1354

1355

Index of Objects

This is an index of macros, environments, booleans, counters, lengths, packages, classes, options, keys, files, and various other programming objects. Each is listed by itself, and also by category. In some cases, they are further subdivided by [class].

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition.

1379

1382

1383

1395

General Index

This is an index of instructions and concepts. Look here when wondering how to do something, and check the Troubleshooting Index when something goes wrong.

Troubleshooting Index

This index is a sorted reference of problems and solutions. In order to make it easier to locate a solution, the same issue may be addressed by more than one entry.

Entries starting with page 207 are often duplicates of entries with lower page numbers, as the same warning may occur within the user manual and again within the source code.

1409

Index of Indexes