Lwarp

LaTeX   HTML5

The lwarp package

LaTeX to HTML

v0.910 — 2023/01/03

© 2016–2023 Brian Dunn
bd@BDTechConcepts.com

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.

Lwarp is still in development. Changes are likely.

License:

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 .

v0.910: 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
packages
- Added tests for additional incompatible 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.

⚠
- 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$_{\exists}$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
docs   and <span>s. See section 7.8.

- 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 XƎTEX 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 5.17.

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 5.19.
- 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 410.
- 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 XƎTEX 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 35.
- 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 8.6.14.

- 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 8.6.20.

- 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 X∃LATEX 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 84.3.

- 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 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.

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, 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
LibreOffice (*Prog*) the 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 ad-
FrameMaker (*Prog*) vancing 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 compli-
cated 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 77.

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

   ⚠ Note: .txt suffix!   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 8.6.12.

[7]Also see section 8.6.15 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 8.6.12.

[11]Also see section 8.6.15 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 Ti*k*Z   lwarp uses the same mechanism for `picture` and Ti*k*Z 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 Ti*k*Z 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 Ti*k*Z 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 Ti*k*Z 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 X∃LATEX or LuaLATEX

X∃LATEX 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. X∃LATEX 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 glossary

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

5.11.1 gloss package

See section 8.6.11.

5.11.2 glossaries package

To process the glossary for the print version:

> Enter ⟹ **lwarpmk printglossary**

⚠ (If **makeglossaries** is not found, see section 8.6.12.)

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 8.6.12 to set options for processing glossaries.

5.12   Cleaning auxiliary files

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

>   Enter ⇒   **lwarpmk clean**

5.13   Cleaning auxiliary and output files

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

>   Enter ⇒   **lwarpmk cleanall**

5.14   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**

5.15   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.

5.16   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]**

5.17   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**

5.18   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.

5.19   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 29, 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 paren math \(x+y\) instead of dollar math $x+y$. (Dollar math works, but appears in the filename.) Or, use a short name for the TOC entry without the math, or use \texorpdfstring from the hyperref package:

   \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 8.6.15.

   (g) If using many indexes, glossaries, .aux files, etc., see section 8.6.15 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: LʿTẸX 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 8.6.21.
>
> **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 laateximages. 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 reference 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
Computer Modern
⚠ *pdflatex*
⚠ DVI *latex*
cm-super (*Pkg*)

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

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}

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 if 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*, fontspec X∃LATEX and LuaLATEX users must use the fontspec package. Do NOT use fontenc!

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 X∃LATEX 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*)

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

    (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 the dotless j (\j):

⚠ dotless 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 X⅂LATEX 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.

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

⚠ JETBRAIN MONO
⚠ HTML corrupted

```
\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 8.6.14 for indexing, and section 8.6.12 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
mathjax (*Opt*)   section 8.7.
Default: mathsvg

latexmk (*Opt*)   **latexmk:**   Tells *lwarpmk* to use *latexmk* to recompile the document several times
Default: false   if 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
Default: image-   objects 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 8.6.15) and glossaries (section 8.6.12):

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 8.6.20.

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 8.6.21.

xindyLanguage (*Opt*)
Default: english

xindyLanguage: If using an index or glossary, see section 29.

xindyCodepage (*Opt*)
Default: utf8

xindyCodepage: If using an index, see section 29.

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 8.6.22.

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 8.6.15.

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, xindyLanuage, 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*)   **HTMLIndexCmd:**   Sets the shell commands executed by **lwarpmk htmlindex**. If not
Default: <automatic>   specified, will be set by the selection of makeindex or xindy. May be used to
                          specify the creation of multiple indexes. See section 8.6.15.

⚠ 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, xindyLanuage, 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 worth-
                          while to use a shell script, then refer to that script with HTMLIndexCmd.

LatexmkIndexCmd (*Opt*)   **LatexmkIndexCmd:**   Sets the shell commands executed by *latexmk*.   Unlike
Default: <automatic>   PrintIndexCmd and HTMLIndexCmd, LatexmkIndexCmd does not include any
                          filenames, which will be provided instead by *latexmk*. See section 8.6.15.

                          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 ei-
                          ther of the two settings show above. lwarp's makeindexStyle, xindyStyle,
                          xindyLanuage, and xindyCodepage options will be used if specified. Unlike
                          PrintIndexCmd and HTMLIndexCmd, *latexmk* uses either of the single-line set-
                          tings 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*)   **IndexRef:**   Describes how to display the index entries for HTML output. Possible
Default: cref   values are ref, nameref, refnameref, cref, crefnameref, autoref, or a text
                          string such as (link) or (*) for each index entry reference. (Adding paren-
                          theses 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*) **GlossaryCmd:** Sets the shell command executed by **lwarpmk printglossary** and
Default: **makeglossaries** **lwarpmk htmlglossary**. The print or HTML glossary filename is appended to this command. See section 8.6.12.

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

pdftotextEnc (*Opt*) **pdftotextEnc:** Used to specify the encoding used by *pdftotext* during the PDF-
Default: UTF-8 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 29.

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

---

warpprint (*Opt*) **warpprint** and **warpHTML:** Usually controlled by *lwarpmk*, and not set in the
warpHTML (*Opt*) 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*) **BaseJobname:** Not intended for the user. Used internally by lwarp when creating
Default: \jobname the *_html.tex file used to compile the HTML version. See section 29.

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
⚠ Placement!   for the correct placement of each!

⚠ 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
applied 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 description 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   **\sidetocname:**   Name of the sideTOC. Paragraphs are allowed. Redefine
Default: Contents   with \renewcommand.

FileDepth (*Ctr*)   **FileDepth:**   Sectioning depth of file splits. Defaults to -5, causing the entire
Default: -5   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*)   **CombineHigherDepths:**   Combine a higher section with its first lower sub-
Default: true   sections, down to the FileDepth. Defaults to true. Set to false to simu-
                      late the concept of a chapter opening on its own page, for example.

                      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 tu-
⚠ Inaccesible pages! torial 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*)   **FileSectionNames:**   If true, web page filenames are derived from a sanitized
Default: true   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
⚠ Unique filename!   that filenames are unique after begin 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.

\FilenameLimit   **\FilenameLimit:**   The maximum length of the filenames generated by
Default: 80   lwarp. ".html" is added to this length. Redefine with \renewcommand.

FootnoteDepth (*Ctr*)   **FootnoteDepth:**   Determines where to place pending footnotes. 3 places
Default: 3   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*)   **FixSmallCaps:**   Set true if SMALL CAPS are rendering in all caps ("SMALL
Default: false

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 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 pagess 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 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 each sectioning command which would cause a file split.

\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: (
\AltTextClose
Default:)

\AltTextOpen: Redefine with \renewcommand.

\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}
```

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:

---

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

```
...
\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, additional 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, contain-
project.css (*file*)   ing only things which are different from lwarp.css. The file project.css should
sample_project.css (*file*)   refer 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 /
Mac OS (*Prog*)   Linux is the default (collectively referred to as "Unix" in the configuration files),
Linux (*Prog*)   and MS-Windows is supported as well.
MS-Windows (*Prog*)
Windows (*Prog*)   If MS-Windows is not correctly detected,   use the lwarp option OSWindows.

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
is used to mark the end of the environment. Do not nest warpHTML inside itself.
⚠ nesting 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.*) 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
⚠ \end{warpsvg} on its own line. Do not nest warpsvg inside itself. warpsvg may be used inside
⚠ nesting warpHTML.

\LWR@formatted To define macros or environments which behave differently depending on print
or HTML output, see section 36.

7.11 Commands to be placed into the `warpprint` environment

Certain print-related commands should always be placed inside a warpprint envi-
ronment, 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 69.8.

\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 69.8). 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, is 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 example, 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 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.

7.15   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 convert &, <, and > to proper HTML entities.

verbatim   The various verbatim-related environments do not convert &, <, and >, so care must be taken to avoid accidentally including valid HTML code inside these environments. Adding a space on either side may be sufficient.

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
\hskip   by *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   Both \hrule and \vrule are ignored for HTML. To create a horizontal dividing
\vrule   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 **Filenames and URLs in lists or footnotes**

filename underscore Escape underscores in the filenames:

```
\item[\href{file\_name.pdf}{text}]
```

8.2.10   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

\UseMinipageWidths

\IgnoreMinipageWidths
   Inside a tabular or multicols environment, where the width depends on the browser window, \minipagefullwidth is effectively used by default for every minipage or \parbox inside the environment. \UseMinipageWidths may be used 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
For display tabular, fminipage is included. For print output, this is identical to minipage, except
minipages, and lists: 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 89 for a new environment fcolorminipage. Also see below regarding the "Misplaced alignment tab character &." error.

colored boxes and frames: **To create colored frames and boxes:** See section 674 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
⚠ *Misplaced alignment* tabular, you will have to use \StartDefiningTabulars before the start of
tab character & the macro which uses \fboxBlock and the tabular, and \StopDefiningTabulars afterwards. Also see the lwarp documentation for the fancybox package.

To frame equations: See section 259 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)   fancybox's documentation has an example FramedEqn environment which com-
framed equation example bines 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 con-
taining 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 documen-
tation'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.

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 sup-

⚠ math

ported. 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}, 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*⟩}

⚠ macros in section names   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}{}%
}
```

⚠ duplicate filename   Avoid duplicate file names. Section names at levels which result in HTML file splits must be unique. lwarp will generate an error if a duplicate HTML filename is generated. Use the optional TOC caption entry parameter for formatting. Remember to \protect LATEX commands which appear in section names and TOC captions.

⚠ math in section names   If using named HTML files, in section names use paren math \(x+y\) instead of dollar math $x+y$. (Dollar math works, but appears in the filename.) Or, use a short name for the TOC entry without the math, or use \texorpdfstring from the hyperref package:

```
\section{Some math \texorpdfstring{$1+2=3$}{three}}
```

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

⚠ LᴬTEX page numbersThe 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 503.

8.5.2   cleveref and varioref packages

cleveref (*Pkg*)

varioref (*Pkg*)

⚠ 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 734 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*)

url (*Pkg*)

lwarp emulates hyperref, including the creation of active hyperlinks, but does not require that hyperref be loaded by the document.

⚠ comments between arguments   Do not place a comment with a % character between arguments for \hyperref, 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   If using **memoir** class, with which lwarp preloads **footmisc**, the stable option must
⚠ **memoir**   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   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.

At present there is a bug such that paragraph closing tags are not present in footnotes when \VerbatimFootnotes are selected. The browser usually compensates.

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 X‌ELaTeX or LuaLaTeX instead of PDF LaTeX.

8.5.5 xr, xr-hyper, and xcite packages

See section 5.17.

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*)   If using the number option with file splits, be sure to place the table of contents
⚠ missing TOC   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*)   lwarp supports the native LATEX titling commands, and also supports the pack-
authblk (*Pkg*)   ages authblk and titling. If both are used, authblk should be loaded before titling.
package support

If using the titling package, additional titlepage fields for \published and \subtitle
⚠ load order   may be added by using \AddSubtitlePublished in the preamble. See section 69.8.
\published and \subtitle

8.6.5 tocloft package

titles (*Opt*) [tocloft]   If using tocloft with tocbibind, anonchap, fncychap, or other packages which
tocloft (*Pkg*)   change chapter title formatting, load tocloft with its titles option, which tells
tocloft (*Pkg*)   tocloft to use standard LATEX commands to create the titles, allowing other pack-
⚠ tocloft & other packages   ages to work with it.

8.6.6   appendix package

appendix (*Pkg*)   During HTML conversion, the option toc without the option page results in
⚠ incorrect TOC link   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*)   To place the endnotes in the TOC, use:
table of contents

```
\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   If using MATHJAX, see section 8.5.4 regarding the use of \endnotemark and
numbering   \endnotetext.

8.6.9   *BibTeX*

\etalchar   Displays a superscript "+" to indicate "and others".

⚠ Modify *.bib   When enough authors are cited for a source, BIBTEX may use the \etalchar com-
mand to display a math superscript with a + character to indicate "and oth-
ers". 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   xcite package

See section 5.17.

8.6.11   gloss package

gloss (*Pkg*)   To process the HTML glossary:
⚠ compiling

```
bibtex <projectname>_html.gls
```

8.6.12 glossaries package

glossaries (*Pkg*)
processing glossaries
GlossaryCmd (*Opt*)
Default: makeglossaries
printglossary (*Opt*) [lwarpmk]
htmlglossary (*Opt*) [lwarpmk]

lwarpmk has the commands **lwarpmk printglossary** and **lwarpmk htmlglossary**, which process the glossaries created by the glossaries package using that package's *makeglossaries* program.

The shell command to execute is set by the lwarp option GlossaryCmd, which defaults to **makeglossaries**. The print or HTML glossary filename is appended to this command.

⚠ *makeglossaries* not found

In some situations it may be required to modify the default command, such as to 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.

8.6.13 nomencl package

nomencl (*Pkg*)   To process the HTML nomenclature:

```
makeindex    <project>_html.nlo    -s    nomencl.ist    -o
<project>_html.nls
```

8.6.14 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 indexes 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 8.6.20 to 8.6.22.

link appearance   To control how the index links appear in the HTML output, see the IndexRef option in section 7.5, page 106.

source code   See section 79 for lwarp's core index and glossary code, section 340 for index, section 571 for splitidx, section 338 for imakeidx, section 628 for tocbibind, and section 695.17 for memoir's indexing patches.

8.6.15 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 8.6.20.

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 8.6.21.

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 8.6.22.

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.

8.6.16 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.

8.6.17 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 695.17 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.

8.6.18   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 8.6.20 to 8.6.22.

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.

8.6.19 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.

8.6.20   Using a custom *makeindex* style file

makeindex (*Prog*)   When using *makeindex*, *lwarpmk* uses the file lwarp.ist to process the index.
lwarp.ist (*file*)   This 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.

8.6.21   Using a custom *xindy* style file

xindy (*Prog*)   When using *xindy*, *lwarpmk* uses the file lwarp.xdy to process the index. This file
lwarp.xdy (*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.

8.6.22 Using a custom *xindex* style file

xindex (*Prog*)   To use a custom *xindex* style file:

1. Copy xindex-cfg.lua to a new filename such as xindex-projectname.lua.

⚠ filename   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.

8.6.23   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.

8.6.24   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 paren math \(x+y\) instead of
dollar math $x+y$. (Dollar math works, but appears in the filename.) Or, use a
short name for the TOC entry without the math, or use \texorpdfstring from the
hyperref package:

```
\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 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 680 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 $\mathcal{A}_{\mathcal{M}}\mathcal{S}$ 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
$\mathcal{A}_{\mathcal{M}}\mathcal{S}$ 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 equa-
   tion 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 out-
   put 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 ig-
   nored 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 399,
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 MᴀᴛʜJᴀх, 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 MᴀᴛʜJᴀх limitations

MᴀᴛʜJᴀх limitations   Limitations when using MᴀᴛʜJᴀх include:
MathJax (*Prog*)

⚠ \multicolumn, multirow

- MᴀᴛʜJᴀх does not support \multicolumn or multirow. These may be used in text tabulars or sᴠɢ math, but in MᴀᴛʜJᴀх 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 MᴀᴛʜJᴀх 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 MᴀᴛʜJᴀх.

⚠ references

- Inside a MᴀᴛʜJᴀх 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 MᴀᴛʜJᴀх 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 MᴀᴛʜJᴀхis 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 MᴀᴛʜJᴀх 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

- MᴀᴛʜJᴀх 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 MᴀᴛʜJᴀх. Either use XᴇLᴬTᴇX 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 MᴀᴛʜJᴀх, including \ensuremath and occasionally-used macros such as \relax. While using MᴀᴛʜJᴀх, 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

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 44.

changing contents
complicated alt tag

MathJax limitations

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

<div style="float: left">⚠ cref reference format undefined</div>

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:

- 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}.

---

[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 LᴬTᴇX 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 TᴇX tree, such as a test image from some LᴬTᴇX package. TᴇX looks in the local document
⚠ image not displayed directory before considering the directories specified by \graphicspath, but the TᴇX 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*)

⚠ displaymath and matrices

If using display math with tikzpicture or \tikz, along with matrices with the & 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 \&.

8.8.2 grffile package

grffile (*Pkg*)

⚠ matching PDF and SVG

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 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

\colorboxBlock and \fcolorboxBlock are provided for increased HTML compatibility, 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 tables

\rowcolors is supported, except that the optional argument is ignored so far.

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
⚠ convert image files   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`**

8.8.8 psfrag package

psfrag (*Pkg*)   The psfrags environment is modified to use lateximage to encapsulate the im-
⚠ use psfrags   age. Always use a psfrags environment to contain any local \psfrag macros and
the associated \includegraphics or \epsfig calls. Outside of a psfrags environ-
ment, 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*)   The macros \overpicfontsize and \overpicfontskip are used during HTML
⚠ scaling   generation. These are sent to \fontsize to adjust the font size for scaling differ-
ences 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*)   The packages multimedia, movie15, and media9 are supported.

movie15 (*Pkg*)
HTML5 <audio> and <video> objects are created for .mp3 and .mp4 files.

media9 (*Pkg*)
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 sup-
ported 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 429.2 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 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.

⚠ empty cells

```
. . . &   \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 tab character &
```
\StartDefiningTabulars
\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.

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
number of empty cells in the same column below \ldelim or \rdelim, but not in
⚠ use \mrowcell   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 639.1.

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
subcaption (*Pkg*)   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}}

⚠ 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
⚠ Misplaced alignment   defining macros using \ttabbox with a tabular inside. See section 8.10.1.
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.

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.

8.11.6 keyfloat package

keyfloat (*Pkg*)   If placing a \keyfig[H] inside a keywrap, use an absolute width for \keyfig,
⚠ keywrap 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*)   lwarp uses caption, which causes a warning from memoir. This is normal. Adjust
⚠ captions captions via caption, instead of memoir.

While emulating memoir, lwarp pre-loads a number of packages (section 695.1).
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}
```

⚠ 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
\HTMLvleftskip (*Len*)   and \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

⚠ \CaptionSeparator   a 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
luatodonotes (*Pkg*)   todo 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 410.

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*)   The lwarp options PrintLatexCmd and HTMLLatexCmd are used to set customized
HTMLLatexCmd (*Opt*) 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/).

⚠ search order   The EPUB conversion program must know what order the files are included. For 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

⚠ encoding   Check the box **Add linked files in breadth first order**. Set the document encoding as utf-8, which is what lwarp generates for HTML, even if the original printed document uses some other encoding.

⚠ section breaks   The EPUB-conversion program must also know where the section breaks are located. 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 LIBREOF-FICE 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
$\mathcal{AMS}$ 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 201.

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 93 for an example of the picture environment.

css classes
To create a custom HTML block or inline css class, see section 52.10.

print/HTML macros
To create print and HTML versions of the same macro or environment, see section 36.

⚠ 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 second round of index processing is 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 457 (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 20.1. 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. pdflatex lwarp.dtx

4. If modifying *lwarpmk* the new version should now be active.

5. 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.

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 delimiter.

Source name is set to `lwarp`: lwarp has recently been recompiled in this directory, 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. Supported 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 references 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. Recompile.

`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 error is generated, describing the filename and which section name generated it. A warning is 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 X$\exists$L^{A}T$_E$X 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 L^{A}T$_E$X.

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 L^{A}T$_E$X code is correct, before attempting to generate an HTML version.

\end{warpHTML}, \end{warpprint}, \end{warpall}, \end{warpMathJax}:

⚠ warpHTML, warpprint, warpMathJax, warpall
Each of these must be without any other characters on the same line.

"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 decrementing it after. Doing so tells lwarp to avoid using a \newpage inside the macro, which may avoid this error.

⚠ macros in section, table, figure names Also, custom macros which appear inside a section, figure, or table 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 environments **"\begin{equation} ended by \end{document}":** Do not use custom macros 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 36 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 ' character, 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 734 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 XƎLATEX 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 references.[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.

⚠ 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.

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 185.

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 Detecting the TeX engine — *pdflatex*, *lualatex*, *xelatex*

See: http://tex.stackexchange.com/a/47579.

Detects XƎTEX and LuaLATEX:

```
1 \RequirePackage{iftex}[2019/11/07]
2 \RequirePackage{ifpdf}
3 \RequirePackage{ifptex}
4
5 \newif\ifxetexorluatex
6
7 \ifXeTeX
8     \xetexorluatextrue
9 \else
10     \ifLuaTeX
11         \xetexorluatextrue
12     \else
13         \xetexorluatexfalse
14     \fi
15 \fi
```

19 Early package requirements

etoolbox (*Pkg*)   Provides \ifbool and other functions.

```
16 \RequirePackage{etoolbox}[2011/01/03]% v2.6 for \BeforeBeginEnvironment, etc.
```

Patch to fix copy of environment with a \par:

https://github.com/josephwright/etoolbox/issues/35

```
17 \long\def\etb@carsquare#1#2#3\@nil{#1#2}
```

xpatch (*Pkg*)   Patches macros with optional arguments.

```
18 \RequirePackage{xpatch}
```

ifplatform (*Pkg*)   Provides \ifwindows to try to automatically detect WINDOWS OS.

```
19 \RequirePackage{ifplatform}% sense op-system platform
```

letltxmacro (*Pkg*)

```
20 \RequirePackage{letltxmacro}
```

20 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 enfoced 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.

20.1   Tests of package load order

\LWR@loadafter {⟨*packagename*⟩}   Error if this package was loaded before lwarp.

```
21 \newcommand*{\LWR@loadafter}[1]{%
22 \IfPackageLoadedTF{#1}
23 {
24     \PackageError{lwarp}
25         {%
26             Package #1,\MessageBreak
27             or one which uses #1,\MessageBreak
28             must be loaded after Lwarp.\MessageBreak
29             Enter 'H' for possible solutions%
30         }
31         {%
32             Move ''\protect\usepackage{#1}'' after
33             ''\protect\usepackage{lwarp}''.\MessageBreak
34             Package #1 may also be loaded by something else,\MessageBreak
35             which must also be moved after Lwarp.%
36         }
37 }
38 {\relax}
39 }
```

\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.

```
40 \IfClassLoadedTF{memoir}
41 {\newcommand*{\LWR@notmemoirloadafter}[1]{}}
42 {\LetLtxMacro\LWR@notmemoirloadafter\LWR@loadafter}
```

\LWR@notltjloadafter {⟨*packagename*⟩}   Error if not a ltjs* class and this package was loaded before lwarp.

```
43 \LetLtxMacro\LWR@notltjloadafter\LWR@loadafter
44
45 \IfClassLoadedTF{ltjarticle}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
46 \IfClassLoadedTF{ltjbook}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
47 \IfClassLoadedTF{ltjreport}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
48 \IfClassLoadedTF{ltjsarticle}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
```

```
49 \IfClassLoadedTF{ltjsbook}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
50 \IfClassLoadedTF{ltjsreport}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
51 \IfClassLoadedTF{ltjspf}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
52 \IfClassLoadedTF{ltjskiyou}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
53 \IfClassLoadedTF{ltjtarticle}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
54 \IfClassLoadedTF{ltjtbook}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
55 \IfClassLoadedTF{ltjtreport}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
```

\LWR@loadbefore {⟨*packagename*⟩}   Error if this package is loaded after lwarp.

```
56 \newcommand*{\LWR@loadbefore}[1]{%
57 \IfPackageLoadedTF{#1}
58 {\relax}
59 {
60     \PackageError{lwarp}
61     {%
62         Package #1 must be loaded before lwarp.\MessageBreak
63         Enter 'H' for possible solutions%
64     }
65     {Move ''\protect\usepackage{#1}'' before ''\protect\usepackage{lwarp}''.}
66 }
67 }
```

\LWR@checkloadbefore {⟨*packagename*⟩}

Given \LWR@tempone is the package name to compare to, if package names match, error if it is loaded after lwarp.

```
68 \newcommand*{\LWR@checkloadbefore}[1]{%
69     \ifdefstring{\LWR@tempone}{#1}{%
70         \LWR@loadbefore{#1}%
71     }{}%
72 }
```

\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.

```
73 \newcommand*{\LWR@loadnever}[2]{%
74 \PackageError{lwarp}
75 {%
76     Package #1 is not yet supported\MessageBreak
77     by lwarp's HTML conversion%
78     \ifblank{#2}{}{%
79         .\MessageBreak
80         Package(s)\MessageBreak
81         \space\space#2\MessageBreak
82         may be useful instead%
83     }%
84 }
85 {%
86     Package #1 might conflict with lwarp in some way,\MessageBreak
87     or is superceded by another package.%
88     \ifblank{#2}{}{%
89         \MessageBreak
90         For possible alternatives, see package(s) #2.%
```

```
91      }%
92 }
93 }
```

\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.

```
94 \newcommand*{\LWR@afterloadnever}[2]{%
95    \ifdefstring{\LWR@tempone}{#1}{%
96        \LWR@loadnever{#1}{#2}%
97    }{}%
98 }
```

\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.

```
 99 \newcommand*{\LWR@earlyloadnever}[2]{%
100     \IfPackageLoadedTF{#1}{%
101         \LWR@loadnever{#1}{#2}%
102     }{}%
103 }
```

\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.

```
104 \newcommand*{\LWR@earlyclassloadnever}[2]{%
105 \IfClassLoadedTF{#1}{%
106 \PackageError{lwarp}
107 {%
108     Class #1 is not supported\MessageBreak
109     by lwarp's HTML conversion%
110     \ifblank{#2}{}{%
111         .\MessageBreak
112         #2 may be useful instead%
113     }%
114 }
115 {%
116     Class #1 might conflict with lwarp in some way,\MessageBreak
117     or is superceded by another class.%
118     \ifblank{#2}{}{%
119         \MessageBreak
120         For a possible alternative, see #2.%
121     }%
122 }
123 }{\relax}%
124 }
```

20.2 Error for disallowed packages and classes loaded before lwarp

`\LWR@checkloadnevers` Checks against a list of incompatible packages.

```
125 \newcommand*{\LWR@checkloadnevers}{
126 \LWR@checkloadnever{ae}{cm-super, lmodern}
127 \LWR@checkloadnever{aecompl}{cm-super, lmodern}
128 \LWR@checkloadnever{aecc}{cm-super, lmodern}
129 \LWR@checkloadnever{alg}{algorithm2e, algorithmicx}
130 \LWR@checkloadnever{algorithmic}{algorithm2e, algorithmicx}
131 \LWR@checkloadnever{bitfield}{bytefield}
```

bxcjkatype is based on CJK:

```
132 \LWR@checkloadnever{bxcjkjatype}{upLaTeX, bxjsarticle, ujarticle, utarticle}
```

```
133 \LWR@checkloadnever{caption2}{caption}
134 % \LWR@checkloadnever{ccaption}{caption}% might be preloaded by memoir
135 \LWR@checkloadnever{colortab}{colortbl}
136 \LWR@checkloadnever{csvtools}{datatool}
137 \LWR@checkloadnever{doublespace}{setspace}
138 \LWR@checkloadnever{fancyheadings}{fancyhdr}
139 \LWR@checkloadnever{fncylab}{cleveref}
140 \LWR@checkloadnever{formula}{siunitx}
141 \LWR@checkloadnever{glossary}{glossaries}
```

hangul is not in TeXLive, and is not tested:

```
142 \LWR@checkloadnever{hangul}{kotex, xetexko, luatexko}
```

```
143 \LWR@checkloadnever{hyper}{hyperref}
144 \LWR@checkloadnever{libgreek}{libertinust1math, newtx}
145 \LWR@checkloadnever{newthm}{ntheorem}
146 \LWR@checkloadnever{pdfcprot}{microtype}
147 \LWR@checkloadnever{picins}{floatflt, wrapfig, wrapfig2}
148 \LWR@checkloadnever{rplain}{fancyhdr}
149 \LWR@checkloadnever{si}{siunitx}
150 \LWR@checkloadnever{sistyle}{siunitx}
151 \LWR@checkloadnever{slashbox}{diagbox}
152 \LWR@checkloadnever{statex}{statex2}
153 \LWR@checkloadnever{t1enc}{fontenc, inputenc, inputenx}
154 \LWR@checkloadnever{ucs}{inputenc, inputencx}
155 \LWR@checkloadnever{wasysym}{textcomp, amssymb, amsfonts, mnsymbol, fdsymbol}
```

The following may one day be supported by lwarp:

```
156 % \LWR@checkloadnever{adjustbox}{}% req'd for menukeys
157 \LWR@checkloadnever{animate}{}
158 \LWR@checkloadnever{auto-pst-pdf}{}
159 \LWR@checkloadnever{auto-pst-pdf-lua}{}
160 \LWR@checkloadnever{algorithms}{}
161 \LWR@checkloadnever{arraycols}{}
162 \LWR@checkloadnever{bidi}{}
163 \LWR@checkloadnever{cals}{}
```

```
164 \LWR@checkloadnever{cellspace}{tabls}
```

```
165 \LWR@checkloadnever{cgloss4e}{}
166 \LWR@checkloadnever{collcell}{}
167 \LWR@checkloadnever{colophon}{}
168 \LWR@checkloadnever{cooltooltips}{}
169 \LWR@checkloadnever{covington}{}
170 \LWR@checkloadnever{crbox}{}
171 \LWR@checkloadnever{decision-table}{}
172 \LWR@checkloadnever{dvgloss}{}
173 \LWR@checkloadnever{ednotes}{}
174 \LWR@checkloadnever{edfnotes}{}
175 \LWR@checkloadnever{eledform}{}
176 \LWR@checkloadnever{eledmac}{}
177 \LWR@checkloadnever{embedfile}{}
178 \LWR@checkloadnever{endnotes-hy}{endnotes}
179 \LWR@checkloadnever{expex}{}
180 \LWR@checkloadnever{fancytooltips}{}
181 \LWR@checkloadnever{fixocgx}{}
182 \LWR@checkloadnever{flowfram}{}
183 \LWR@checkloadnever{gb4e}{}
184 \LWR@checkloadnever{gmverse}{}
185 \LWR@checkloadnever{graphbox}{}
186 \LWR@checkloadnever{graphicxbox}{}
187 \LWR@checkloadnever{hvfloat}{}
188 \LWR@checkloadnever{inline-images}{}
189 \LWR@checkloadnever{isorot}{rotating}
190 \LWR@checkloadnever{ledmac}{}
191 \LWR@checkloadnever{linguex}{}
192 \LWR@checkloadnever{longdiv}{}
193 \LWR@checkloadnever{longfigure}{}
194 \LWR@checkloadnever{longtabu}{}
195 \LWR@checkloadnever{mdwenv}{}
196 \LWR@checkloadnever{mdwlist}{}
197 \LWR@checkloadnever{mdwtab}{}
198 \LWR@checkloadnever{navigator}{}
199 \LWR@checkloadnever{nccpic}{}
200 \LWR@checkloadnever{nccsect}{}
201 \LWR@checkloadnever{newvbtm}{}
202 \LWR@checkloadnever{ocg-p}{}
203 \LWR@checkloadnever{ocgtools}{}
204 \LWR@checkloadnever{ocgx}{}
205 \LWR@checkloadnever{ocgx2}{}
206 \LWR@checkloadnever{parrun}{}
207 \LWR@checkloadnever{poemscol}{}
208 \LWR@checkloadnever{poetry}{}
209 \LWR@checkloadnever{program}{}
210 \LWR@checkloadnever{proofread}{}
211 \LWR@checkloadnever{pst-pdf}{}
212 \LWR@checkloadnever{refstyle}{}
213 \LWR@checkloadnever{robustindex}{}
214 \LWR@checkloadnever{robustglossary}{}
215 \LWR@checkloadnever{semioneside}{}
216 \LWR@checkloadnever{slemph}{}
217 \LWR@checkloadnever{snotez}{sidenotes}
218 \LWR@checkloadnever{spacingtricks}{}
219 \LWR@checkloadnever{sverb}{verbatim, fancyvrb}
220 \LWR@checkloadnever{syntax}{}
221 \LWR@checkloadnever{tablists}{}
222 \LWR@checkloadnever{tabto}{}
223 \LWR@checkloadnever{tabu}{}
224 \LWR@checkloadnever{tabularht}{}
```

```
225 \LWR@checkloadnever{tabularkv}{}
226 \LWR@checkloadnever{thumby}{}
227 \LWR@checkloadnever{titles}{}
228 \LWR@checkloadnever{typehtml}{}
229 \LWR@checkloadnever{unicode-bidi}{}
230 \LWR@checkloadnever{vcell}{}
231 \LWR@checkloadnever{xhfill}{}
232 }
```

\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.

```
233 \LetLtxMacro\LWR@checkloadnever\LWR@earlyloadnever
```

Now check for incompatible packages which have been loaded before lwarp:

```
234 \LWR@checkloadnevers
```

The older CJK and CJKutf8 only work with xeCJK:

```
235 \IfPackageLoadedTF{xeCJK}{}{
236     \LWR@checkloadnever{CJK}{ctex, xeCJK}
237     \LWR@checkloadnever{CJKutf8}{ctex, xeCJK}
238 }
```

Some classes do not work with lwarp:

```
239 \LWR@earlyclassloadnever{beamer}{beamerarticle}
240 \LWR@earlyclassloadnever{jarticle}{ujarticle}
241 \LWR@earlyclassloadnever{jbook}{ujbook}
242 \LWR@earlyclassloadnever{jreport}{ujreport}
243 \LWR@earlyclassloadnever{tarticle}{utarticle}
244 \LWR@earlyclassloadnever{tbook}{utbook}
245 \LWR@earlyclassloadnever{treport}{utreport}
246 \LWR@earlyclassloadnever{novel}{}
247 \LWR@earlyclassloadnever{powerdot}{}
```

20.3   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:

```
248 \LWR@loadafter{2in1}
249 \LWR@loadafter{2up}
250 \LWR@loadafter{a4}
251 \LWR@loadafter{a4wide}
252 \LWR@loadafter{a5comb}
253 \LWR@notmemoirloadafter{abstract}
```

```
254 \LWR@loadafter{academicons}
255 \LWR@loadafter{accents}
256 \LWR@loadafter{accessibility}
257 \LWR@loadafter{accsupp}
258 \LWR@loadafter{acro}
259 \LWR@loadafter{acronym}
260 \LWR@loadafter{adjmulticol}
261 \LWR@loadafter{addlines}
262 \LWR@loadafter{afterpage}
263 \LWR@loadafter{algorithm2e}
264 \LWR@loadafter{algorithmicx}
265 \LWR@loadafter{alltt}
266 \LWR@loadafter{amscdx}
267 % \LWR@loadafter{amsmath}% may be preloaded
268 % \LWR@loadafter{amsthm}% may be preloaded
269 \LWR@loadafter{anonchap}
270 \LWR@loadafter{anysize}
271 \LWR@notmemoirloadafter{appendix}
272 \LWR@loadafter{ar}
273 \LWR@loadafter{arabicfront}
274 \LWR@notmemoirloadafter{array}
275 \LWR@loadafter{arydshln}
276 \LWR@loadafter{asymptote}
277 % \LWR@loadafter{atbegshi}% now in LaTeX core, also used by morewrites
278 \LWR@loadafter{attachfile}
279 \LWR@loadafter{attachfile2}
280 \LWR@loadafter{authblk}
281 \LWR@loadafter{authoraftertitle}% Supported as-is, but must be loaded after.
282 \LWR@loadafter{autobreak}
283 \LWR@loadafter{autonum}
284 \LWR@loadafter{awesomebox}
285 \LWR@loadafter{axessibility}
286 \LWR@loadafter{axodraw2}
287 \LWR@loadafter{backnaur}
288 \LWR@loadafter{backref}
289 \LWR@loadafter{balance}
290 \LWR@loadafter{bbding}
291 \LWR@loadafter{beamerarticle}
292 \LWR@loadafter{bigdelim}
293 \LWR@loadafter{bigfoot}
294 \LWR@loadafter{bigstrut}
295 \LWR@loadafter{bitpattern}
296 \LWR@loadafter{blowup}
297 \LWR@loadafter{bm}
298 \LWR@loadafter{booklet}
299 \LWR@loadafter{bookmark}
300 \LWR@notmemoirloadafter{booktabs}
301 \LWR@loadafter{bophook}
302 \LWR@loadafter{bounddvi}
303 \LWR@loadafter{boxedminipage}
304 \LWR@loadafter{boxedminipage2e}
305 \LWR@loadafter{braket}
306 \LWR@loadafter{breakurl}
307 \LWR@loadafter{breqn}
308 \LWR@loadafter{bsheaders}
309 \LWR@loadafter{bussproofs}
310 \LWR@loadafter{bxpapersize}
311 \LWR@loadafter{bytefield}
312 \LWR@loadafter{ccicons}
313 \LWR@loadafter{cancel}
```

```
314 \LWR@loadafter{canoniclayout}
315 \LWR@loadafter{caption}
316 \LWR@loadafter{caption2}
317 \LWR@loadafter{caption3}
318 \LWR@loadafter{cases}
319 % catoptions is supported by the lwarp core
320 % \LWR@loadafter{ccaption}% may be preloaded by memoir
321 \LWR@loadafter{centerlastline}
322 % \LWR@loadafter{centernot}% may be preloaded by newtx
323 \LWR@loadafter{changebar}
324 \LWR@loadafter{changelayout}
325 \LWR@notmemoirloadafter{changepage}
326 \LWR@loadafter{changes}
327 \LWR@loadafter{chappg}
328 \LWR@loadafter{chapterbib}
329 \LWR@loadafter{chemfig}
330 \LWR@loadafter{chemformula}
331 \LWR@loadafter{chemgreek}
332 \LWR@loadafter{chemmacros}
333 \LWR@loadafter{chemnum}
334 \LWR@loadafter{chkfloat}
335 \LWR@notmemoirloadafter{chngpage}
336 \LWR@loadafter{cite}
337 \LWR@loadafter{citeref}
338 \LWR@loadafter{classicthesis}
339 \LWR@loadafter{cleveref}
340 % cmbright may be preloaded
341 \LWR@loadafter{cmdtrack}
342 \LWR@loadafter{colonequals}
343 \LWR@loadafter{color}
344 \LWR@loadafter{colortbl}
345 \LWR@loadafter{continue}
346 \LWR@loadafter{copyrightbox}
347 \LWR@notmemoirloadafter{crop}
348 % ctex must be loaded before lwarp
349 \LWR@loadafter{ctable}
350 \LWR@loadafter{cuted}
351 \LWR@loadafter{cutwin}
352 \LWR@loadafter{dblfloatfix}
353 \LWR@loadafter{dblfnote}
354 \LWR@notmemoirloadafter{dcolumn}
355 \LWR@loadafter{decimal}
356 \LWR@loadafter{decorule}
357 \LWR@loadafter{diagbox}
358 \LWR@loadafter{dingbat}
359 \LWR@loadafter{DotArrow}
360 \LWR@loadafter{dotlessi}
361 \LWR@loadafter{dprogress}
362 \LWR@loadafter{draftcopy}
363 \LWR@loadafter{draftfigure}
364 \LWR@loadafter{draftwatermark}
365 \LWR@loadafter{drftcite}
366 \LWR@loadafter{easy-todo}
367 \LWR@loadafter{ebook}
368 \LWR@loadafter{econometrics}
369 \LWR@loadafter{ed}
370 \LWR@loadafter{ellipsis}
371 \LWR@loadafter{embrac}
372 \LWR@loadafter{emptypage}
373 \LWR@loadafter{endfloat}
```

```
374 \LWR@loadafter{endheads}
375 \LWR@loadafter{endnotes}
376 \LWR@loadafter{engtlc}
377 \LWR@loadafter{enotez}
378 \LWR@notmemoirloadafter{enumerate}
379 \LWR@loadafter{enumitem}
380 \LWR@notmemoirloadafter{epigraph}
381 \LWR@loadafter{epsf}
382 \LWR@loadafter{epsfig}
383 \LWR@loadafter{epstopdf}
384 \LWR@loadafter{epstopdf-base}
385 \LWR@loadafter{eqlist}
386 \LWR@loadafter{eqparbox}
387 \LWR@loadafter{errata}
388 \LWR@loadafter{eso-pic}
389 \LWR@loadafter{esvect}
390 \LWR@loadafter{etoc}
391 \LWR@loadafter{eurosym}
392 \LWR@loadafter{everypage}
393 % \LWR@loadafter{everyshi}% now in LaTeX core
394 \LWR@loadafter{extarrows}
395 \LWR@loadafter{extramarks}
396 \LWR@loadafter{fancybox}
397 \LWR@loadafter{fancyhdr}
398 \LWR@loadafter{fancypar}
399 \LWR@loadafter{fancyref}
400 \LWR@loadafter{fancytabs}
401 \LWR@loadafter{fancyvrb}
402 \LWR@loadafter{fbox}
403 \LWR@loadafter{fewerfloatpages}
404 \LWR@loadafter{figcaps}
405 \LWR@loadafter{figsize}
406 \LWR@loadafter{fitbox}
407 \LWR@loadafter{fix2col}
408 \LWR@loadafter{fixmath}
409 \LWR@loadafter{fixme}
410 \LWR@loadafter{fixmetodonotes}
411 \LWR@loadafter{flafter}
412 \LWR@loadafter{flippdf}
413 \LWR@loadafter{float}
414 \LWR@loadafter{floatflt}
415 \LWR@loadafter{floatpag}
416 \LWR@loadafter{floatrow}
417 \LWR@loadafter{fltrace}
418 \LWR@loadafter{flushend}
419 \LWR@loadafter{fnbreak}
420 \LWR@loadafter{fncychap}
421 \LWR@loadafter{fnlineno}
422 \LWR@loadafter{fnpara}
423 \LWR@loadafter{fnpos}
424 \LWR@loadafter{fontawesome}
425 \LWR@loadafter{fontawesome5}
426 % fontenc must be loaded before lwarp
427 % fontspec must be loaded before lwarp
428 \LWR@loadafter{footmisc}
429 \LWR@loadafter{footnote}
430 \LWR@loadafter{footnotebackref}
431 \LWR@loadafter{footnotehyper}
432 \LWR@loadafter{footnoterange}
433 \LWR@loadafter{footnpag}
```

```
434 \LWR@loadafter{foreign}
435 \LWR@loadafter{forest}
436 \LWR@loadafter{fouridx}
437 % fourier may be loaded before lwarp
438 \LWR@loadafter{framed}
439 \LWR@loadafter{froufrou}
440 \LWR@loadafter{ftcap}
441 \LWR@loadafter{ftnright}
442 \LWR@loadafter{fullminipage}
443 \LWR@loadafter{fullpage}
444 \LWR@loadafter{fullwidth}
445 \LWR@loadafter{fvextra}
446 \LWR@loadafter{fwlw}
447 \LWR@loadafter{gensymb}
448 \LWR@loadafter{gentombow}
449 % geometry is always loaded by lwarp, and lwarp-geometry is AtBeginDocument
450 \LWR@loadafter{ghsystem}
451 \LWR@loadafter{gindex}
452 \LWR@loadafter{glossaries}
453 \LWR@loadafter{gmeometric}
454 % \LWR@loadafter{graphics}% pre-loaded by xunicode
455 % \LWR@loadafter{graphicx}% pre-loaded by xunicode
456 \LWR@loadafter{gloss}
457 \LWR@loadafter{glossary}
458 \LWR@loadafter{grffile}
459 \LWR@loadafter{grid}
460 \LWR@loadafter{grid-system}
461 \LWR@loadafter{gridset}
462 \LWR@loadafter{hang}
463 \LWR@loadafter{hanging}
464 \LWR@loadafter{hepunits}
465 \LWR@loadafter{hhline}
466 \LWR@loadafter{hhtensor}
467 \LWR@loadafter{hypbmsec}
468 \LWR@loadafter{hypcap}
469 \LWR@loadafter{hypdestopt}
470 \LWR@loadafter{hypernat}
471 \LWR@loadafter{hyperref}
472 \LWR@loadafter{hyperxmp}
473 \LWR@loadafter{hyphenat}
474 \LWR@loadafter{idxlayout}
475 \LWR@loadafter{ifoddpage}
476 \LWR@loadafter{imakeidx}
477 \LWR@loadafter{impnattypo}
478 \LWR@notmemoirloadafter{index}
479 % inputenc must be loaded before lwarp
480 % inputenx must be loaded before lwarp
481 % inputtrc may be loaded before lwarp
482 \LWR@loadafter{intopdf}
483 \LWR@loadafter{isomath}
484 \LWR@loadafter{isotope}
485 \LWR@loadafter{jurabib}
486 \LWR@loadafter{karnaugh-map}
487 \LWR@loadafter{keyfloat}
488 \LWR@loadafter{keystroke}
489 % kpfonts may be loaded before lwarp
490 % kpfonts-otf may be loaded before lwarp
491 \LWR@loadafter{layaureo}
492 \LWR@loadafter{layout}
493 \LWR@loadafter{layouts}
```

```
494 \LWR@loadafter{leading}
495 \LWR@loadafter{leftidx}
496 \LWR@loadafter{letterspace}
497 \LWR@loadafter{lettrine}
498 % libertinust1math may be loaded before lwarp
499 \LWR@loadafter{lineno}
500 \LWR@loadafter{lips}
501 \LWR@loadafter{listings}
502 \LWR@loadafter{listliketab}
503 \LWR@loadafter{lltjp-siunitx}
504 \LWR@loadafter{lltjp-tascmac}
505 \LWR@loadafter{longtable}
506 \LWR@loadafter{lpic}
507 \LWR@loadafter{lscape}
508 \LWR@loadafter{ltablex}
509 \LWR@loadafter{ltcaption}
510 \LWR@loadafter{ltxgrid}
511 \LWR@loadafter{ltxtable}
512 \LWR@loadafter{lua-check-hyphen}
513 \LWR@loadafter{lua-visual-debug}
514 \LWR@loadafter{luacolor}
515 \LWR@loadafter{luamplib}
516 \LWR@loadafter{luatodonotes}
517 \LWR@loadafter{luavlna}
518 \LWR@loadafter{lyluatex}
519 \LWR@loadafter{magaz}
520 \LWR@notmemoirloadafter{makeidx}
521 \LWR@loadafter{manyfoot}
522 \LWR@loadafter{marginfit}
523 \LWR@loadafter{marginfix}
524 \LWR@loadafter{marginnote}
525 \LWR@loadafter{marvosym}
526 % mathalpha may be loaded before lwarp
527 \LWR@loadafter{mathastext}
528 \LWR@loadafter{mathcomp}
529 \LWR@loadafter{mathdesign}
530 \LWR@loadafter{mathdots}
531 \LWR@loadafter{mathfixs}
532 \LWR@loadafter{mathpazo}
533 \LWR@loadafter{mathptmx}
534 \LWR@loadafter{mathspec}
535 \LWR@loadafter{mathtools}
536 \LWR@loadafter{mattens}
537 \LWR@loadafter{maybemath}
538 \LWR@loadafter{mcaption}
539 \LWR@loadafter{mdframed}
540 \LWR@loadafter{mdwmath}
541 \LWR@loadafter{media9}
542 \LWR@loadafter{memhfixc}
543 \LWR@loadafter{menukeys}
544 \LWR@loadafter{metalogo}
545 \LWR@loadafter{metalogox}
546 \LWR@loadafter{mhchem}
547 \LWR@loadafter{microtype}
548 \LWR@loadafter{midfloat}
549 \LWR@loadafter{midpage}
550 \LWR@loadafter{minibox}
551 \LWR@loadafter{minitoc}
552 \LWR@loadafter{minted}
553 \LWR@loadafter{mismath}
```

```
554 \LWR@loadafter{mleftright}
555 % morefloats must be allowed early for print mode
556 \LWR@notmemoirloadafter{moreverb}
557 % morewrites must be loaded before lwarp
558 \LWR@notmemoirloadafter{movie15}
559 \LWR@notmemoirloadafter{mparhack}
560 \LWR@loadafter{multibib}
561 \LWR@loadafter{multicap}
562 %\LWR@loadafter{multicol}% loaded by ltxdoc
563 \LWR@loadafter{multicolrule}
564 \LWR@loadafter{multimedia}
565 \LWR@loadafter{multiobjective}
566 \LWR@loadafter{multirow}
567 \LWR@loadafter{multitoc}
568 \LWR@loadafter{musicography}
569 \LWR@loadafter{mwe}
570 \LWR@loadafter{nameauth}
571 \LWR@loadafter{nameref}
572 \LWR@loadafter{natbib}
573 \LWR@notmemoirloadafter{nccfancyhdr}
574 \LWR@loadafter{nccfoots}
575 \LWR@loadafter{nccmath}
576 \LWR@notmemoirloadafter{needspace}
577 % newclude must be loaded before lwarp
578 % newpxmath may be preloaded
579 % newtxmath may be loaded before lwarp
580 % newtxsf may be loaded before lwarp
581 % newunicodechar must be loaded before lwarp
582 \LWR@notmemoirloadafter{nextpage}
583 \LWR@loadafter{nicefrac}
584 \LWR@loadafter{niceframe}
585 \LWR@loadafter{nicematrix}
586 \LWR@loadafter{noitcrul}
587 \LWR@loadafter{nolbreaks}
588 \LWR@loadafter{nomencl}
589 \LWR@loadafter{nonfloat}
590 \LWR@loadafter{nonumonpart}
591 \LWR@loadafter{nopageno}
592 \LWR@loadafter{notes}
593 \LWR@loadafter{notespages}
594 \LWR@loadafter{nowidow}
595 \LWR@loadafter{ntheorem}
596 \LWR@loadafter{octave}
597 \LWR@loadafter{orcidlink}
598 \LWR@loadafter{overpic}
599 \LWR@loadafter{pagegrid}
600 \LWR@notmemoirloadafter{pagenote}
601 \LWR@loadafter{pagesel}
602 \LWR@loadafter{paralist}
603 \LWR@loadafter{parallel}
604 \LWR@loadafter{parcolumns}
605 \LWR@loadafter{parnotes}
606 \LWR@notmemoirloadafter{parskip}
607 \LWR@loadafter{pbalance}
608 \LWR@loadafter{pbox}
609 \LWR@loadafter{pdfcol}
610 \LWR@loadafter{pdfcolfoot}
611 \LWR@loadafter{pdfcolmk}
612 \LWR@loadafter{pdfcolparallel}
613 \LWR@loadafter{pdfcolparcolumns}
```

```
614 \LWR@loadafter{pdfcomment}
615 \LWR@loadafter{pdfcrypt}
616 \LWR@loadafter{pdflscape}
617 \LWR@loadafter{pdfmarginpar}
618 \LWR@loadafter{pdfpages}
619 \LWR@loadafter{pdfprivacy}
620 \LWR@loadafter{pdfrender}
621 \LWR@loadafter{pdfsync}
622 \LWR@loadafter{pdftricks}
623 \LWR@loadafter{pdfx}
624 \LWR@loadafter{perpage}
625 \LWR@loadafter{pfnote}
626 \LWR@loadafter{phfqit}
627 \LWR@loadafter{physics}
628 \LWR@loadafter{physunits}
629 \LWR@loadafter{picinpar}
630 \LWR@loadafter{pifont}
631 \LWR@loadafter{pinlabel}
632 \LWR@loadafter{placeins}
633 \LWR@loadafter{plarray}
634 \LWR@loadafter{plarydshln}
635 \LWR@loadafter{plextarray}
636 \LWR@loadafter{plextarydshln}
637 \LWR@loadafter{plcolortbl}
638 \LWR@loadafter{plextdelarray}
639 \LWR@loadafter{plimsoll}
640 \LWR@loadafter{prelim2e}
641 \LWR@loadafter{prettyref}
642 \LWR@loadafter{preview}
643 \LWR@loadafter{psfrag}
644 \LWR@loadafter{psfragx}
645 \LWR@loadafter{pst-eps}
646 \LWR@loadafter{pstool}
647 \LWR@loadafter{pstricks}
648 % \LWR@loadafter{pxatbegshi}% may be used by morewrites
649 \LWR@loadafter{pxeveryshi}
650 % \LWR@loadafter{pxfonts}% may be loaded before lwarp
651 \LWR@loadafter{pxftnright}
652 \LWR@loadafter{pxjahyper}
653 \LWR@loadafter{quotchap}
654 \LWR@loadafter{quoting}
655 \LWR@loadafter{ragged2e}
656 \LWR@loadafter{refcheck}
657 \LWR@loadafter{register}
658 \LWR@loadafter{relsize}
659 \LWR@loadafter{repeatindex}
660 \LWR@loadafter{resizegather}
661 \LWR@loadafter{returntogrid}
662 \LWR@loadafter{rlepsf}
663 \LWR@loadafter{rmathbr}
664 \LWR@loadafter{rmpage}
665 \LWR@loadafter{romanbar}
666 \LWR@loadafter{romanbarpagenumber}
667 \LWR@loadafter{rotating}
668 \LWR@loadafter{rotfloat}
669 \LWR@loadafter{rviewport}
670 \LWR@loadafter{savetrees}
671 % scalefnt is loaded by babel-french
672 \LWR@loadafter{scalerel}
673 \LWR@loadafter{schemata}
```

```
674 \LWR@loadafter{scrextend}
675 \LWR@loadafter{scrhack}
676 \LWR@loadafter{scrlayer}
677 \LWR@loadafter{scrlayer-notecolumn}
678 \LWR@loadafter{scrlayer-scrpage}
679 \LWR@loadafter{scrpage2}
680 \LWR@loadafter{section}
681 \LWR@loadafter{sectionbreak}
682 \LWR@loadafter{sectsty}
683 \LWR@loadafter{selectp}
684 \LWR@loadafter{semantic-markup}
685 \LWR@notmemoirloadafter{setspace}
686 \LWR@loadafter{shadow}
687 \LWR@loadafter{shapepar}
688 \LWR@notmemoirloadafter{showidx}
689 \LWR@loadafter{showlabels}
690 \LWR@loadafter{showkeys}
691 \LWR@loadafter{showtags}
692 \LWR@loadafter{shuffle}
693 \LWR@loadafter{sidecap}
694 \LWR@loadafter{sidenotes}
695 \LWR@loadafter{simplebnf}
696 \LWR@loadafter{SIunits}
697 \LWR@loadafter{siunitx}
698 \LWR@loadafter{siunitx-v2}
699 \LWR@loadafter{skmath}
700 \LWR@loadafter{slantsc}
701 \LWR@loadafter{slashed}
702 \LWR@loadafter{soul}
703 \LWR@loadafter{soulpos}
704 \LWR@loadafter{soulutf8}
705 \LWR@loadafter{splitbib}
706 \LWR@loadafter{splitidx}
707 \LWR@loadafter{srcltx}
708 \LWR@loadafter{srctex}
709 \LWR@loadafter{stabular}
710 \LWR@loadafter{stackengine}
711 \LWR@loadafter{stackrel}
712 \LWR@loadafter{statex2}
713 \LWR@loadafter{statistics}
714 \LWR@loadafter{statmath}
715 \LWR@loadafter{steinmetz}
716 \LWR@notltjloadafter{stfloats}
717 \LWR@loadafter{struktex}
718 \LWR@loadafter{subcaption}
719 \LWR@loadafter{subfig}
720 \LWR@loadafter{subfigure}
721 \LWR@loadafter{subsupscripts}
722 \LWR@loadafter{supertabular}
723 \LWR@loadafter{svg}
724 \LWR@loadafter{swfigure}
725 \LWR@loadafter{sympytex}
726 \LWR@loadafter{syntonly}
727 \LWR@loadafter{t1inc}
728 \LWR@loadafter{tabfigures}
729 \LWR@loadafter{tabls}
730 \LWR@loadafter{tablefootnote}
731 \LWR@notmemoirloadafter{tabularx}
732 \LWR@loadafter{tabulary}
733 \LWR@loadafter{tagpdf}
```

```
734 \LWR@loadafter{tagpdf-mc-code-generic}
735 \LWR@loadafter{tagpdf-mc-code-lua}
736 \LWR@loadafter{tascmac}
737 \LWR@loadafter{tcolorbox}
738 \LWR@loadafter{tensor}
739 \LWR@loadafter{termcal}
740 \LWR@loadafter{textarea}
741 % \LWR@loadafter{textcomp}% maybe before lwarp with font packages
742 \LWR@loadafter{textfit}
743 \LWR@loadafter{textpos}
744 \LWR@loadafter{theorem}
745 \LWR@loadafter{thinsp}
746 \LWR@loadafter{thm-listof}
747 \LWR@loadafter{thm-restate}
748 \LWR@loadafter{thmbox}
749 \LWR@loadafter{thmtools}
750 \LWR@loadafter{threadcol}
751 \LWR@loadafter{threeparttable}
752 \LWR@loadafter{threeparttablex}
753 \LWR@loadafter{thumb}
754 \LWR@loadafter{thumbs}
755 \LWR@loadafter{tikz}
756 \LWR@loadafter{tikz-imagelabels}
757 \LWR@loadafter{titleps}
758 \LWR@loadafter{titlesec}
759 \LWR@loadafter{titletoc}
760 \LWR@notmemoirloadafter{titling}
761 % \LWR@loadafter{tocbasic}% preloaded by koma-script classes
762 \LWR@notmemoirloadafter{tocbibind}
763 \LWR@loadafter{tocdata}
764 \LWR@loadafter{tocenter}
765 \LWR@notmemoirloadafter{tocloft}
766 \LWR@loadafter{tocstyle}
767 \LWR@loadafter{todo}
768 \LWR@loadafter{todonotes}
769 \LWR@loadafter{topcapt}
770 \LWR@loadafter{tram}
771 \LWR@loadafter{transparent}
772 \LWR@loadafter{trimclip}
773 \LWR@loadafter{trivfloat}
774 \LWR@loadafter{truncate}
775 \LWR@loadafter{turnthepage}
776 \LWR@loadafter{twoup}
777 % \LWR@loadafter{txfonts}% may be loaded before lwarp
778 % txgreeks may be loaded before lwarp

779 % \LWR@loadafter{typearea}% preloaded by koma-script classes
780 \LWR@loadafter{typicons}
781 % \LWR@loadafter{ulem}% preloaded by ctexart and related classes
782 \LWR@loadafter{umoline}
783 \LWR@loadafter{underscore}
784 % unicode-math may be loaded before lwarp
785 \LWR@loadafter{units}
786 \LWR@loadafter{unitsdef}
787 \LWR@loadafter{upgreek}
788 \LWR@loadafter{upref}
789 \LWR@loadafter{url}
790 \LWR@loadafter{ushort}
791 \LWR@loadafter{uspace}
792 \LWR@loadafter{varioref}
```

```
793 \LWR@notmemoirloadafter{verse}
794 \LWR@loadafter{versonotes}
795 \LWR@loadafter{vertbars}
796 \LWR@loadafter{vmargin}
797 \LWR@loadafter{vowel}
798 \LWR@loadafter{vpe}
799 \LWR@loadafter{vwcol}
800 \LWR@loadafter{wallpaper}
801 \LWR@loadafter{watermark}
802 \LWR@loadafter{widetable}
803 \LWR@loadafter{widows-and-orphans}
804 \LWR@loadafter{witharrows}
805 \LWR@loadafter{wrapfig}
806 \LWR@loadafter{wrapfig2}
807 \LWR@loadafter{xbmks}
808 \LWR@loadafter{xcolor}
809 \LWR@loadafter{xechangebar}
810 \LWR@loadafter{xellipsis}
811 % xetexko must be loaded before lwarp
812 \LWR@loadafter{xevlna}
813 \LWR@loadafter{xfakebold}
814 \LWR@loadafter{xfrac}
815 \LWR@loadafter{xltabular}
816 \LWR@loadafter{xltxtra}
817 \LWR@loadafter{xmpincl}
818 \LWR@loadafter{xpiano}
819 \LWR@loadafter{xpinyin}
820 \LWR@loadafter{xr}
821 \LWR@loadafter{xr-hyper}
822 \LWR@loadafter{xtab}
823 % xunicode must be loaded before lwarp
824 \LWR@loadafter{xurl}
825 \LWR@loadafter{xy}
826 \LWR@loadafter{zwpagelayout}
```

21   MD5 hashing

The MD5 hash is used for lateximage filenames for svg math.

```
827 \newcommand{\LWR@mdfive}[1]{%
828     \PackageError{lwarp}
829         {No MD5 macro was found}
830         {%
831             Lwarp must find the macros \protect\pdfmdfivesum\space
832             or \protect\mdfivesum.%
833         }
834 }
```

The default for pdf LaTeX, dvi LaTeX, upLaTeX, etc:

```
835 \let\LWR@mdfive\pdfmdfivesum
```

For LuaLaTeX:

```
836 \ifLuaTeX
837 \RequirePackage{pdftexcmds}
```

```
838 \let\LWR@mdfive\pdf@mdfivesum
839 \fi
```

For X∃LATEX:

```
840 \ifXeTeX
841 \@ifundefined{pdffivesum}{}
842     {\let\LWR@mdfive\pdfmdfivesum}
843 \@ifundefined{mdfivesum}{}
844     {\let\LWR@mdfive\mdfivesum}
845 \fi
```

22   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.

X∃LATEX and LuaLATEX are both UTF-8 by nature.

\LWR@pdfencoding Sets T1, and also utf8 if not already set.

```
846 \newcommand*{\LWR@pdfencoding}{%
847     \RequirePackage[T1]{fontenc}
848
849     \IfPackageLoadedTF{inputenc}{}{
850         \IfPackageLoadedTF{inputenx}{}{
851             \RequirePackage[utf8]{inputenc}
852         }
853     }
854 }
```

```
855 \ifPDFTeX% pdflatex or dvi latex
856     \LWR@pdfencoding
857 \fi
858
859 \ifpTeX
860     \LWR@pdfencoding
861 \fi
```

23   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.

```
862 \ifpTeX
```

```
863 \else
864 \RequirePackage{newunicodechar}
865
866 \newunicodechar{×}{\texttimes}
867
868 \ifPDFTeX% pdflatex or dvi latex
869 \newunicodechar{ff}{ff}% Here, the first arguments are ligatures.
870 \newunicodechar{fi}{fi}
871 \newunicodechar{fl}{fl}
872 \newunicodechar{ffi}{ffi}
873 \newunicodechar{ffl}{ffl}
874 \newunicodechar{—}{---}
875 \newunicodechar{–}{--}
876 \fi
877
878 \fi
```

24   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.

```
879 \ifxetexorluatex
880 \else
881     \ifdefstring{\f@family}{cmr}{
882         \IfFileExists{type1ec.sty}% found in cm-super
883         {}
884         {% cm-super not installed
885             \IfFileExists{lmodern.sty}{
886                 \PackageInfo{lwarp}{cm-super not installed, loading lmodern}
887                 \RequirePackage{lmodern}
888             }{
889                 \PackageError{lwarp}
890                 {%
891                     Lwarp requires a vector font.\MessageBreak
892                  Install and load cm-super, lmodern, or another\MessageBreak
893                     Type-1 vector font before loading lwarp.\MessageBreak
894                     Enter 'H' for possible solutions%
895                 }
896                 {%
897                     Install cm-super or lmodern.\MessageBreak
898                     If lmodern, load it before lwarp:\MessageBreak
899                         \space\space\protect\usepackage{lmodern}\MessageBreak
900                         \space\space\protect\usepackage{lwarp}%
901                 }
902             }
903         }% cm-super not installed
904     }{}% f@family
905 \fi
```

25   Upright quotes

In PDF TEX, preserve upright quotes in verbatim text. **upquote** also loads **textcomp**.

```
906 \ifPDFTeX
907 \RequirePackage{upquote}
908 \fi
909
910 \ifpTeX
911     \RequirePackage{upquote}
912 \fi
```

26 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.

```
913 \ifxetexorluatex
914     \AtBeginDocument{
915         \IfPackageLoadedTF{kpfonts}{
916             \PackageError{lwarp}
917             {%
918                 When using XeLaTeX or LuaLaTeX,\MessageBreak
919                 use kpfonts-otf instead of kpfonts%
920             }
921             {%
922                 Replace: \protect\usepackage{kpfonts}\MessageBreak
923                 with: \protect\usepackage{kpfonts-otf}
924             }
925         }{}
926     }
927 \fi
```

27 Miscellaneous tools

27.1 Variables

```
928 \newlength{\LWR@templengthone}
929 \newlength{\LWR@templengthtwo}
930 \newlength{\LWR@templengththree}
931 \newcounter{LWR@tempcountone}
```

27.2 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.

```
932 \newcommand*\LWR@providelength[1]{%
933     \ifdeflength{#1}{}{\newlength{#1}}%
934 }
```

\LWR@convertto {⟨*dest unit*⟩} {⟨*length*⟩}

Prints a length in the given units, without printing the unit itself.

```
935 \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.

```
936 \newcommand*{\LWR@printpercentlength}[2]{%
937     \setcounter{LWR@tempcountone}{100*\ratio{#1}{#2}}%
938     \arabic{LWR@tempcountone}%
939 }
```

27.3   Counters

\defaddtocounter {⟨*name*⟩} {⟨*value*⟩}

Locally add to a counter.

```
940 \providecommand*{\defaddtocounter}[2]{%
941     \defcounter{#1}{\value{#1}+#2}%
942 }
```

27.4   Patching macros

\LWR@patcherror {⟨*packagename*⟩} {⟨*macroname*⟩}

Prints an error if could not patch a macro.

```
943 \newcommand*{\LWR@patcherror}[2]{%
944     \PackageError{lwarp}%
945         {%
946             Unable to patch package #1,\MessageBreak
947             macro \LWRbackslash #2.\MessageBreak
948             Lwarp or #1 may need to be updated%
949         }%
950         {Please contact the maintainer of the Lwarp package.}%
951 }
```

27.5   Copying macros

\csNewCommandCopycs {⟨*dest csname*⟩} {⟨*source csname*⟩}

Given a cs-name for each, copies a macro to a new definition.

```
952 \providecommand*{\csNewCommandCopycs}[2]{%
953     \expandafter\NewCommandCopy\csname#1\expandafter\endcsname%
954         \csname#2\endcsname%
955 }
```

\NewEnvironmentCopy {⟨*dest*⟩} {⟨*source*⟩}

Copies an environment to a new definition.

```
956 \providecommand*{\NewEnvironmentCopy}[2]{%
957     \csNewCommandCopycs{#1}{#2}%
958     \csNewCommandCopycs{end#1}{end#2}%
959 }
```

27.6   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.

```
960 \newcommand{\LWR@isolate}[1]{#1}%
961
962 \IfPackageLoadedTF{ctexpatch}{
963     \renewcommand{\LWR@isolate}[1]{\null#1\null}%
964 }{}
965
966 \IfPackageLoadedTF{xeCJK}{
967     \renewcommand{\LWR@isolate}[1]{\null#1\null}%
968 }{}
```

\LWR@disablepinyin Disable xpinyin during file, sideTOC, and footnote generation. Set by xpinyin.

```
969 \newcommand*{\LWR@disablepinyin}{}
```

27.7   Inserting vertical space

\LWR@forceemptyline Extra vertical space in the HTML output. Use after \LWR@stoppars.

```
970 \newcommand*{\LWR@forceemptyline}{%
971     \LWR@origrule{0pt}{1\baselineskip}%
972     \LWR@orignewline%
973 }
```

27.8   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.

```
974 \long\def\LWR@thirdofthree#1#2#3{#3}%
975 \long\def\LWR@fourthoffour#1#2#3#4{#4}%
976
977 \long\def\LWR@firstoffive#1#2#3#4#5{#1}
978 \long\def\LWR@secondoffive#1#2#3#4#5{#2}
979 \long\def\LWR@thirdoffive#1#2#3#4#5{#3}
980 \long\def\LWR@fourthoffive#1#2#3#4#5{#4}
981 \long\def\LWR@fifthoffive#1#2#3#4#5{#5}
```

27.9   Inside boxes

Greater than zero if currently inside a TeX box, thus should not use \LWR@orignewpage.
See section 13.2.

```
982 \newcounter{LWR@texboxdepth}
983 \setcounter{LWR@texboxdepth}{0}
```

\LWR@maybe@orignewpage   Only do \LWR@orignewpage if not inside a TeX box.

```
984 \newcommand*{\LWR@maybe@orignewpage}{%
985     \LWR@traceinfo{LWR@maybe@orignewpage}%
986     \ifnumgreater{\value{LWR@texboxdepth}}{0}
987         {}%
988         {\LWR@orignewpage}%
989     \LWR@traceinfo{LWR@maybe@orignewpage done}%
990 }
```

27.10   Global boxes

\LWR@gsavebox {⟨*macroname*⟩} {⟨*contents*⟩}

From https://tex.stackexchange.com/questions/288702/
savebox-forgets-its-content-across-columns-inside-align

```
991 \DeclareRobustCommand\LWR@gsavebox[1]{%
992   \@ifnextchar(%
993     {\LWR@@gsavepicbox#1}{\@ifnextchar[{\LWR@@gsavebox#1}{\LWR@gsbox#1}}}%
994 \long\def\LWR@gsbox#1#2{\global\setbox#1\hbox{%
995   \color@setgroup#2\color@endgroup}}
996 \def\LWR@@gsavebox#1[#2]{%
997   \@ifnextchar [{\LWR@@igsavebox#1[#2]}{\LWR@@igsavebox#1[#2][c]}}
998 \long\def\LWR@@igsavebox#1[#2][#3]#4{%
999   \LWR@gsbox#1{\@imakebox[#2][#3]{#4}}}
1000 \def\LWR@@gsavepicbox#1(#2,#3){%
1001   \@ifnextchar[%
1002     {\LWR@@igsavepicbox#1(#2,#3)}{\LWR@@igsavepicbox#1(#2,#3)[]}}
1003 \long\def\LWR@@igsavepicbox#1(#2,#3)[#4]#5{%
1004   \LWR@gsbox#1{\@imakepicbox(#2,#3)[#4]{#5}}}
```

LWR@glrbox (*env.*) {⟨*macroname*⟩}

```
1005 \def\LWR@glrbox#1{%
1006   \edef\reserved@a{%
1007     \endgroup
1008     \global\setbox#1\hbox{%
1009       \begingroup\aftergroup}%
1010         \def\noexpand\@currenvir{\@currenvir}%
1011         \def\noexpand\@currenvline{\on@line}}%
1012   \reserved@a
1013     \@endpefalse
1014     \color@setgroup
1015       \ignorespaces}
1016 \let\LWR@endglrbox\LWR@endlrbox
```

27.11   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

```
1017 \newcommand*{\macrotocsname}[1]{%
1018   \ifcat\relax\noexpand#1%
1019     \expandafter\expandafter\expandafter\@gobble\expandafter\string
1020   \fi
1021   #1%
1022 }
```

27.12   Title case

\LWRtexttitlecase

```
1023 \ExplSyntaxOn
1024 \newcommand*{\LWRtexttitlecase}[1]{%
1025     \text_titlecase:n{#1}%
1026 }
1027 \ExplSyntaxOff
```

27.13   LetLtxMacrocs

\LWR@LetLtxMacrocs {⟨*newcsname*⟩} {⟨*oldcsname*⟩}

\LetLtxMacro with cs names.

```
1028 \newcommand*{\LWR@LetLtxMacrocs}[2]{%
1029     \expandafter\LetLtxMacro\csname #1\expandafter\endcsname%
1030     \csname#2\endcsname%
1031 }
```

27.14 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.

```
1032 \newcommand*{\LWR@absorbstar}[1]{%
1033     \LWR@LetLtxMacrocs{LWR@origns@#1}{#1}%
1034     \csdef{#1}{\@ifstar{\csuse{LWR@origns@#1}}{\csuse{LWR@origns@#1}}}
1035     \expandafter\robustify\csname #1\endcsname
1036 }
```

28 Operating-System portability

Unix (*Prog*)
Mac OS (*Prog*)
Linux (*Prog*)
MS-Windows (*Prog*)
Windows (*Prog*)
OSWindows (*Opt*)

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

If MS-Windows is not correctly detected, use the lwarp option OSWindows.

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.

28.1 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:

```
1037 \let\LWRpercent\@percentchar
```

The literal $ character:

```
1038 \catcode`\$=12
1039 \def\LWRdollar{$}
1040 \def\LWRdollar{$}% syntax highlighting
1041 \catcode`\$=3
```

The literal & character:

```
1042 \catcode`\&=12
1043 \def\LWRamp{&}
1044 \catcode`\&=4
```

The literal \ character. The ampersand is temporarily set to the escape character during the definition of the backslash macro.

```
1045 \catcode`\&=0
1046 &catcode`&\=12
1047 &def&LWRbackslash{\}
```

```
1048 &catcode'&\=0
1049 \catcode'\&=4
```

The literal { character. The ampersand is temporarily set to the begin group character during the definition of the leftbrace macro.

```
1050 \catcode'\&=1
1051 \catcode'\{=12
1052 \def\LWRleftbrace&{}
1053 \catcode'\{=1
1054 \catcode'\&=4
```

The literal } character. The ampersand is temporarily set to the end group character during the definition of the leftbrace macro.

```
1055 \catcode'\&=2
1056 \catcode'\}=12
1057 \def\LWRrightbrace{}&
1058 \catcode'\}=2
1059 \catcode'\&=4
```

The literal # character:

```
1060 \catcode'\#=12
1061 \def\LWRhash{#}
1062 \catcode'\#=6
```

\LWRopquote The operating system's quote mark, Unix default. For Windows, see \LWR@setOSWindows, below.

```
1063 \def\LWRopquote{'}
```

\LWRopseq The operating system's sequential execution command, Unix default. For Windows, see \LWR@setOSWindows, below.

```
1064 \def\LWRopseq{\space\LWRamp\LWRamp\space\space}
```

28.2   Common portability code

usingOSWindows (*bool*)   Set if the OSWindows option is used, or if Windows is automatically detected.

```
1065 \newbool{usingOSWindows}
1066 \boolfalse{usingOSWindows}
```

28.3   Unix, Linux, and Mac OS

\OSPathSymbol Symbol used to separate directories in a path.

```
1067 \newcommand*{\OSPathSymbol}{/}
```

28.4 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.

```
1068 \newcommand*{\LWR@setOSWindows}
1069 {
1070 \booltrue{usingOSWindows}
1071 \renewcommand*{\OSPathSymbol}{\@backslashchar}
1072 \def\LWRopquote{"}
1073 \def\LWRopseq{\space\LWRamp\space\space}
1074 }
```

Test for windows during compile. The user may also specify OSWindows package option in case this test fails.

```
1075 \ifwindows
1076 \LWR@setOSWindows
1077 \fi
```

29 Package options

kvoptions (*Pkg*)   Allows key/value package options.

```
1078 \RequirePackage{kvoptions}
1079 \SetupKeyvalOptions{family=LWR,prefix=LWR@}
```

\lwarpsetup A user interface to set the keys:

```
1080 \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.

```
1081 \newbool{warpingprint}
1082 \newbool{warpingHTML}
1083 \newbool{mathjax}
1084 \newbool{LWR@origmathjax}
```

defaults The default is print output, and svg math if the user chose HTML output.

```
1085 \booltrue{warpingprint}%
1086 \boolfalse{warpingHTML}%
1087 \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.

```
1088 \DeclareVoidOption{warpdisable}{%
1089     \PackageInfo{lwarp}{Using option 'warpdisable'}
1090     \boolfalse{warpingprint}%
1091     \boolfalse{warpingHTML}%
1092 }
```

warpprint (*Opt*)   If the warpprint option is given, boolean warpingprint is true and boolean warpingHTML is false, and may be used for \ifbool tests.

```
1093 \DeclareVoidOption{warpprint}{%
1094     \PackageInfo{lwarp}{Using option 'warpprint'}
1095     \booltrue{warpingprint}%
1096     \boolfalse{warpingHTML}%
1097 }
```

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.

```
1098 \DeclareVoidOption{warpHTML}{%
1099     \PackageInfo{lwarp}{Using option 'warpHTML'}%
1100     \booltrue{warpingHTML}%
1101     \boolfalse{warpingprint}%
1102 }
```

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.

```
1103 \DeclareVoidOption{mathsvg}{%
1104     \PackageInfo{lwarp}{Using option 'mathsvg'}
1105     \boolfalse{mathjax}%
1106     \boolfalse{LWR@origmathjax}%

1107 }
```

mathjax (*Opt*)   Option mathjax selects MATHJAX math display: If the mathjax option is given, boolean mathjax is true, may be used for \ifbool tests.

```
1108 \DeclareVoidOption{mathjax}{%
1109     \PackageInfo{lwarp}{Using option 'mathjax'}
1110     \booltrue{mathjax}%
1111     \booltrue{LWR@origmathjax}%

1112 }
```

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.

```
1113 \DeclareStringOption[\jobname]{BaseJobname}
```

ImagesDirectory (*Opt*)   Option ImagesDirectory sets the name of the directory to use for the lateximage
Default: \jobname-images   images.

```
1114 \DeclareStringOption[\BaseJobname-images]{ImagesDirectory}
```

ImagesName (*Opt*)   Option ImagesName sets the prefix to use for the lateximage images.
Default: image-

```
1115 \DeclareStringOption[image-]{ImagesName}
```

makeindexStyle (*Opt*)   Selects a custom .ist file. A customized file should be based on lwarp.ist. See
Default: lwarp.ist   section 8.6.20.

```
1116 \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 8.6.21.

```
1117 \DeclareStringOption[lwarp.xdy]{xindyStyle}
```

xindyLanguage (*Opt*)   Sets the *xindy* language to be assigned in *lwarpmk*'s configuration files. This is
Default: english   then used by *lwarpmk* while processing the index and glossary.

```
1118 \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.

```
1119 \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 8.6.22.

```
1120 \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.

```
1121 \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:

```
1122 \newbool{LWR@creatinglwarpmk}
1123 \boolfalse{LWR@creatinglwarpmk}
1124
1125 \DeclareVoidOption{lwarpmk}{
1126     \PackageInfo{lwarp}{Using option 'lwarpmk'}
1127     \booltrue{LWR@creatinglwarpmk}
1128 }
```

OSWindows (*Opt*)   Tells lwarp to use MS-WINDOWS compatibility. Auto-detection of the operating system 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.

```
1129 \DeclareVoidOption{OSWindows}{
1130     \PackageInfo{lwarp}{Using option 'OSWindows'}
1131     \LWR@setOSWindows
1132 }
```

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.

```
1133 \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.

```
1134 \DeclareStringOption[]{HTMLFilename}
```

PrintLatexCmd (*Opt*)   The shell commands to use to compile the print document.
Default: <automatic>
```
1135 \DeclareStringOption[]{PrintLatexCmd}
```

HTMLLatexCmd (*Opt*)   The shell commands to use to compile the HTML document.
Default: <automatic>
```
1136 \DeclareStringOption[]{HTMLLatexCmd}
```

PrintIndexCmd (*Opt*)   The shell commands to use to compile the print indexes.
Default: <empty>
```
1137 \DeclareStringOption[]{PrintIndexCmd}
```

HTMLIndexCmd (*Opt*)   The shell commands to use to compile the HTML indexes.
Default: <empty>
```
1138 \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*.

```
1139 \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.

```
1140 \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.

```
1141 \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.

1142 \DeclareBoolOption[false]{xindex}

IndexRef (*Opt*) Tells lwarp how to display the index entries in HTMLoutput. See section 7.5.
Default: cref
1143 \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.

1144 \DeclareStringOption[makeglossaries]{GlossaryCmd}

latexmk (*Opt*) Option latexmk tells *lwarpmk* to use *latexmk* when compiling documents.

1145 \DeclareBoolOption[false]{latexmk}

dvips (*Opt*) Option dvips tells *lwarpmk* to use *dvips* when compiling DVI *latex* documents.

1146 \DeclareBoolOption[false]{dvips}

dvipdfm (*Opt*) Option dvipdfm tells *lwarpmk* to use *dvipdfm* when compiling DVI *latex* documents.

1147 \DeclareBoolOption[false]{dvipdfm}

dvipdfmx (*Opt*) Option dvipdfmx tells *lwarpmk* to use *dvipdfmx* when compiling DVI *latex* documents.

1148 \DeclareBoolOption[false]{dvipdfmx}

Execute options Execute the package options, with the defaults which have been set just above:

1149 \ProcessKeyvalOptions*\relax

29.1 Additional options support

Assign the \BaseJobname if the user hasn't provided one:

1150 \providecommand*{\BaseJobname}{\LWR@BaseJobname}

Defaults unless already over-ridden by the user:

```
1151 \ifcsempty{LWR@HomeHTMLFilename}{
1152     \newcommand*{\HomeHTMLFilename}{\BaseJobname}
1153 }{
1154     \csedef{HomeHTMLFilename}{\LWR@HomeHTMLFilename}
1155 }
1156
1157 \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.

```
1158 \newcommand*{\LWR@sanitized}{}
```

\LWR@sanitize {⟨*text*⟩}

Sanitizes the text and returns the result in \LWR@sanitized.

```
1159 \newcommand*{\LWR@sanitize}[1]{%
1160 \edef\LWR@sanitized{#1}%
1161 \edef\LWR@sanitized{\detokenize\expandafter{\LWR@sanitized}}%
1162 }
```

Sanitize some string options to neutralize underscores.

```
1163 \LWR@sanitize{\LWR@BaseJobname}
1164 \edef\LWR@BaseJobname{\LWR@sanitized}
1165
1166 \LWR@sanitize{\LWR@ImagesDirectory}
1167 \edef\LWR@ImagesDirectory{\LWR@sanitized}
1168
1169 \LWR@sanitize{\LWR@ImagesName}
1170 \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.

```
1171 \ifdefempty{\LWR@PrintIndexCmd}{
1172    \renewcommand{\LWR@PrintIndexCmd}{%
1173        makeindex  -s \LWR@makeindexStyle \space  \jobname.idx%
1174    }
1175    \ifbool{LWR@xindy}{
1176        \renewcommand{\LWR@PrintIndexCmd}{%
1177            xindy
1178            -M \LWR@xindyStyle \space
1179            -L \LWR@xindyLanguage \space
1180            -C \LWR@xindyCodepage \space
1181            \jobname.idx%
1182        }
1183    }{}
1184    \ifbool{LWR@xindex}{
1185        \ifdefvoid{\LWR@xindexConfig}{
1186            \renewcommand{\LWR@PrintIndexCmd}{%
1187                xindex
1188                \jobname.idx%
1189            }
1190        }{
1191            \renewcommand{\LWR@PrintIndexCmd}{%
1192                xindex
1193                -c \LWR@xindexConfig \space
1194                \jobname.idx%
1195            }
1196        }
1197    }{}
1198 }{}
```

```
1199
1200 \ifdefempty{\LWR@HTMLIndexCmd}{
1201     \renewcommand{\LWR@HTMLIndexCmd}{%
1202         makeindex  -s \LWR@makeindexStyle \space  \jobname_html.idx%
1203     }
1204     \ifbool{LWR@xindy}{
1205         \renewcommand{\LWR@HTMLIndexCmd}{%
1206             xindy
1207             -M \LWR@xindyStyle \space
1208             -L \LWR@xindyLanguage \space
1209             -C \LWR@xindyCodepage \space
1210             \jobname_html.idx%
1211         }
1212     }{}
1213     \ifbool{LWR@xindex}{
1214         \ifdefvoid{\LWR@xindexConfig}{
1215             \renewcommand{\LWR@HTMLIndexCmd}{%
1216                 xindex
1217                 \jobname_html.idx%
1218             }
1219         }{
1220             \renewcommand{\LWR@HTMLIndexCmd}{%
1221                 xindex
1222                 -c \LWR@xindexConfig \space
1223                 \jobname_html.idx%
1224             }
1225         }
1226     }{}
1227 }{}
1228
1229 \ifdefempty{\LWR@LatexmkIndexCmd}{
1230     \renewcommand{\LWR@LatexmkIndexCmd}{%
1231         makeindex  -s \LWR@makeindexStyle%
1232     }
1233     \ifbool{LWR@xindy}{
1234         \renewcommand{\LWR@LatexmkIndexCmd}{%
1235             xindy
1236             -M \LWR@xindyStyle \space
1237             -L \LWR@xindyLanguage \space
1238             -C \LWR@xindyCodepage%
1239         }
1240     }{}
1241     \ifbool{LWR@xindex}{
1242         \ifdefvoid{\LWR@xindexConfig}{
1243             \renewcommand{\LWR@LatexmkIndexCmd}{%
1244                 xindex
1245             }
1246         }{
1247             \renewcommand{\LWR@LatexmkIndexCmd}{%
1248                 xindex
1249                 -c \LWR@xindexConfig
1250             }
1251         }
1252     }{}
1253 }{}
```

29.2 Conditional compilation

\warpprintonly {⟨*contents*⟩}

    Only process the contents if producing printed output.

```
1254 \newcommand{\warpprintonly}[1]{\ifbool{warpingprint}{#1}{}}
```

\warpHTMLonly {⟨*contents*⟩}

    Only process the contents if producing HTML output.

```
1255 \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.

```
1256 \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.

```
1257 \def\LWR@includecomment
1258 #1#2{\message{Lwarp: Including comment '#1'}%
1259     \csarg\def{After#1Comment}{%
1260         \CloseAndInputCutFile%
1261         \csundef{LWR@#1commentused}%
1262     }
1263     \csarg\def{#1}{%
1264         \endgroup
1265         \ifcsdef{LWR@#1commentused}{
1266             \PackageError{lwarp}%
1267                 {Nested #1 environment}%
1268                 {%
1269                     Environment #1 cannot be nested.\MessageBreak
1270                     This can happen when a package is loaded
1271                     from inside a\MessageBreak
1272                     #1 environment.%
1273                 }%
1274         }{\relax}
1275         \csdef{LWR@#1commentused}{}
1276         \message{Including '#1' comment.}%
1277         \def\CommentCutFile{comment_#2.cut}
1278         \SetUpCutFile
1279         \ProcessComment{#1}
1280     }%
1281     \CommentEndDef{#1}
1282 }
1283
1284 \def\LWR@excludecomment
```

```
1285 #1#2{\message{Lwarp: Excluding comment '#1'}%
1286    \csarg\def{#1}{
1287       \endgroup
1288       \message{Excluding '#1' comment.}%
1289       \begingroup
1290         \def\CommentCutFile{comment_#2.cut}
1291          \def\ProcessCutFile{}%
1292         \def\ThisComment####1{}%
1293          \ProcessComment{#1}
1294    }%
1295    \csarg\def{After#1Comment}{\CloseAndInputCutFile \endgroup}
1296    \CommentEndDef{#1}}
```

warpall (*env.*) Anything in the warpall environment will be generated for print or HTML outputs.

```
1297 \LWR@includecomment{warpall}{all}
```

warpHTML (*env.*) For HTML output:

```
1298 \ifbool{warpingHTML}
1299    {\LWR@includecomment{warpHTML}{html}}
1300    {\LWR@excludecomment{warpHTML}{html}}
```

warpprint (*env.*) Anything in the warpprint environment will be generated for print output only.

```
1301 \ifbool{warpingprint}
1302    {\LWR@includecomment{warpprint}{print}}
1303    {\LWR@excludecomment{warpprint}{print}}
```

If warpdisable, turn off both print and HTML output:

```
1304 \ifboolexpr{bool {warpingprint} or bool {warpingHTML}}
1305    {}
1306    {
1307       \LWR@excludecomment{warpHTML}{html}
1308       \LWR@excludecomment{warpprint}{print}
1309       \LWR@excludecomment{warpMathJax}{mathjax}
1310    }
```

warpMathJax (*env.*) Only if MathJax is being used along with HTML.

```
1311 \begin{warpprint}
1312 \LWR@excludecomment{warpMathJax}{mathjax}
1313 \end{warpprint}
1314
1315 \begin{warpHTML}
1316 \ifbool{mathjax}
1317    {\LWR@includecomment{warpMathJax}{mathjax}}
1318    {\LWR@excludecomment{warpMathJax}{mathjax}}
1319 \end{warpHTML}
```

warpsvg (*env.*) Only if SVG math is being used along with HTML, or in print mode.

```
1320 \begin{warpprint}
1321 \LWR@includecomment{warpsvg}{mathsvg}
```

```
1322 \end{warpprint}
1323
1324 \begin{warpHTML}
1325 \ifbool{mathjax}
1326     {\LWR@excludecomment{warpsvg}{mathsvg}}
1327     {\LWR@includecomment{warpsvg}{mathsvg}}
1328 \end{warpHTML}
```

LWRcreatelwarpmk (*env.*) Optionally generate a local copy of *lwarpmk*. Default to no.

```
1329 \ifbool{LWR@creatinglwarpmk}
1330     {\LWR@includecomment{LWRcreatelwarpmk}{lwarpmk}}
1331     {\LWR@excludecomment{LWRcreatelwarpmk}{lwarpmk}}
```

30 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: `1332 \begin{warpHTML}`

fontspec (*Pkg*)   Load fontspec if necessary:

```
1333 \ifxetexorluatex
1334 \IfPackageLoadedTF{fontspec}{}{
1335     \usepackage[no-math]{fontspec}
1336 }
```

The monospaced font is used for HTML tags, so turn off its TeX ligatures and common ligatures:

```
1337 \defaultfontfeatures[\rmfamily]{Ligatures={NoCommon,TeX}}
1338 \defaultfontfeatures[\sffamily]{Ligatures={NoCommon,TeX}}
1339 \defaultfontfeatures[\ttfamily]{Ligatures=NoCommon}
1340 \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.

```
1341 \RequirePackage{microtype}
1342
1343 \microtypesetup{
1344     protrusion=false,
1345     expansion=false,
1346     tracking=false,
1347     kerning=false,
1348     spacing=false}
1349 %    \begin{macrocode}
1350 %
```

```
1351 % Disable ligatures for typewriter fonts.
1352 % The comma was causing issues with \MathJax\ and \cs{,} followed by a comma.
1353 % Ligatures for f, q, t, etc used to be disabled for non-typewriter fonts, but
1354 % are now allowed.
1355 % \changes{v0.89}{2020/08/01}{Disable typewriter ligatures.}
1356 % ^^A \DisableLigatures[{,},f,q,t,T,Q]{encoding = *,family = *}% previous
1357 %     \begin{macrocode}
1358 \DisableLigatures{encoding = *,family = tt*}
```

```
1359 \fi
```

```
1360 \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 overflow.

- 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.

1361 \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.

```
1362 \IfPackageLoadedTF{geometry}
1363 {}{
1364     \RequirePackage[
1365         reset,
1366         paperwidth=\paperwidth,
1367         paperheight=\paperheight,
1368         textwidth=\textwidth,
1369         textheight=\textheight,
1370         left=\oddsidemargin,
1371         top=\topmargin,
1372         marginparsep=\marginparsep,
1373         marginparwidth=\marginparwidth,
1374     ]{geometry}
1375 }
```

Remember the original definitions for later reuse. If the **geometry** package is loaded by the user, **lwarp-geometry** will nullify the user-level originals.

```
1376 \LetLtxMacro\LWR@origgeometry\geometry
1377 \LetLtxMacro\LWR@orignewgeometry\newgeometry
1378 \LetLtxMacro\LWR@origrestoregeometry\restoregeometry
1379 \LetLtxMacro\LWR@origsavegeometry\savegeometry
1380 \LetLtxMacro\LWR@origloadgeometry\loadgeometry
```

LWR@allowanothergeometry   **geometry** may be loaded by the user before lwarp, after lwarp, or not at all. If
(*bool*)   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.

```
1381 \newbool{LWR@allowanothergeometry}
1382 \booltrue{LWR@allowanothergeometry}
```

Use \AtEndPreamble to avoid class and option conflict by changing settings after
other packages load, instead of using geometry package options:

```
1383 \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.

```
1384 \LWR@origsavegeometry{LWR@usergeometry}
```

The user's paper size is saved for later reuse, such as by the pdfpages or parallel
packages.

```
1385 \newlength{\LWR@userspaperwidth}
1386 \setlength{\LWR@userspaperwidth}{\paperwidth}
1387
1388 \newlength{\LWR@userspaperheight}
1389 \setlength{\LWR@userspaperheight}{\paperheight}
1390
1391 \newlength{\LWR@usersmarginparwidth}
1392 \setlength{\LWR@usersmarginparwidth}{\marginparwidth}
1393
1394 \newlength{\LWR@userstextwidth}
1395 \setlength{\LWR@userstextwidth}{\textwidth}
1396
1397 \newlength{\LWR@userstextheight}
1398 \setlength{\LWR@userstextwidth}{\textheight}
```

For lwarp, use a very large page and margins to help avoid letting HTML tags run
off the edge:

```
1399 \LWR@origgeometry{
1400     reset,
1401     paperheight=190in,
1402     paperwidth=20in,
1403     left=2in,
1404     right=6in,
1405     top=1in,
1406     bottom=1in,
1407     heightrounded,%
1408 }
```

The lwarp page geometry is saved for future restore:

```
1409 \LWR@origsavegeometry{LWR@lwarpgeometry}
```

No longer adjust the page layout when lwarp-geometry is loaded \AtBeginDocument:

```
1410 \boolfalse{LWR@allowanothergeometry}%
```

ltjsbook and other classes can print vertically, and require these to be reset by lwarp:

```
1411 \setlength{\textheight}{0.8\paperheight}
1412 \setlength{\textwidth}{0.7\paperwidth}
1413
1414 \@twosidefalse
1415 \@mparswitchfalse
1416 }% \AtEndPreamble
1417
1418 \end{warpHTML}
```

for HTML & PRINT: `1419 \begin{warpall}`

xparse (*Pkg*)

LATEX3 command argument parsing

```
1420 \RequirePackage{xparse}
```

calc (*Pkg*)

```
1421 \RequirePackage{calc}
```

```
1422 \end{warpall}
```

for HTML output: `1423 \begin{warpHTML}`

expl3 (*Pkg*)

LATEX3 programming

```
1424 \RequirePackage{expl3}
```

gettitlestring (*Pkg*)

Used to emulate \nameref.

```
1425 \RequirePackage{gettitlestring}
1426
1427
1428 \end{warpHTML}
```

for HTML & PRINT: `1429 \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.

```
1430 \@ifundefined{filec@ntents@opt}{% older kernel, discard optional args
1431
1432     \RequirePackage{filecontents}
1433
1434     \LetLtxMacro\LWR@orig@filec@ntents\filec@ntents
1435
1436     \IfPackageAtLeastTF{filecontents}{2011/10/08}
1437     {
```

For a newer version of the filecontents package, simply discard the optional argument.

```
1438         \renewcommand*{\filec@ntents}[1][]{\LWR@orig@filec@ntents}
1439     }
1440     {% 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

```
1441         \newwrite\fcwrite
1442         \renewcommand*{\filec@ntents}[1][]{%
1443             \def\chardef##1\write{\let\reserved@c\fcwrite}%
1444             \LWR@orig@filec@ntents%
1445         }
1446     }
1447
1448 }% older kernel
1449 {% newer kernel
```

For a newer kernel with a `filecontents` environment which accepts the optional overwrite argument, use the environment as-is.

```
1450 }% newer kernel, filecontents env accepts optional args, do not load package
```

```
1451 \end{warpall}
```

`1452 \begin{warpHTML}`

xifthen (*Pkg*)

`1453 \RequirePackage{xifthen}`

verbatim (*Pkg*)

`1454 \RequirePackage{verbatim}`

refcount (*Pkg*)

Provides \setcounterref, \setcounterpageref, etc.

`1455 \RequirePackage{refcount}`

newfloat (*Pkg*)

```
1456 \RequirePackage{newfloat}
```

```
1457 \end{warpHTML}
```

for HTML & PRINT: 1458 `\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.

```
1459 \RequirePackage{xstring}[2019/02/01]
```

environ (*Pkg*)   Used to encapsulate math environments for re-use in HTML <alt> text.

```
1460 \RequirePackage{environ}
```

```
1461 \end{warpall}
```

for HTML output: 1462 `\begin{warpHTML}`

printlen (*Pkg*)   Used to convert lengths for image width/height options.

```
1463 \RequirePackage{printlen}
```

\LWR@printlength {⟨*length*⟩}

Prints a length using a locally-controlled unit and space. Rounding is used unless
the length is small.

```
1464 \newrobustcmd*{\LWR@printlength}[1]{%
1465     \begingroup%
1466     \uselengthunit{PT}%
1467     \renewcommand*{\unitspace}{}%
1468     \ifdimless{#1}{10pt}{%
1469         \printlength{#1}%
1470     }{%
1471         \rndprintlength{#1}%
1472     }%
1473     \endgroup%
1474 }
```

```
1475 \end{warpHTML}
```

31 Loading packages

\RequirePackage and \usepackage are modified to error-check for certain pack-
ages, and for HTML they load the lwarp- version if it exists.

for HTML & PRINT: 1476 `\begin{warpall}`

Remember the original \RequirePackage:

```
1477 \LetLtxMacro\LWR@origRequirePackage\RequirePackage
1478 \LetLtxMacro\LWR@origRequirePackageWithOptions\RequirePackageWithOptions
```

`\LWR@requirepackagenames` Stores the list of required package names.

```
1479 \newcommand*{\LWR@requirepackagenames}{}
```

`\LWR@parsedrequirepackagenames` Stores the parsed list of required package names after spaces are removed and lwarp- is prepended.

```
1480 \newcommand*{\LWR@parsedrequirepackagenames}{}
```

`\LWR@nullifycomment` Remove the preexisting comment environment. Certain packages define it for their own use.

```
1481 \newcommand*{\LWR@nullifycomment}{%
1482     \PackageInfo{lwarp}%
1483       {Nullifying the comment environment before loading \LWR@strresulttwo,}%
1484     \let\comment\relax%
1485     \let\endcomment\relax%
1486 }
```

`\LWR@findword` [⟨*1: separator*⟩] {⟨*2: list*⟩} {⟨*3: index*⟩} [⟨*4: destination*⟩]

Note that argument 4 is passed directly to `\StrBetween`.

```
1487 \newcommand*\LWR@findword[3][,]{%
1488     \StrBetween[#3,\numexpr#3+1]{#1#2#1}{#1}{#1}%
1489 }
```

`\LWR@checkloadnever` {⟨*bad package name*⟩} {⟨*replacement package names*⟩}

From now on, check for incompatible packages loaded via \usepackage, instead of packages loaded before lwarp:

```
1490 \LetLtxMacro\LWR@checkloadnever\LWR@afterloadnever
```

`\LWR@checkloadfilename` {⟨*filename*⟩} Checks if this filename should be loaded after lwarp, or never at all.

```
1491 \newcommand*{\LWR@checkloadfilename}[1]{%
```

Remember the package name to compare with, to be used by \LWR@checkloadnever and \LWR@checkloadbefore.

```
1492     \edef\LWR@tempone{#1}%
```

Check against the list of packages which should never be loaded:

```
1493     \LWR@checkloadnevers
```

The following should only be loaded before lwarp:

```
1494     \LWR@checkloadbefore{ctex}
1495     \LWR@checkloadbefore{fontspec}
1496     \LWR@checkloadbefore{inputenc}
1497     \LWR@checkloadbefore{inputenx}
1498     \LWR@checkloadbefore{nfssext-cfr}
```

```
1499      \LWR@checkloadbefore{fontaxes}
1500      \LWR@checkloadbefore{kotex}
1501      \LWR@checkloadbefore{kpfonts}% textcomp option clash
1502      \LWR@checkloadbefore{luatexja}
1503      \LWR@checkloadbefore{luatexja-fontspec}
1504      \LWR@checkloadbefore{luatexko}
1505      \LWR@checkloadbefore{morewrites}
1506      \LWR@checkloadbefore{newclude}
1507      \LWR@checkloadbefore{newunicodechar}
1508      \LWR@checkloadbefore{plext}
1509      \LWR@checkloadbefore{xeCJK}
1510      \LWR@checkloadbefore{xetexko}
1511      \LWR@checkloadbefore{zxjatype}
1512 }
```

\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.

```
1513 \newcommand*{\LWR@lookforpackagename}[1]{%
```

> Find the index'th package name from the list:

```
1514 \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.

```
1515 \StrSubstitute[100]{\LWR@strresult}{ }{}[\LWR@strresulttwo]%
```

> See if the package name was found:

```
1516 \IfStrEq{\LWR@strresulttwo}{}%
1517 {}% no filename
1518 {% yes filename was found
```

> Possible adjustments before loading the package. Maybe nullify the comment envi-
> ronment if the new package will be redefining it for a new purpose.

```
1519      \ifdefstring{\LWR@strresulttwo}{easyReview}{\LWR@nullifycomment}{}%
1520      \ifdefstring{\LWR@strresulttwo}{changes}{\LWR@nullifycomment}{}%
```

> If HTML, check if the package should be loaded before lwarp, or never at all:

```
1521      \ifbool{warpingHTML}{\LWR@checkloadfilename{\LWR@strresulttwo}}{}%
```

> If HTML, and if found, and if an lwarp-equivalent name exists, use lwarp-* instead.

```
1522      \ifboolexpr{
1523          bool{warpingHTML} and
1524          test{\IfFileExists{lwarp-\LWR@strresulttwo.sty}}
1525      }%
1526      {% lwarp-* file found
1527          \ifdefvoid{\LWR@parsedrequirepackagenames}{%
```

```
1528              \edef\LWR@parsedrequirepackagenames{lwarp-\LWR@strresulttwo}%
1529          }{%
1530              \edef\LWR@parsedrequirepackagenames{%
1531                  \LWR@parsedrequirepackagenames,lwarp-\LWR@strresulttwo%
1532              }%
1533          }%
1534      }%
1535      {%
```

Otherwise, use the current package name.

```
1536          \ifdefvoid{\LWR@parsedrequirepackagenames}{%
1537              \edef\LWR@parsedrequirepackagenames{\LWR@strresulttwo}%
1538          }{%
1539              \edef\LWR@parsedrequirepackagenames{%
1540                  \LWR@parsedrequirepackagenames,\LWR@strresulttwo%
1541              }%
1542          }%
1543      }% no lwarp-* file
1544 }% yes filename
1545 }
```

\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.

```
1546 \RenewDocumentCommand{\RequirePackage}{o m o}{%
```

Redirect up to twenty names:[17]

```
1547 \renewcommand*{\LWR@requirepackagenames}{#2}%
1548 \renewcommand*{\LWR@parsedrequirepackagenames}{}%
1549 \LWR@lookforpackagename{1}%
1550 \LWR@lookforpackagename{2}%
1551 \LWR@lookforpackagename{3}%
1552 \LWR@lookforpackagename{4}%
1553 \LWR@lookforpackagename{5}%
1554 \LWR@lookforpackagename{6}%
1555 \LWR@lookforpackagename{7}%
1556 \LWR@lookforpackagename{8}%
1557 \LWR@lookforpackagename{9}%
1558 \LWR@lookforpackagename{10}%
1559 \LWR@lookforpackagename{11}%
1560 \LWR@lookforpackagename{12}%
1561 \LWR@lookforpackagename{13}%
1562 \LWR@lookforpackagename{14}%
1563 \LWR@lookforpackagename{15}%
1564 \LWR@lookforpackagename{16}%
1565 \LWR@lookforpackagename{17}%
1566 \LWR@lookforpackagename{18}%
1567 \LWR@lookforpackagename{19}%
1568 \LWR@lookforpackagename{20}%
```

Error if braces are used in optional argument. This can cause an error, so tell how
to avoid.

---

[17]This was originally nine names, but then I came across a package which used twelve...

```
1569 \IfSubStr{\detokenize\expandafter{#1}}{\LWRleftbrace}%
1570     {%
1571         \PackageError{lwarp}{%
1572             You used:\MessageBreak
1573             \protect\usepackage[#1]{#2}\MessageBreak
1574             Braces in the package options will fail with Lwarp.\MessageBreak
1575             Instead, use:\MessageBreak
1576             \protect\PassOptionsToPackage{#1}{#2}\MessageBreak
1577             \protect\usepackage{#2}\MessageBreak
1578             near the line number given below.\MessageBreak
1579             Enter 'h' for more info%
1580         }%
1581         {%
1582             See the Lwarp manual troubleshooting index entry for\MessageBreak
1583             ``package, options with braces''%
1584         }%
1585     }%
1586     {}% no brace
```

\RequirePackage depending on the options and version:

```
1587 \IfValueTF{#1}%
1588 {% options given
1589     \IfValueTF{#3}% version given?
1590         {\LWR@origRequirePackage[#1]{\LWR@parsedrequirepackagenames}[#3]}%
1591         {\LWR@origRequirePackage[#1]{\LWR@parsedrequirepackagenames}}%
1592 }%
1593 {% no options given
1594     \IfValueTF{#3}% version given?
1595         {\LWR@origRequirePackage{\LWR@parsedrequirepackagenames}[#3]}%
1596         {\LWR@origRequirePackage{\LWR@parsedrequirepackagenames}}%
1597 }%
1598 }
1599 \LetLtxMacro\usepackage\RequirePackage
1600 \@onlypreamble\RequirePackage
1601 \@onlypreamble\usepackage
```

```
1602 \end{warpall}
```

for HTML output: 1603 \begin{warpHTML}

\LWR@ProvidesPackagePass {⟨*pkgname*⟩} [⟨*version*⟩]

Uses the original package, including options.

```
1604 \NewDocumentCommand{\LWR@ProvidesPackagePass}{m o}{
1605     \PackageInfo{lwarp}{%
1606         Using package '#1',\MessageBreak
1607         and adding lwarp modifications, including options,\MessageBreak%
1608     }%
1609     \IfValueTF{#2}%
1610         {\ProvidesPackage{lwarp-#1}[#2]}%
1611         {\ProvidesPackage{lwarp-#1}}%
1612     \DeclareOption*{%
1613         \PassOptionsToPackage{\CurrentOption}{#1}%
1614     }%
1615     \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.

```
1616        \IfPackageLoadedTF{#1}{%
1617            \edef\LWR@tempone{\csuse{opt@#1.sty}}%
1618            \IfValueTF{#2}%
1619                {%
1620                    \expandafter\LWR@origRequirePackage%
1621                        \expandafter[\LWR@tempone]{#1}[#2]%
1622                }%
1623                {%
1624                    \expandafter\LWR@origRequirePackage%
1625                        \expandafter[\LWR@tempone]{#1}%
1626                }%
1627        }{%
1628            \IfValueTF{#2}%
1629                {\LWR@origRequirePackage{#1}[#2]}%
1630                {\LWR@origRequirePackage{#1}}%
1631        }%
```

In some cases, the following seems to be required to avoid an "unknown option" error, such as when loading xcolor with options.

```
1632        \DeclareOption*{}%
1633        \ProcessOptions\relax%
1634 }
```

\LWR@ProvidesPackageDropA {⟨*name*⟩} {⟨*date or -NoValue-*⟩}

Declares the package. Factored for reuse.

```
1635 \newcommand*{\LWR@ProvidesPackageDropA}[2]{%
1636     \PackageInfo{lwarp}{%
1637         Replacing package '#1' with the lwarp version,\MessageBreak
1638         and discarding options,%
1639     }%
1640     \IfValueTF{#2}
1641     {\ProvidesPackage{lwarp-#1}[#2]}
1642     {\ProvidesPackage{lwarp-#1}}
1643 }
```

\LWR@ProvidesPackageDropB Nullifies then processes the options.

Seems to be required when options contain curly braces, which were causing "Missing \begin{document}".

```
1644 \newcommand*{\LWR@ProvidesPackageDropB}{%
1645 % \ProcessOptions\relax% original LaTeX code
1646 \let\ds@\@empty%          from the original \ProcessOptions
1647 \edef\@curroptions{}%     lwarp modification to \ProcessOptions
1648 \@process@ptions\relax%   from the original \ProcessOptions
1649 }
```

\LWR@ProvidesPackageDrop {⟨*pkgname*⟩} [⟨*version*⟩]

Ignores the original package and uses lwarp's version instead. Drops/discards all options.

```
1650 \NewDocumentCommand{\LWR@ProvidesPackageDrop}{m o}{
```

Declare the package:

```
1651 \LWR@ProvidesPackageDropA{#1}{#2}
```

Ignore all options:

```
1652 \DeclareOption*{}
```

Process the options:

```
1653 \LWR@ProvidesPackageDropB
1654 }
```

```
1655 \end{warpHTML}
```

32   File handles

Defines file handles for writes.

for HTML & PRINT:
```
1656 \begin{warpall}
```

`\LWR@quickfile`   For quick temporary use only. This is reused in several places.

```
1657 \newwrite\LWR@quickfile%
```

```
1658 \end{warpall}
```

for HTML output:
```
1659 \begin{warpHTML}
```

`\LWR@lateximagesfile`   For <project>-images.txt:

```
1660 \newwrite\LWR@lateximagesfile
```

```
1661 \end{warpHTML}
```

33   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.

1662 `\begin{warpHTML}`

`\@include` {⟨*filename*⟩} Modified to load _html_inc files.

(Below, `\clearpage` caused missing text, and was changed to `\newpage`.)

```
1663 \def\@include#1 {%
1664 \immediate\openout\LWR@quickfile #1_html_inc.tex% lwarp
1665 \immediate\write\LWR@quickfile{\string\input{#1.tex}}% lwarp
1666 \immediate\closeout\LWR@quickfile% lwarp
1667 \LWR@maybe@orignewpage% changed from clearpage
1668 \if@filesw
1669     \immediate\write\@mainaux{\string\@input{#1_html_inc.aux}}% changed
1670 \fi
1671 \@tempswatrue
1672 \if@partsw
1673     \@tempswafalse
1674     \edef\reserved@b{#1}%
1675     \@for\reserved@a:=\@partlist\do
1676     {\ifx\reserved@a\reserved@b\@tempswatrue\fi}%
1677 \fi
1678 \if@tempswa
1679     \let\@auxout\@partaux
1680     \if@filesw
1681         \immediate\openout\@partaux #1_html_inc.aux % changed
1682         \immediate\write\@partaux{\relax}%
1683     \fi
1684     \@input@{#1_html_inc.tex}% changed
1685     \LWR@maybe@orignewpage% changed from clearpage
1686     \@writeckpt{#1}%
1687     \if@filesw
1688         \immediate\closeout\@partaux
1689     \fi
1690 \else
1691     \deadcycles\z@
1692     \@nameuse{cp@#1}%
1693 \fi
1694 \let\@auxout\@mainaux%
1695 }
```

```
1696 \end{warpHTML}
```

34   Copying a file

1697 `\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.

```
1698 \newwrite\LWR@copyoutfile   % open the file to write to
1699 \newread\LWR@copyinfile     % open the file to read from
1700
1701 \newcommand*{\LWR@copyfile}[2]{%
1702     \LWR@traceinfo{LWR@copyfile: copying #1 to #2}
```

```
1703
1704     \immediate\openout\LWR@copyoutfile=#2
1705     \openin\LWR@copyinfile=#1
1706     \begingroup\endlinechar=-1
1707     \makeatletter
1708
1709     \LWR@traceinfo{LWR@copyfile: about to loop}
1710
1711     \loop\unless\ifeof\LWR@copyinfile
1712         \LWR@traceinfo{LWR@copyfile: one line}
1713     \read\LWR@copyinfile to\LWR@fileline  % Read one line and store it into \LWR@fileline
1714 %   \LWR@fileline\par                     % print the content into the pdf
1715 % print the content:
1716       \immediate\write\LWR@copyoutfile{\unexpanded\expandafter{\LWR@fileline}}%
1717     \repeat
1718     \immediate\closeout\LWR@copyoutfile
1719     \LWR@traceinfo{LWR@copyfile: done}
1720     \endgroup
1721 }
```

```
1722 \end{warpHTML}
```

35   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:
```
1723 \begin{warpall}
```

LWR@tracinglwarp (*bool*)   True if tracing is turned on.

```
1724 \newbool{LWR@tracinglwarp}
```

\tracinglwarp   Turns on the debug tracing messages.

```
1725 \newcommand{\tracinglwarp}{\booltrue{LWR@tracinglwarp}}
```

\LWR@traceinfo {⟨*text*⟩}   If tracing is turned on, writes the text to the .log file.

```
1726 \newcommand{\LWR@traceinfo}[1]{%
1727 \ifbool{LWR@tracinglwarp}%
1728 {%
1729     \typeout{*** lwarp: #1}%
1730 }%
1731 {}%
1732 }
```

HTMLDebugComments (*bool*)   Add comments in HTML about closing <div>s, sections, etc.
Default: false

```
1733 \newbool{HTMLDebugComments}
1734 \boolfalse{HTMLDebugComments}
```

If \tracinglwarp, show where preamble hooks occur:

```
1735 \AfterEndPreamble{
1736 \LWR@traceinfo{AfterEndPreamble}
1737 }
1738
1739 \AtBeginDocument{
1740 \LWR@traceinfo{AtBeginDocument}
1741 }
```

```
1742 \end{warpall}
```

36 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" error may occur.

⚠ Misplaced \omit error

```
\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: 1743 \begin{warpall}

1744 \newcommand*{\LWR@formatted}[1]{}
1745 \newcommand*{\LWR@expandableformatted}[1]{}
1746 \newcommand*{\LWR@formattedenv}[1]{}
1747 \newcommand*{\LWR@expandableformattedenv}[1]{}

1748 \end{warpall}

for HTML output: HTML versions:

1749 \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".

1750 \newcommand*{\LWR@formatting}{HTML}

\LWR@formatted@checkname {⟨name⟩}

1751 \newcommand*{\LWR@formatted@checkname}[1]{%
1752   \ifcsundef{#1}{%
1753   \ifcsundef{LWR@print@#1}{%
1754   \PackageError{lwarp}
1755   {%
1756   \LWRbackslash#1 or \protect\LWR@print@#1\MessageBreak
1757   must be defined before using \protect\LWR@formatted, etc%
1758   }
1759   {Perhaps #1 is misspelled.}
1760   }{\relax}%
1761   }{\relax}%
1762   \ifcsundef{LWR@HTML@#1}{%
1763   \PackageError{lwarp}
1764   {%
1765   \protect\LWR@HTML@#1 must be defined
1766   before using \protect\LWR@formatted, etc%
1767   }
1768   {Perhaps #1 is misspelled.}
1769   }{\relax}%
1770 }

\LWR@formatted@checkendname {⟨name⟩}

1771 \newcommand*{\LWR@formatted@checkendname}[1]{%
1772   \ifcsundef{end#1}{%
1773   \ifcsundef{endLWR@print@#1}{%
1774   \PackageError{lwarp}
1775   {%
1776   \protect\end#1 or \protect\endLWR@print@#1\MessageBreak

```
1777                    must be defined before using \protect\LWR@formatted, etc%
1778                }
1779                {Perhaps #1 is misspelled.}
1780            }{\relax}%
1781        }{\relax}%
1782        \ifcsundef{endLWR@HTML@#1}{%
1783            \PackageError{lwarp}
1784            {%
1785                \protect\endLWR@HTML@#1 must be defined
1786                before using \protect\LWR@formatted, etc%
1787            }
1788            {Perhaps #1 is misspelled.}
1789        }{\relax}%
1790 }
```

\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.

```
1791 \renewcommand*{\LWR@formatted}[1]{%
1792     \LWR@formatted@checkname{#1}%
1793     \ifcsundef{LWR@print@#1}{%
1794         \csNewCommandCopycs{LWR@print@#1}{#1}%
1795     }{}%
1796     \ifcsundef{#1}{%
1797         \expandafter\newrobustcmd\csname #1\endcsname{%
1798             \@nameuse{LWR@\LWR@formatting @#1}%
1799         }%
1800     }{%
1801         \expandafter\renewrobustcmd\csname #1\endcsname{%
1802             \@nameuse{LWR@\LWR@formatting @#1}%
1803         }%
1804     }%
1805 }
```

\LWR@expandableformatted {⟨*macroname*⟩}   No backslash in the macro name.

An expandable version of \LWR@formatted.

```
1806 \renewcommand*{\LWR@expandableformatted}[1]{%
1807     \LWR@formatted@checkname{#1}%
1808     \ifcsundef{LWR@print@#1}{%
1809         \csNewCommandCopycs{LWR@print@#1}{#1}%
1810     }{}%
1811     \ifcsundef{#1}{%
1812         \expandafter\newcommand\csname #1\endcsname{%
1813             \@nameuse{LWR@\LWR@formatting @#1}%
1814         }%
1815     }{%
1816         \expandafter\renewcommand\csname #1\endcsname{%
1817             \@nameuse{LWR@\LWR@formatting @#1}%
1818         }%
1819     }%
1820 }
```

\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.

```
1821 \renewcommand*{\LWR@formattedenv}[1]{%
1822     \LWR@formatted@checkname{#1}%
1823     \LWR@formatted@checkendname{#1}%
1824     \ifcsundef{LWR@print@#1}{%
1825         \NewEnvironmentCopy{LWR@print@#1}{#1}%
1826     }{}%
1827     \DeclareDocumentEnvironment{#1}{}%
1828     {%
1829         \@nameuse{LWR@\LWR@formatting @#1}%
1830     }%
1831     {%
1832         \@nameuse{endLWR@\LWR@formatting @#1}%
1833     }%
1834 }
```

\LWR@expandableformattedenv {⟨*environmentname*⟩}

An expandable version of LWR@formattedenv.

```
1835 \renewcommand*{\LWR@expandableformattedenv}[1]{%
1836     \LWR@formatted@checkname{#1}%
1837     \LWR@formatted@checkendname{#1}%
1838     \ifcsundef{LWR@print@#1}{%
1839         \NewEnvironmentCopy{LWR@print@#1}{#1}%
1840     }{}%
1841     \DeclareExpandableDocumentEnvironment{#1}{}%
1842     {%
1843         \@nameuse{LWR@\LWR@formatting @#1}%
1844     }%
1845     {%
1846         \@nameuse{endLWR@\LWR@formatting @#1}%
1847     }%
1848 }
```

```
1849 \end{warpHTML}
```

37 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: 1850 \begin{warpall}

37.1 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.

```
1851 \newbool{FormatEPUB}
1852 \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.

```
1853 \newbool{FormatWP}
1854 \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]

```
1855 \newbool{WPMarkFloats}
1856 \boolfalse{WPMarkFloats}
```

WPMarkMinipages (*bool*)   Adds
Default: false

```
=== begin minipage ===
. . .
=== end minipage ===
```

around minipages while formatting for word processors. This helps identify bound-
aries of minipages to be manually converted to word-processor frames.

```
1857 \newbool{WPMarkMinipages}
1858 \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.*

```
1859 \newbool{WPMarkTOC}
1860 \booltrue{WPMarkTOC}
```

WPMarkLOFT (*bool*)   While formatting for word processors, adds
Default: false

```
=== list of figures ===    and/or
=== list of tables ===
```

---

[18]Perhaps some day word processors will have HTML import options for identifying <figure> and
caption tags for figures and 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.*

```
1861 \newbool{WPMarkLOFT}
1862 \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.

```
1863 \newbool{WPMarkMath}
1864 \boolfalse{WPMarkMath}
```

WPTitleHeading (*bool*)   While formatting for word processors, true sets the document title to <h1>,
Default: false   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**.

See table 11 on page 185.

```
1865 \newbool{WPTitleHeading}
1866 \boolfalse{WPTitleHeading}
```

```
1867 \end{warpall}
```

37.2   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**.

for HTML output: 1868 \begin{warpHTML}

```
1869 \AtBeginDocument{
1870 \ifbool{FormatWP}{
1871 \@ifundefined{chapter}{
1872 \ifbool{WPTitleHeading}{% part and section starting at h2
1873 \renewcommand*{\LWR@tagtitle}{h1}
1874 \renewcommand*{\LWR@tagtitleend}{/h1}
1875 \renewcommand*{\LWR@tagpart}{h2}
1876 \renewcommand*{\LWR@tagpartend}{/h2}
1877 \renewcommand*{\LWR@tagsection}{h3}
1878 \renewcommand*{\LWR@tagsectionend}{/h3}
1879 \renewcommand*{\LWR@tagsubsection}{h4}
1880 \renewcommand*{\LWR@tagsubsectionend}{/h4}
1881 \renewcommand*{\LWR@tagsubsubsection}{h5}
1882 \renewcommand*{\LWR@tagsubsubsectionend}{/h5}
1883 \renewcommand*{\LWR@tagparagraph}{h6}
1884 \renewcommand*{\LWR@tagparagraphend}{/h6}
1885 \renewcommand*{\LWR@tagsubparagraph}{span class=\textquotedbl{}subparagraph\textquotedbl}
1886 \renewcommand*{\LWR@tagsubparagraphend}{/span}
1887 }% WPTitleHeading
1888 {% not WPTitleHeading, part and section starting at h1
```

```
1889 \renewcommand*{\LWR@tagtitle}{div class=\textquotedbl{}title\textquotedbl}
1890 \renewcommand*{\LWR@tagtitleend}{/div}
1891 \renewcommand*{\LWR@tagpart}{h1}
1892 \renewcommand*{\LWR@tagpartend}{/h1}
1893 \renewcommand*{\LWR@tagsection}{h2}
1894 \renewcommand*{\LWR@tagsectionend}{/h2}
1895 \renewcommand*{\LWR@tagsubsection}{h3}
1896 \renewcommand*{\LWR@tagsubsectionend}{/h3}
1897 \renewcommand*{\LWR@tagsubsubsection}{h4}
1898 \renewcommand*{\LWR@tagsubsubsectionend}{/h4}
1899 \renewcommand*{\LWR@tagparagraph}{h5}
1900 \renewcommand*{\LWR@tagparagraphend}{/h5}
1901 \renewcommand*{\LWR@tagsubparagraph}{h6}
1902 \renewcommand*{\LWR@tagsubparagraphend}{/h6}
1903 }% not WPTitleHeading
1904 }% chapter undefined
1905 {% chapter defined
1906 \ifbool{WPTitleHeading}{}
1907 {% not WPTitleHeading, part and chapter starting at h1
1908 \renewcommand*{\LWR@tagtitle}{div class=\textquotedbl{}title\textquotedbl}
1909 \renewcommand*{\LWR@tagtitleend}{/div}
1910 \renewcommand*{\LWR@tagpart}{h1}
1911 \renewcommand*{\LWR@tagpartend}{/h1}
1912 \renewcommand*{\LWR@tagchapter}{h2}
1913 \renewcommand*{\LWR@tagchapterend}{/h2}
1914 \renewcommand*{\LWR@tagsection}{h3}
1915 \renewcommand*{\LWR@tagsectionend}{/h3}
1916 \renewcommand*{\LWR@tagsubsection}{h4}
1917 \renewcommand*{\LWR@tagsubsectionend}{/h4}
1918 \renewcommand*{\LWR@tagsubsubsection}{h5}
1919 \renewcommand*{\LWR@tagsubsubsectionend}{/h5}
1920 \renewcommand*{\LWR@tagparagraph}{h6}
1921 \renewcommand*{\LWR@tagparagraphend}{/h6}
1922 \renewcommand*{\LWR@tagsubparagraph}{span class=\textquotedbl{}subparagraph\textquotedbl}
1923 \renewcommand*{\LWR@tagsubparagraphend}{/span}
1924 }% not WPTitleHeading
1925 }% chapter defined
1926 }{}% FormatWP
1927 }% AtBeginDocument

1928 \end{warpHTML}
```

38   Remembering original formatting macros

`1929 \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.

`1930 \chardef\LWR@origpound=`\#`

```
1931 \let\LWR@origcomma\,
1932 \let\LWR@origtilde~
1933 \let\LWR@orighfil\hfil
1934 \let\LWR@orighss\hss
1935 \let\LWR@origllap\llap
1936 \let\LWR@origrlap\rlap
1937 \let\LWR@orighfilneg\hfilneg
1938 \let\LWR@orighspace\hspace
1939
1940 \let\LWR@origrule\rule
1941
1942 \let\LWR@origmedskip\medskip
1943 \let\LWR@origbigskip\bigskip
```

libertinus-otf has too much kerning for \textquotedbl, causing an extra space.

```
1944 \LetLtxMacro\LWR@orig@@textquotedbl\textquotedbl
1945 \LetLtxMacro\LWR@orig@textquotedbl\LWR@orig@@textquotedbl
1946
1947 \AtEndPreamble{
1948 \IfPackageLoadedTF{libertinus-otf}{
1949     \renewcommand{\LWR@orig@textquotedbl}{\LWR@orig@@textquotedbl\kern-.15em}
1950      \LetLtxMacro\textquotedbl\LWR@orig@textquotedbl
1951 }{}
1952 }
1953 \LetLtxMacro\LWR@origttfamily\ttfamily
1954
1955 \LetLtxMacro\LWR@origem\em
1956
1957 \LetLtxMacro\LWR@orignormalfont\normalfont
1958
1959 \let\LWR@origonecolumn\onecolumn
1960
1961 \let\LWR@origsp\sp
1962 \let\LWR@origsb\sb
1963
1964 \LetLtxMacro\LWR@origunderline\underline
1965 \let\LWR@orignewpage\newpage
1966
1967 \let\LWR@origpagestyle\pagestyle
1968 \let\LWR@origthispagestyle\thispagestyle
1969 \LetLtxMacro\LWR@origpagenumbering\pagenumbering
1970
1971 \let\LWR@orignewline\newline
1972
1973 \AtBeginDocument{% in case packages change definition
1974 \let\LWR@orig@trivlist\@trivlist
1975 \let\LWR@origtrivlist\trivlist
1976 \let\LWR@origendtrivlist\endtrivlist
1977 \LetLtxMacro\LWR@origitem\item
1978 \LetLtxMacro\LWR@origitemize\itemize
1979 \LetLtxMacro\LWR@endorigitemize\enditemize
1980 \LetLtxMacro\LWR@origenumerate\enumerate
1981 \LetLtxMacro\LWR@endorigenumerate\endenumerate
1982 \LetLtxMacro\LWR@origdescription\description
1983 \LetLtxMacro\LWR@endorigdescription\enddescription
1984 \let\LWR@orig@mklab\@mklab
```

```
1985 \let\LWR@origmakelabel\makelabel
1986 \let\LWR@orig@donoparitem\@donoparitem
1987 \LetLtxMacro\LWR@orig@item\@item
1988 \let\LWR@orig@nbitem\@nbitem
1989 }
1990
1991 \let\LWR@origpar\par
1992
1993 \LetLtxMacro\LWR@origfootnote\footnote
1994 \let\LWR@orig@mpfootnotetext\@mpfootnotetext
1995
1996
1997 \AtBeginDocument{% in case packages change definition
1998 \LetLtxMacro\LWR@orighline\hline%
1999 \LetLtxMacro\LWR@origcline\cline%
2000 }
```

```
2001 \end{warpHTML}
```

39 Accents

Native LaTeX accents such as \" will work, but many more kinds of accents are
available when using Unicode-aware X͇LaTeX 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: 2002 \begin{warpHTML}

Without \AtBeginDocument, \t was being re-defined somewhere.

```
2003 \AtBeginDocument{
```

The following are restored for print when inside a lateximage.

For Unicode engines, only \t needs to be redefined:

```
2004 \LetLtxMacro\LWR@origtie\t
```

For PDF LaTeX, additional work is required:

```
2005 \ifPDFTeX% pdflatex or dvi latex
2006 \LetLtxMacro\LWR@origgraveaccent\`
2007 \LetLtxMacro\LWR@origacuteaccent\'
2008 \LetLtxMacro\LWR@origcircumflexaccent\^
2009 \LetLtxMacro\LWR@origtildeaccent\~
2010 \LetLtxMacro\LWR@origmacronaccent\=
2011 \LetLtxMacro\LWR@origbreve\u
2012 \LetLtxMacro\LWR@origdotaccent\.
2013 \LetLtxMacro\LWR@origdiaeresisaccent\"
2014 \LetLtxMacro\LWR@origdoubleacuteaccent\H
2015 \LetLtxMacro\LWR@origcaronaccent\v
2016 \LetLtxMacro\LWR@origdotbelowaccent\d
2017 \LetLtxMacro\LWR@origcedillaaccent\c
2018 \LetLtxMacro\LWR@origmacronbelowaccent\b
```

The HTML redefinitions follow.

For PDF LATEX, Unicode diacritical marks are used:

```
2019 \renewcommand*{\'}[1]{#1\HTMLunicode{0300}}
2020 \renewcommand*{\'}[1]{#1\HTMLunicode{0301}}
2021 \renewcommand*{\^}[1]{#1\HTMLunicode{0302}}
2022 \renewcommand*{\~}[1]{#1\HTMLunicode{0303}}
2023 \renewcommand*{\=}[1]{#1\HTMLunicode{0304}}
2024 \renewcommand*{\u}[1]{#1\HTMLunicode{0306}}
2025 \renewcommand*{\.}[1]{#1\HTMLunicode{0307}}
2026 \renewcommand*{\"}[1]{#1\HTMLunicode{0308}}
2027 \renewcommand*{\H}[1]{#1\HTMLunicode{030B}}
2028 \renewcommand*{\v}[1]{#1\HTMLunicode{030C}}
2029 \renewcommand*{\d}[1]{#1\HTMLunicode{0323}}
2030 \renewcommand*{\c}[1]{#1\HTMLunicode{0327}}
2031 \renewcommand*{\b}[1]{#1\HTMLunicode{0331}}
2032 \fi
```

For all engines, a Unicode diacritical tie is used:

```
2033 \def\LWR@t#1#2{#1\HTMLunicode{0361}#2}
2034 \renewcommand*{\t}[1]{\LWR@t#1}
```

\LWR@restoreorigaccents Called from \restoreoriginalformatting when a lateximage is begun.

```
2035 \ifPDFTeX% pdflatex or dvi latex
2036 \newcommand*{\LWR@restoreorigaccents}{%
2037     \LetLtxMacro\'\LWR@origgraveaccent%
2038     \LetLtxMacro\'\LWR@origacuteaccent%
2039     \LetLtxMacro\^\LWR@origcircumflexaccent%
2040     \LetLtxMacro\~\LWR@origtildeaccent%
2041     \LetLtxMacro\=\LWR@origmacronaccent%
2042     \LetLtxMacro\u\LWR@origbreve%
2043     \LetLtxMacro\.\LWR@origdotaccent%
2044     \LetLtxMacro\"\LWR@origdiaeresisaccent%
2045     \LetLtxMacro\H\LWR@origdoubleacuteaccent%
2046     \LetLtxMacro\v\LWR@origcaronaccent%
2047     \LetLtxMacro\t\LWR@origtie%
2048     \LetLtxMacro\d\LWR@origdotbelowaccent%
2049     \LetLtxMacro\c\LWR@origcedillaaccent%
2050     \LetLtxMacro\b\LWR@origmacronbelowaccent%
2051 }%
2052 \else% XeLaTeX, LuaLaTeX:
2053 \newcommand*{\LWR@restoreorigaccents}{%
2054     \LetLtxMacro\t\LWR@origtie%
2055 }%
2056 \fi%
2057 }% AtBeginDocument

2058 \end{warpHTML}
```

40 Configuration files

40.1 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:

```
2059 \LWR@excludecomment{LWRwriteconf}{writeconf}
```

Generate configuration files if print mode and not -pstool:

for PRINT output:
```
2060 \begin{warpprint}
2061 \fullexpandarg%
2062 \IfSubStr*{\jobname}{-pstool}
2063     {
2064         \PackageInfo{lwarp}{%
2065             Jobname with -pstool is found.\MessageBreak
2066             Not generating lwarp configuration files,%
2067         }
2068     }
2069     {
2070         \PackageInfo{lwarp}{Generating lwarp configuration files,}%
2071         \LWR@includecomment{LWRwriteconf}{writeconf}
2072     }
2073 \end{warpprint}
```

40.2 <project>_html.tex

*_html.tex (*file*)   Used to allow an HTML version of the document to exist alongside the print version.

Config file:
```
2074 \begin{LWRwriteconf}
2075 \immediate\openout\LWR@quickfile=\jobname_html.tex
2076 \immediate\write\LWR@quickfile{%
2077 \detokenize{\PassOptionsToPackage}%
2078 {warpHTML,BaseJobname=\jobname}{lwarp}%
2079 }
2080 \immediate\write\LWR@quickfile{%
2081 \detokenize{\input}\string{\jobname.tex\string }%
2082 }
2083 \immediate\closeout\LWR@quickfile
2084 \end{LWRwriteconf}
```

40.3 *lwarpmk* configuration files

Config file:
```
2085 \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.)

```
2086 \newcommand*{\LWR@lwarpconfversion}{2}% also in lwarpmk.lua
```

40.3.1 Helper macros

\LWR@shellescapecmd The LaTeX compile option for shell escape, if used.

```
2087 \ifshellescape
2088     \def\LWR@shellescapecmd{--shell-escape }
2089 \else
2090     \def\LWR@shellescapecmd{}
2091 \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

```
2092 \newcommand*{\LWR@compilecmd}[2]{%
2093     #1 \LWR@shellescapecmd \jobname#2%
2094 }
```

\LWR@addcompilecmd {⟨*cmd*⟩} {⟨*suffix*⟩}

Adds to the compilation command.

Cmd is *dvipdfmx*, etc. Suffix is empty or _html

```
2095 \newcommand*{\LWR@addcompilecmd}[2]{%
2096     \LWRopseq
2097     #1 \jobname#2%
2098 }
```

\LWR@unknownengine Error message if not sure which LATEX engine is being used.

```
2099 \newcommand*{\LWR@unknownengine}{%
2100     \PackageError{lwarp}%
2101         {Unknown LaTeX engine}%
2102         {%
2103             Lwarp only knows about pdflatex, DVI latex,
2104             xelatex, lualatex, and upLateX.%
2105         }%
2106 }
```

\LWR@latexmkvar {⟨*varname*⟩} {⟨*value*⟩}

Adds a *latexmk* variable assignment.

```
2107 \newcommand*{\LWR@latexmkvar}[2]{%
2108     -e
```

```
2109        \LWRopquote%
2110        \LWRdollar #1=q/#2/%
2111        \LWRopquote
2112 }
```

\LWR@latexmkcmd {⟨*latexmk options*⟩}

Sets a call to *latexmk* with the given options, possibly adding --shell-escape, and also adding the indexing program.

```
2113 \newcommand*{\LWR@latexmkcmd}[1]{%
2114        latexmk  \space \LWR@shellescapecmd \space  #1 \space
2115        -recorder \space
2116        \LWR@latexmkvar{makeindex}{\LWR@LatexmkIndexCmd}%
2117 }
```

\LWR@latexmkdvipdfm {⟨*dvipdfm or dvipdfmx*⟩}

Adds the options settings for *dvipdfm* or *dvipdfmx*.

```
2118 \newcommand*{\LWR@latexmkdvipdfm}[1]{%
2119        -pdfdvi \space
2120        \LWR@latexmkvar{dvipdf}{%
2121            #1
2122            \@percentchar O
2123            -o \@percentchar D
2124            \@percentchar S%
2125        }
2126 }
```

\LWR@compileuplatex   Sets compile options for upLATEX with ujarticle or related classes.

```
2127 \newcommand*{\LWR@compileuplatex}{
2128        \def\LWR@tempprintlatexcmd{%
2129            \LWR@compilecmd{uplatex}{}
2130            \LWR@addcompilecmd{dvipdfmx}{}
2131        }
2132        \def\LWR@tempHTMLlatexcmd{%
2133            \LWR@compilecmd{uplatex}{_html}
2134            \LWR@addcompilecmd{dvipdfmx}{_html}
2135        }
2136 }
```

\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.

2137 \ifbool{LWR@latexmk}{

For *latexmk* with *pdflatex* or *lualatex*:

2138   \ifpdf

For *latexmk* with *pdflatex*:

```
2139       \ifPDFTeX
2140           \def\LWR@latexcmd{\LWR@latexmkcmd{-pdf -dvi- -ps-}}
2141       \else
```

For *latexmk* with *lualatex*:

```
2142           \ifLuaTeX
2143               \def\LWR@latexcmd{\LWR@latexmkcmd{-lualatex}}
2144           \else
2145               \LWR@unknownengine
2146           \fi
2147       \fi
2148   \else% \ifpdf
```

For *latexmk* with *xelatex* or DVI *latex*:

```
2149       \ifXeTeX
```

For *latexmk* with *xelatex*:

```
2150           \def\LWR@latexcmd{\LWR@latexmkcmd{-xelatex}}
2151       \else% \ifXeTeX
```

For *latexmk* with DVI *latex*:

```
2152           \ifbool{LWR@dvipdfm}{
2153               \def\LWR@latexcmd{%
2154                   \LWR@latexmkcmd{%
2155                       \LWR@latexmkdvipdfm{dvipdfm}%
2156                   }
2157               }
2158           }{
2159               \ifbool{LWR@dvipdfmx}{
2160                   \def\LWR@latexcmd{%
2161                       \LWR@latexmkcmd{%
2162                           \LWR@latexmkdvipdfm{dvipdfmx}%
2163                       }
2164                   }
2165               }{
2166                   \def\LWR@latexcmd{\LWR@latexmkcmd{-pdfps}}
2167               }
```

```
2168                 }
2169         \fi
2170     \fi% \ifpdf
```

The final assignment if *latexmk*:

```
2171     \def\LWR@tempprintlatexcmd{\LWR@latexcmd \space \jobname}
2172     \def\LWR@tempHTMLlatexcmd{\LWR@latexcmd \space \jobname_html}
2173 }% latexmk
```

Without *latexmk*, the compiling command is simply the compiler name and the optional shell escape:

```
2174 {% not latexmk
2175     \ifpdf
```

For *pdflatex* or *lualatex*:

```
2176         \ifPDFTeX
```

For *pdflatex*:

```
2177             \def\LWR@tempprintlatexcmd{\LWR@compilecmd{pdflatex}{}}
2178             \def\LWR@tempHTMLlatexcmd{\LWR@compilecmd{pdflatex}{_html}}
2179         \else
2180             \ifLuaTeX
```

For *lualatex*:

```
2181                 \def\LWR@tempprintlatexcmd{\LWR@compilecmd{lualatex}{}}
2182                 \def\LWR@tempHTMLlatexcmd{\LWR@compilecmd{lualatex}{_html}}
2183             \else
2184                 \LWR@unknownengine
2185             \fi
2186         \fi
2187     \else% \ifpdf
```

For DVI *latex* or *xelatex*:

```
2188         \ifXeTeX
```

For *xelatex*:

```
2189             \def\LWR@tempprintlatexcmd{\LWR@compilecmd{xelatex}{}}
2190             \def\LWR@tempHTMLlatexcmd{\LWR@compilecmd{xelatex}{_html}}
2191         \else
```

For DVI *latex*. Default to *dvips*, unless told to use *dvipdfm* or *dvipdfmx*:

```
2192             \ifbool{LWR@dvipdfm}{
```

For DVI *latex* with *dvipdfm*:

```
2193                 \def\LWR@tempprintlatexcmd{%
2194                     \LWR@compilecmd{latex}{}
2195                     \LWR@addcompilecmd{dvipdfm}{}
2196                 }
2197                 \def\LWR@tempHTMLlatexcmd{%
```

```
2198                    \LWR@compilecmd{latex}{_html}
2199                    \LWR@addcompilecmd{dvipdfm}{_html}
2200                }
2201            }{
2202                \ifbool{LWR@dvipdfmx}{
```

For DVI *latex* with *dvipdfmx*:

```
2203                    \def\LWR@tempprintlatexcmd{%
2204                        \LWR@compilecmd{latex}{}
2205                        \LWR@addcompilecmd{dvipdfmx}{}
2206                    }
2207                    \def\LWR@tempHTMLlatexcmd{%
2208                        \LWR@compilecmd{latex}{_html}
2209                        \LWR@addcompilecmd{dvipdfmx}{_html}
2210                    }
2211                }{% dvips
```

For DVI *latex* with *dvips* and *ps2pdf*:

```
2212                    \def\LWR@tempprintlatexcmd{%
2213                        \LWR@compilecmd{latex}{}
2214                        \LWR@addcompilecmd{dvips}{}
2215                       \LWR@addcompilecmd{ps2pdf -dALLOWPSTRANSPARENCY}{}.ps
2216                    }
2217                    \def\LWR@tempHTMLlatexcmd{%
2218                        \LWR@compilecmd{latex}{_html}
2219                        \LWR@addcompilecmd{dvips}{_html}
2220                      \LWR@addcompilecmd{ps2pdf -dALLOWPSTRANSPARENCY}{_html}.ps
2221                    }
2222                }
2223            }
2224        \fi% \ifXeTeX
2225    \fi% \ifpdf
2226 }% latexmk
```

For ujarticle, utarticle, and related, using upLATEX and *dvipdfmx*:

```
2227 \IfClassLoadedTF{ujarticle}{\LWR@compileuplatex}{}
2228 \IfClassLoadedTF{ujbook}{\LWR@compileuplatex}{}
2229 \IfClassLoadedTF{ujreport}{\LWR@compileuplatex}{}
2230 \IfClassLoadedTF{utarticle}{\LWR@compileuplatex}{}
2231 \IfClassLoadedTF{utbook}{\LWR@compileuplatex}{}
2232 \IfClassLoadedTF{utreport}{\LWR@compileuplatex}{}
```

Only make the setting permanent if the original was empty:

```
2233 \ifdefempty{\LWR@PrintLatexCmd}{
2234    \def\LWR@PrintLatexCmd{\LWR@tempprintlatexcmd}
2235 }{}
2236 \ifdefempty{\LWR@HTMLLatexCmd}{
2237    \def\LWR@HTMLLatexCmd{\LWR@tempHTMLlatexcmd}
2238 }{}
```

\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.

```
2239 \newcommand{\LWR@writeconf}[1]{
2240 \ifcsdef{LWR@quickfile}{}{\newwrite{\LWR@quickfile}}
2241 \immediate\openout\LWR@quickfile=#1
2242 \immediate\write\LWR@quickfile{confversion = [[\LWR@lwarpconfversion]]}
2243 \ifbool{usingOSWindows}{
2244     \immediate\write\LWR@quickfile{opsystem = [[Windows]]}
2245 }{
2246     \immediate\write\LWR@quickfile{opsystem = [[Unix]]}
2247 }
2248 \immediate\write\LWR@quickfile{sourcename = [[\jobname]]}
2249 \immediate\write\LWR@quickfile{homehtmlfilename = [[\HomeHTMLFilename]]}
2250 \immediate\write\LWR@quickfile{htmlfilename = [[\HTMLFilename]]}
2251 \immediate\write\LWR@quickfile{imagesdirectory = [[\LWR@ImagesDirectory]]}
2252 \immediate\write\LWR@quickfile{imagesname = [[\LWR@ImagesName]]}
2253 \immediate\write\LWR@quickfile{latexmk = [[\ifbool{LWR@latexmk}{true}{false}]]}
2254 \immediate\write\LWR@quickfile{printlatexcmd = [[\LWR@PrintLatexCmd]]}
2255 \immediate\write\LWR@quickfile{HTMLlatexcmd = [[\LWR@HTMLLatexCmd]]}
2256 \immediate\write\LWR@quickfile{printindexcmd = [[\LWR@PrintIndexCmd]]}
2257 \immediate\write\LWR@quickfile{HTMLindexcmd = [[\LWR@HTMLIndexCmd]]}
2258 \immediate\write\LWR@quickfile{latexmkindexcmd = [[\LWR@LatexmkIndexCmd]]}
2259 \immediate\write\LWR@quickfile{glossarycmd = [[\LWR@GlossaryCmd]]}
2260 \immediate\write\LWR@quickfile{pdftotextenc = [[\LWR@pdftotextEnc]]}
2261 \immediate\closeout\LWR@quickfile
2262 }
2263
2264 \end{LWRwriteconf}
```

40.3.2 `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:
```
2265 \begin{LWRwriteconf}
2266
2267 \AtBeginDocument{\LWR@writeconf{lwarpmk.conf}}
2268
2269 \end{LWRwriteconf}
```

40.3.3 `<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:
```
2270 \begin{LWRwriteconf}
2271
2272 \AtBeginDocument{\LWR@writeconf{\jobname.lwarpmkconf}}
2273
2274 \end{LWRwriteconf}
```

40.4 `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:
```
2275 \begin{LWRwriteconf}
2276 \begin{filecontents*}[overwrite]{lwarp.css}
2277 /*
2278   CSS stylesheet for the LaTeX Lwarp package
2279   Copyright 2016-2022 Brian Dunn — BD Tech Concepts LLC
2280 */
2281
2282
2283 /* a fix for older browsers: */
2284 header, section, footer, aside, nav, main,
2285     article, figure { display: block; }
2286
2287
2288 A:link {color:#000080 ; text-decoration: none ; }
2289 A:visited {color:#800000 ; }
2290 A:hover {color:#000080 ; text-decoration: underline ;}
2291 A:active {color:#800000 ; }
2292
2293 a.tocbook {display: inline-block ; margin-left: 0em ;
2294     font-weight: bold ; margin-top: 1ex ; margin-bottom: 1ex ; }
2295 a.tocpart {display: inline-block ; margin-left: 0em ;
2296     font-weight: bold ;}
2297 a.tocchapter {display: inline-block ; margin-left: 0em ;
2298     font-weight: bold ;}
2299 a.tocsection {display: inline-block ; margin-left: 1em ;
2300     text-indent: -.5em ; font-weight: bold ; }
2301 a.tocsubsection {display: inline-block ; margin-left: 2em ;
2302     text-indent: -.5em ; }
2303 a.tocsubsubsection {display: inline-block ; margin-left: 3em ;
2304     text-indent: -.5em ; }
2305 a.tocparagraph {display: inline-block ; margin-left: 4em ;
2306     text-indent: -.5em ; }
2307 a.tocsubparagraph {display: inline-block ; margin-left: 5em ;
2308     text-indent: -.5em ; }
2309 a.tocfigure {margin-left: 0em}
2310 a.tocsubfigure {margin-left: 2em}
2311 a.toctable {margin-left: 0em}
2312 a.tocsubtable {margin-left: 2em}
2313 a.toctheorem {margin-left: 0em}
2314 a.toclstlisting {margin-left: 0em}
2315
2316 body {
2317     font-family: "DejaVu Serif", "Bitstream Vera Serif",
2318         "Lucida Bright", Georgia, serif;
2319     background: #FAF7F4 ;
2320     color: black ;
2321     margin:0em ;
2322     padding:0em ;
2323     font-size: 100% ;
2324     line-height: 1.2 ;
2325 }
2326
2327 p {margin: 1.5ex 0em 1.5ex 0em ;}
```

```
2328 table p {margin: .5ex 0em .5ex 0em ;}
2329
2330 /* Holds a section number */
2331 span.sectionnumber { margin-right: 0em }
2332
2333 /* Inserted in front of index lines */
2334 span.indexitem {margin-left: 0em}
2335 span.indexsubitem {margin-left: 2em}
2336 span.indexsubsubitem {margin-left: 4em}
2337 div.indexheading {margin-top: 2ex ; font-weight: bold}
2338
2339 div.hidden, span.hidden { display: none ; }
2340
2341 kbd, span.texttt, p span.texttt {
2342     font-family: "DejaVu Mono", "Bitstream Vera Mono", "Lucida Console",
2343         "Nimbus Mono L", "Liberation Mono", "FreeMono", "Andale Mono",
2344         "Courier New", monospace;
2345     font-size: 100% ;
2346 }
2347
2348 pre { padding: 3pt ; }
2349
2350 span.strong, span.textbf, div.strong, div.textbf, table td.tdbfseries { font-weight: bold; }
2351
2352 span.textit, div.textit, table td.tditshape { font-style: italic; }
2353
2354 table td.tdbfit { font-weight: bold ; font-style:italic }
2355
2356 span.textnormal, div.textnormal {
2357     font-weight: normal;
2358     font-style: normal;
2359     font-variant: normal;
2360     font-variant-numeric: normal ;
2361     font-family: "DejaVu Serif", "Bitstream Vera Serif",
2362         "Lucida Bright", Georgia, serif;
2363 }
2364
2365 span.textmd, div.textmd { font-weight: normal; }
2366
2367 span.textup, div.textup {
2368     font-style: normal;
2369     font-variant: normal;
2370     font-variant-numeric: normal ;
2371 }
2372
2373 span.textsc, div.textsc {
2374     font-variant: small-caps;
2375     font-variant-numeric: oldstyle-nums ;
2376 }
2377
2378 span.textulc, div.textulc {
2379     font-variant: normal ;
2380     font-variant-numeric: normal ;
2381 }
2382
2383 span.textsl, div.textsl { font-style: oblique; }
2384
2385 span.textrm, div.textrm {
2386     font-family: "DejaVu Serif", "Bitstream Vera Serif",
2387     "Lucida Bright", Georgia, serif;
```

```
2388 }
2389
2390 span.textsf, div.textsf {
2391     font-family: "DejaVu Sans", "Bitstream Vera Sans",
2392         Geneva, Verdana, sans-serif ;
2393 }
2394
2395 /* nfssext-cfr lining figures */
2396 span.textln, div.textln {
2397     font-variant-numeric: lining-nums ;
2398 }
2399
2400 /* nfssext-cfr proportional figures */
2401 span.textp, div.textp {
2402     font-variant-numeric: proportional-nums ;
2403 }
2404
2405 /* nfssext-cfr tabular figures */
2406 span.textt, div.textt {
2407     font-variant-numeric: tabular-nums ;
2408 }
2409
2410 /* nfssext-cfr font weights */
2411 span.textdb, div.textdb {
2412     font-weight: 500 ;
2413 }
2414
2415 span.textsb, div.textsb {
2416     font-weight: 600 ;
2417 }
2418
2419 span.texteb, div.texteb {
2420     font-weight: 800 ;
2421 }
2422
2423 span.textub, div.textub {
2424     font-weight: 900 ;
2425 }
2426
2427 span.textlg, div.textlg {
2428     font-weight: 300 ;
2429 }
2430
2431 span.textel, div.textel {
2432     font-weight: 200 ;
2433 }
2434
2435 span.textul, div.textul {
2436     font-weight: 100 ;
2437 }
2438
2439
2440
2441 span.textcircled { border: 1px solid black ; border-radius: 1ex ; }
2442
2443 span.underline {
2444     text-decoration: underline ;
2445     text-decoration-skip: auto ;
2446 }
2447
```

```
2448 span.overline {
2449     text-decoration: overline ;
2450     text-decoration-skip: auto ;
2451 }
2452
2453 div.hrule { border-top: 1px solid silver }
2454
2455
2456 /* for vertical text: */
2457 div.verticalrl { writing-mode: vertical-rl }
2458 div.horizontaltb { writing-mode: horizontal-tb }
2459
2460
2461 /* for diagbox */
2462 div.diagboxtitleN { border-bottom: 1px solid gray }
2463 div.diagboxtitleS { border-top: 1px solid gray }
2464
2465 div.diagboxE {
2466     padding-left: 2em ;
2467     text-align: right ;
2468 }
2469
2470 div.diagboxW {
2471     padding-right: 2em ;
2472     text-align: left ;
2473 }
2474
2475
2476
2477 /* For realscripts */
2478 .supsubscript {
2479     display: inline-block;
2480     text-align:left ;
2481 }
2482
2483 .supsubscript sup,
2484 .supsubscript sub {
2485     position: relative;
2486     display: block;
2487     font-size: .7em;
2488     line-height: 1;
2489 }
2490
2491 .supsubscript sup {
2492     top: .3em;
2493 }
2494
2495 .supsubscript sub {
2496     top: .3em;
2497 }
2498
2499 div.attribution p {
2500     text-align: right ;
2501     font-size: 80%
2502 }
2503
2504 span.poemtitle {
2505   font-size: 120% ; font-weight: bold;
2506 }
2507
```

```
2508 pre.tabbing {
2509     font-family: "Linux Libertine Mono O", "Lucida Console",
2510         "Droid Sans Mono", "DejaVu Mono", "Bitstream Vera Mono",
2511         "Liberation Mono", "FreeMono", "Andale Mono",
2512         "Nimbus Mono L", "Courier New", monospace;
2513 }
2514
2515 blockquote {
2516     display: block ;
2517     margin-left: 2em ;
2518     margin-right: 2em ;
2519 }
2520
2521 /* quotchap is for the quotchap package */
2522 div.quotchap {
2523     display: block ;
2524     font-style: oblique ;
2525     overflow-x: auto ;
2526     margin-left: 2em ;
2527     margin-right: 2em ;
2528 }
2529
2530 blockquote p, div.quotchap p {
2531     line-height: 1.5;
2532     text-align: left ;
2533     font-size: .85em ;
2534 }
2535
2536 /* qauthor is for the quotchap package */
2537 div.qauthor {
2538   display: block ;
2539   text-align: right ;
2540   margin-left: auto ;
2541   margin-right: 2em ;
2542   font-size: 80% ;
2543   font-variant: small-caps;
2544 }
2545
2546 div.qauthor p {
2547   text-align: right ;
2548 }
2549
2550 div.epigraph, div.dictum {
2551   line-height: 1.2;
2552     text-align: left ;
2553     padding: 3ex 1em 0ex 1em ;
2554 /*     margin: 3ex auto 3ex auto ; */ /* Epigraph centered */
2555     margin: 3ex 1em 3ex auto ; /* Epigraph to the right */
2556 /*    margin: 3ex 1em 3ex 1em ; */ /* Epigraph to the left */
2557     font-size: .85em ;
2558     max-width: 27em ;
2559 }
2560
2561 div.epigraphsource, div.dictumauthor {
2562     text-align:right ;
2563     margin-left:auto ;
2564 /*     max-width: 50% ; */
2565     border-top: 1px solid #A0A0A0 ;
2566     padding-bottom: 3ex ;
2567     line-height: 1.2;
```

```
2568 }
2569
2570 div.epigraph p, div.dictum p { padding: .5ex ; margin: 0ex ;}
2571 div.epigraphsource p, div.dictumauthor p { padding: .5ex 0ex 0ex 0ex ; margin: 0ex ;}
2572 div.dictumauthor { font-style:italic }
2573
2574
2575 /* copyrightbox package: */
2576 div.copyrightbox { margin: .5ex .5em }
2577 div.copyrightbox p {margin: 0px .5em ; padding: 0px}
2578 div.copyrightboxnote {text-align: left ; font-size: 60%}
2579
2580
2581 /* lettrine package: */
2582 span.lettrine { font-size: 4ex ; float: left ; }
2583 span.lettrinetext { font-variant: small-caps ; }
2584
2585 /* ulem, soul, umoline packages: */
2586 span.uline {
2587     text-decoration: underline ;
2588     text-decoration-skip: auto ;
2589 }
2590
2591 span.uuline {
2592     text-decoration: underline ;
2593     text-decoration-skip: auto ;
2594     text-decoration-style: double ;
2595 }
2596
2597 span.uwave {
2598     text-decoration: underline ;
2599     text-decoration-skip: auto ;
2600     text-decoration-style: wavy ;
2601 }
2602
2603 span.sout {
2604     text-decoration: line-through ;
2605 }
2606
2607 span.oline {
2608     text-decoration: overline ;
2609     text-decoration-skip: auto ;
2610 }
2611
2612 span.xout {
2613     text-decoration: line-through ;
2614 }
2615
2616 span.dashuline {
2617     text-decoration: underline ;
2618     text-decoration-skip: auto ;
2619     text-decoration-style: dashed ;
2620 }
2621
2622 span.dotuline {
2623     text-decoration: underline ;
2624     text-decoration-skip: auto ;
2625     text-decoration-style: dotted ;
2626 }
2627
```

```
2628 span.letterspacing { letter-spacing: .2ex ; }
2629
2630 span.capsspacing {
2631     font-variant: small-caps ;
2632     letter-spacing: .1ex ;
2633 }
2634
2635 span.highlight { background: #F8E800 ; }
2636
2637
2638 /* keystroke package: */
2639 span.keystroke {
2640     border-style: outset ;
2641     padding: 0pt .5em 0pt .5em ;
2642 }
2643
2644
2645 html body {
2646   margin: 0 ;
2647   line-height: 1.2;
2648 }
2649
2650
2651 body div {
2652   margin: 0ex;
2653 }
2654
2655
2656 div.book, h1, h2, h3, h4, h5, h6, span.paragraph, span.subparagraph
2657 {
2658     font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
2659         "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
2660         "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
2661         "Times New Roman", serif;
2662     font-style: normal ;
2663     font-weight: bold ;
2664     text-align: left ;
2665 }
2666
2667 h1 {    /* title of the entire website, used on each page */
2668     text-align: center ;
2669     font-size: 2.5em ;
2670     padding: .4ex 0em 0ex 0em ;
2671 }
2672
2673 div.book {
2674     text-align: center ;
2675     font-size: 2.325em ;
2676     padding: .4ex 0em 0ex 0em ;
2677 }
2678
2679 h2 { font-size: 2.25em }
2680 h3 { font-size: 2em }
2681 h4 { font-size: 1.75em }
2682 h5 { font-size: 1.5em }
2683 h6 { font-size: 1.25em }
2684 span.paragraph {font-size: 1em ; font-variant: normal ;
2685     margin-right: 1em ; }
2686 span.subparagraph {font-size: 1em ; font-variant: normal ;
2687     margin-right: 1em ;   }
```

```
2688
2689 div.minisec {
2690     font-family: "DejaVu Sans", "Bitstream Vera Sans",
2691         Geneva, Verdana, sans-serif ;
2692     font-style: normal ;
2693     font-weight: bold ;
2694     text-align: left ;
2695 }
2696
2697 h1 {
2698   margin: 0ex 0em 0ex 0em ;
2699   line-height: 1.3;
2700   text-align: center ;
2701 }
2702
2703 h2 {
2704   margin: 1ex 0em 1ex 0em ;
2705   line-height: 1.3;
2706   text-align: center ;
2707 }
2708
2709 h3 {
2710   margin: 3ex 0em 1ex 0em ;
2711   line-height: 1.3;
2712 }
2713
2714 h4 {
2715   margin: 3ex 0em 1ex 0em ;
2716   line-height: 1.3;
2717 }
2718
2719 h5 {
2720   margin: 3ex 0em 1ex 0em ;
2721   line-height: 1.3;
2722 }
2723
2724 h6 {
2725   margin: 3ex 0em 1ex 0em ;
2726   line-height: 1.3;
2727 }
2728
2729
2730 div.titlepage {
2731   text-align: center ;
2732 }
2733
2734 .footnotes {
2735     text-align: left ;
2736     font-size: .85em ;
2737     margin: 3ex 2em 0ex 2em ;
2738     border-top: 1px solid silver ;
2739 }
2740
2741 .marginpar, .marginparblock {
2742     max-width: 50%;
2743     float: right ;
2744     clear: both ;
2745     text-align: left ;
2746     margin: 1ex 0.5em 1ex 1em ;
2747     padding: 1ex 0.5em 1ex 0.5em ;
```

```
2748     font-size: 85% ;
2749     border-top: 1px solid silver ;
2750     border-bottom: 1px solid silver ;
2751     overflow-x: auto ;
2752 }
2753
2754 .marginpar br { margin-bottom: 2ex ; }
2755
2756 div.marginblock, div.marginparblock {
2757     max-width:50%;
2758     min-width: 10em; /* room for caption */
2759     float:right;
2760     text-align:left;
2761     margin: 1ex 0.5em 1ex 1em ;
2762     padding: 1ex 0.5em 1ex 0.5em ;
2763     overflow-x: auto;
2764 }
2765
2766 div.marginblock div.minipage,
2767 div.marginparblock div.minipage {
2768     display: inline-block ;
2769     margin: 0pt auto 0pt auto ;
2770 }
2771
2772 div.marginblock div.minipage p ,
2773 div.marginparblock div.minipage p
2774     { font-size: 85%}
2775
2776 div.marginblock br ,
2777 div.marginparblock br
2778     { margin-bottom: 2ex ; }
2779
2780 main.bodycontainer {
2781     float: left ;
2782     width: 80% ;
2783 }
2784
2785 div.bodywithoutsidetoc main.bodycontainer {
2786     float: none ;
2787     width: 100% ;
2788 }
2789
2790 section.textbody div.footnotes{
2791     margin: 1ex 2em 2ex 2em ;
2792     border-bottom: 2px solid silver ;
2793 }
2794
2795 .footnoteheader {
2796     border-top: 2px solid silver ;
2797     margin-top: 3ex ;
2798     padding-top: 1ex ;
2799     font-weight: bold ;
2800 }
2801
2802 .mpfootnotes {
2803     text-align: left ;
2804     font-size: .85em ;
2805     margin-left: 1em ;
2806     border-top: 1px solid silver ;
2807 }
```

```
2808
2809 /* Remove footnote top border in the title page. */
2810 div.titlepage div.mpfootnotes {
2811     border-top: none ;
2812 }
2813
2814
2815
2816 ul, ol {
2817   margin: 1ex 1em 1ex 0em;
2818   line-height: 1.2;
2819 }
2820
2821 body dir, body menu {
2822   margin: 3ex 1em 3ex 0em;
2823   line-height: 1.2;
2824 }
2825
2826 li { margin: 0ex 0em 1ex 0em; }
2827
2828 li.p { display: inline ; }
2829
2830 html {
2831   margin: 0;
2832   padding: 0;
2833 }
2834
2835 .programlisting {
2836   font-family: "DejaVu Mono", "Bitstream Vera Mono", "Lucida Console",
2837         "Nimbus Mono L", "Liberation Mono", "FreeMono", "Andale Mono",
2838         "Courier New", monospace;
2839   margin: 1ex 0ex 1ex 0ex ;
2840   padding: .5ex 0pt .5ex 0pt ;
2841   overflow-x: auto;
2842 }
2843
2844 section.textbody>pre.programlisting {
2845 border-top: 1px solid silver ;
2846 border-bottom: 1px solid silver ;
2847 }
2848
2849
2850 div.displaymath {
2851     text-align: center ;
2852 }
2853
2854 div.displaymathnumbered {
2855     text-align: right ;
2856     margin-left: 5% ;
2857     margin-right: 5% ;
2858     min-width: 2.5in ;
2859 }
2860
2861 @media all and (min-width: 400px) {
2862     div.displaymathnumbered {
2863         margin-left: 10% ;
2864         margin-right: 10% ;
2865     }
2866 }
2867
```

```
2868 @media all and (min-width: 800px) {
2869     div.displaymathnumbered {
2870         margin-right: 20% ;
2871     }
2872 }
2873
2874 @media all and (min-width: 1200px) {
2875     div.displaymathnumbered {
2876         margin-right: 30% ;
2877     }
2878 }
2879
2880
2881 .inlineprogramlisting {
2882   font-family: "DejaVu Mono", "Bitstream Vera Mono", "Lucida Console",
2883         "Nimbus Mono L", "Liberation Mono", "FreeMono", "Andale Mono",
2884         "Courier New", monospace;
2885   overflow-x: auto;
2886 }
2887
2888 span.listinglabel {
2889     display: inline-block ;
2890     font-size: 70% ;
2891     width: 4em ;
2892     text-align: right ;
2893     margin-right: 2em ;
2894 }
2895
2896 div.abstract {
2897   margin: 2em 5% 2em 5% ;
2898   padding: 1ex 1em 1ex 1em ;
2899 /* font-weight: bold ; */
2900   font-size: 90% ;
2901     text-align: left ;
2902 }
2903
2904 div.abstract dl {line-height:1.5;}
2905 div.abstract dt {color:#304070;}
2906
2907 div.abstracttitle{
2908     font-family: "URW Classico", Optima, "Linux Biolinum O",
2909         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
2910         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
2911     font-weight:bold;
2912     font-size:1.25em;
2913     text-align: center ;
2914 }
2915
2916 span.abstractrunintitle{
2917     font-family: "URW Classico", Optima, "Linux Biolinum O",
2918         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
2919         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
2920     font-weight:bold;
2921 }
2922
2923
2924 .verbatim {
2925     overflow-x: auto ;
2926 }
2927
```

```
2928 .alltt {
2929     overflow-x: auto ;
2930 }
2931
2932
2933 .bverbatim {
2934     margin: 1ex 0pt 1ex 0pt ;
2935     padding: .5ex 0pt .5ex 0pt ;
2936     overflow-x: auto ;
2937 }
2938
2939 .lverbatim {
2940     margin: 1ex 0pt 1ex 0pt ;
2941     padding: .5ex 0pt .5ex 0pt ;
2942     overflow-x: auto ;
2943 }
2944
2945 .fancyvrb {
2946     font-size:.85em ;
2947     margin: 3ex 0pt 3ex 0pt
2948 }
2949
2950 .fancyvrblabel {
2951     font-size: .85em ;
2952     text-align: center ;
2953     font-weight: bold ;
2954     margin-top: 1ex ;
2955     margin-bottom: 1ex ;
2956 }
2957
2958
2959 .verse {
2960     font-family: "Linux Libertine Mono O", "Lucida Console",
2961         "Droid Sans Mono", "DejaVu Mono", "Bitstream Vera Mono",
2962         "Liberation Mono", "FreeMono", "Andale Mono",
2963         "Nimbus Mono L", "Courier New", monospace;
2964     margin-left: 1em ;
2965 }
2966
2967
2968 div.singlespace { line-height: 1.2 ; }
2969 div.onehalfspace { line-height: 1.5 ; }
2970 div.doublespace { line-height: 2 ; }
2971
2972
2973 /* Word processor format output: */
2974 div.wpfigure { border: 1px solid red ; margin: .5ex ; padding: .5ex ; }
2975 div.wptable { border: 1px solid blue ; margin: .5ex ; padding: .5ex ; }
2976 div.wpminipage { border: 1px solid green ; margin: .5ex ; padding: .5ex ;}
2977
2978
2979
2980
2981 /* Minipage environments, vertically aligned to top, center, bottom: */
2982 .minipage, .fminipage, .fcolorminipage {
2983     /* display: inline-block ; */
2984         /* Mini pages which follow each other will be tiled. */
2985     text-align:left;
2986     margin: .25em .25em .25em .25em;
2987     padding: .25em .25em .25em .25em;
```

```
2988      display: inline-flex;
2989      flex-direction: column ;
2990      overflow: auto;
2991 }
2992
2993 .inlineminipage {
2994      display: inline-block ;
2995      text-align: left
2996 }
2997
2998 /* Paragraphs in the flexbox did not collapse their margins. */
2999 /* Have not yet researched this. */
3000 .minipage p {margin: .75ex 0em .75ex 0em ;}
3001
3002 .fboxBlock .minipage, .colorbox .minipage, .colorboxBlock .minipage,
3003 .fcolorbox .minipage, .fcolorboxBlock .minipage
3004      {border: none ; background: none;}
3005
3006 .fbox, .fboxBlock { border: 1px solid black ; padding: 4pt }
3007
3008 .fbox, .fboxBlock, .fcolorbox, .fcolorboxBlock, .colorbox, .colorboxBlock,
3009 .fminipage, .fcolorminipage
3010      {display: inline-block}
3011
3012 .shadowbox, .shabox {
3013   border: 1px solid black;
3014   box-shadow: 3px 3px 3px #808080 ;
3015     border-radius: 0px ;
3016     padding: .4ex .3em .4ex .3em ;
3017     margin: 0pt .3ex 0pt .3ex ;
3018   display: inline-block ;
3019 }
3020
3021 .doublebox {
3022   border: 3px double black;
3023     border-radius: 0px ;
3024     padding: .4ex .3em .4ex .3em ;
3025     margin: 0pt .3ex 0pt .3ex ;
3026   display: inline-block ;
3027 }
3028
3029 .ovalbox, .Ovalbox {
3030   border: 1px solid black;
3031     border-radius: 1ex ;
3032     padding: .4ex .3em .4ex .3em ;
3033     margin: 0pt .3ex 0pt .3ex ;
3034   display: inline-block ;
3035 }
3036
3037 .Ovalbox { border-width: 2px ; }
3038
3039 .framebox {
3040   border: 1px solid black;
3041     border-radius: 0px ;
3042     padding: .3ex .2em 0ex .2em ;
3043     margin: 0pt .1ex 0pt .1ex ;
3044   display: inline-block ;
3045 }
3046
3047
```

```
3048 /* mdframed, tcolorbox, shadebox packages */
3049 .mdframed, .tcolorbox, .shadebox {
3050     padding: 0ex ;
3051     margin: 2ex 0em 2ex 0em ;
3052     border: 1px solid black ;
3053 }
3054
3055 .tcolorbox {
3056     border-radius: 10pt ;
3057     margin: 2ex 1em 2ex 1em ;
3058 }
3059
3060 .mdframed p, .tcolorbox p { padding: 0ex .5em 0ex .5em ; }
3061
3062 .mdframed dl, .tcolorbox dl { padding: 1ex .5em 0ex .5em ; }
3063
3064 .mdframedtitle, .tcolorboxtitle {
3065     padding: .5ex 0pt 0pt 0pt ;
3066     border-radius: 10pt 10pt 0pt 0pt ;
3067     display: block ;
3068     margin-bottom: 1ex ;
3069     border-bottom: 1px solid silver ;
3070 }
3071
3072 .tcolorboxsubtitle .tcolorbox {
3073     margin: 2ex 0em 2ex 0em ;
3074     border-radius: 0pt ;
3075 }
3076
3077 .mdframedsubtitle {
3078     display: block ;
3079 }
3080
3081 .mdframedsubsubtitle {
3082     display: block ;
3083 }
3084
3085 .mdtheorem {
3086     padding: 0ex .5em 0ex .5em ;
3087     margin: 3ex 5% 3ex 5% ;
3088 }
3089
3090
3091 /* framed package */
3092 .framed, pre.boxedverbatim, fcolorbox {
3093     margin: 3ex 0em 3ex 0em ;
3094   border: 1px solid black;
3095     border-radius: 0px ;
3096     padding: .3ex 1em 0ex 1em ;
3097   display: block ;
3098 }
3099
3100 .shaded {
3101     margin: 3ex 0em 3ex 0em ;
3102     padding: .3ex 1em .3ex 1em ;
3103     display: block ;
3104 }
3105
3106 .snugframed {
3107     margin: 3ex 0em 3ex 0em ;
```

```
3108      border: 1px solid black;
3109        border-radius: 0px ;
3110   display: block ;
3111 }
3112
3113 .framedleftbar {
3114      margin: 3ex 0em 3ex 0em ;
3115    border-left: 3pt solid black;
3116        border-radius: 0px ;
3117      padding: .3ex .2em .3ex 1em ;
3118   display: block ;
3119 }
3120
3121 .framedtitle {
3122      margin: 0em ;
3123      padding: 0em ;
3124      font-size: 130%
3125 }
3126
3127 .framedtitle p { padding: .3em }
3128
3129
3130 /* For the niceframe package: */
3131
3132 div.niceframe, div.curlyframe, div.artdecoframe, div.generalframe {
3133      padding: 1ex ;
3134      margin: 2ex auto ;
3135      border-radius: 2ex ;
3136 }
3137
3138 div.niceframe {
3139      border: 6px groove black ;
3140 }
3141
3142 div.curlyframe {
3143      border-left: 3px dotted black ;
3144      border-right: 3px dotted black ;
3145      border-radius: 6ex ;
3146 }
3147
3148 div.artdecoframe {
3149      border-left: 10px double black ;
3150      border-right: 10px double black ;
3151      border-radius: 6ex ;
3152 }
3153
3154 div.generalframe {
3155      border: 6px groove black ;
3156 }
3157
3158
3159 /* For beamerarticle: */
3160 div.beamerframe {
3161      margin: 3ex 1em 3ex 1em ;
3162    border: 1px solid gray;
3163        border-radius: 0px ;
3164      padding: .3ex 1em 0ex 1em ;
3165   display: block ;
3166 }
3167
```

```
3168
3169 dl {
3170   margin: 1ex 2em 1ex 0em;
3171   line-height: 1.3;
3172 }
3173
3174 dl dt {
3175     display: block ;
3176     float:left ;
3177     font-weight: bold;
3178     padding-right: 1em ;
3179 }
3180
3181 dl dd { display: block ; }
3182
3183 dl dd:after { content: "" ; display: block ; clear: both }
3184
3185 dl dd p { margin-top: 0em; }
3186
3187 dd ul, dd ol, dd dl {
3188     clear: both ;
3189 /*    padding-top: 1ex ; */
3190 }
3191
3192
3193 nav {
3194     font-family: "URW Classico", Optima, "Linux Biolinum O",
3195         "DejaVu Sans", "Bitstream Vera Sans",
3196         Geneva, Verdana, sans-serif ;
3197     margin-bottom: 4ex ;
3198 }
3199
3200 nav p {
3201     line-height: 1.2 ;
3202     margin-top:.5ex ;
3203     margin-bottom:.5ex;
3204     font-size: .9em ;
3205 }
3206
3207
3208
3209 img, img.hyperimage, img.borderimage {
3210     max-width: 600px;
3211     border: 1px solid silver;
3212     box-shadow: 3px 3px 3px #808080 ;
3213     padding: .5% ;
3214     margin: .5% ;
3215     background: none ;
3216 }
3217
3218 img.inlineimage{
3219     padding: 0px ;
3220     box-shadow: none ;
3221     border: none ;
3222     background: none ;
3223     margin: 0px ;
3224     display: inline-block ;
3225     border-radius: 0px ;
3226 }
3227
```

```
3228 img.logoimage{
3229     max-width: 300px ;
3230     box-shadow: 3px 3px 3px #808080 ;
3231     border: 1px solid black ;
3232     background:none ;
3233     padding:0 ;
3234     margin:.5ex ;
3235     border-radius: 10px ;
3236 }
3237
3238
3239 .section {
3240 /*
3241     To have each section float relative to each other:
3242 */
3243 /*
3244     display: block ;
3245     float: left ;
3246     position: relative ;
3247     background: white ;
3248     border: 1px solid silver ;
3249     padding: .5em ;
3250 */
3251     margin: 0ex .5em 0ex .5em ;
3252     padding: 0 ;
3253 }
3254
3255
3256 figure {
3257     margin: 5ex auto 5ex auto ;
3258     padding: 1ex 1em 1ex 1em ;
3259     overflow-x: auto ;
3260 }
3261
3262
3263 /* To automatically center images in figures: */
3264 /*
3265 figure img.inlineimage {
3266     margin: 0ex auto 0ex auto ;
3267     display: block ;
3268 }
3269 */
3270
3271 /* To automatically center minipages in figures: */
3272 /*
3273 figure div.minipage, figure div.minipage div.minipage {
3274     margin: 1ex auto 1ex auto ;
3275     display: block ;
3276 }
3277 */
3278
3279 figure figure { margin: 0pt }
3280
3281 figure div.minipage p { font-size: 85% ; }
3282
3283 figure.subfigure, figure.subtable  {
3284     display: inline-block ; margin: 3ex 1em 3ex 1em ;
3285 }
3286
3287 div.figurecaption .minipage { margin:0 ; padding: 0 }
```

```
3288
3289 /* for subcaptions: */
3290 figure div.minipage div.figurecaption {
3291     max-width: 100% ;    /* fallback if min() does not work */
3292     max-width: min(30em,100%)
3293 }
3294
3295 div.minipage figure { border: none ; box-shadow: none ; }
3296 div.minipage figure.table { margin: 0ex }
3297 div.minipage div.footnotes { margin: 1ex 2em 0ex 2em }
3298
3299 div.floatrow { text-align: center; }
3300
3301 div.floatrow figure { display: inline-block ; margin: 1ex 2% ; }
3302
3303 div.floatfoot { font-size: .85em ;
3304     border-top: 1px solid silver ; line-height: 1.2 ; }
3305
3306 /* Center if only one line, "start" align if more than one line: */
3307 div.figurecaption , .lstlistingtitle {
3308     font-size: .85em ;
3309     font-weight: bold ;
3310     text-align: start ;
3311     margin: 1ex auto;
3312     width: max-content;
3313     max-width: 100%;
3314 }
3315
3316 /* A marginblock is small, so always center and don't mess with the width. */
3317 div.marginblock div.figurecaption {
3318     width: 100% ;
3319     text-align: center ;
3320 }
3321
3322 figure.subfigure div.figurecaption, figure.subtable div.figurecaption  {
3323     border-bottom: none ; background: none ;
3324 }
3325
3326 div.nonfloatcaption {
3327     margin: 1ex auto 1ex auto ;
3328     font-size: .85em ;
3329     text-align: center ;
3330     font-weight: bold ;
3331 }
3332
3333 /* For a \RawCaption inside a minipage inside a figure's floatrow: */
3334 figure div.floatrow div.minipage div.figurecaption {
3335     border: none ;
3336     background: none ;
3337 }
3338
3339
3340 /* For packages such as float, rotfloat, and algorithm2e: */
3341
3342 figure.boxed, figure.boxruled {
3343     border: 1px solid black ;
3344 }
3345
3346 figure.ruled {
3347     border-top: 1px solid black ;
```

```
3348      border-bottom: 1px solid black ;
3349      border-left: 0px ;
3350      border-right: 0px ;
3351      border-radius: 0px ;
3352      background: none ;
3353      box-shadow: none ;
3354 }
3355
3356 figure.ruled div.figurecaption, figure.boxruled div.figurecaption {
3357      border-top: 1px solid silver ;
3358      border-bottom: 1px solid silver ;
3359 }
3360
3361
3362 table {
3363      margin: 1ex auto 1ex auto ;
3364      border-collapse: separate ;
3365      border-spacing: 0px ;
3366      line-height: 1.3 ;
3367      }
3368
3369 table > tbody > tr.hline > td {border-top: 1px solid #808080 ; margin-top: 0ex ;
3370      margin-bottom: 0ex ; } /* for \hline */
3371
3372 tr.tbrule td {border-top: 1px solid black ; margin-top: 0ex ;
3373      margin-bottom: 0ex ; } /* for \toprule, \bottomrule */
3374
3375 td {padding: .5ex .5em .5ex .5em ;}
3376
3377 table td.tdl { text-align: left ; vertical-align: middle ; }
3378 table td.tdc { text-align: center ; vertical-align: middle ; }
3379 table td.tdat { text-align: center ; vertical-align: middle ; padding: 0px ; margin: 0px ; }
3380 table td.tdbang { text-align: center ; vertical-align: middle ; }
3381 table td.tdr { text-align: right ; vertical-align: middle ; }
3382 table td.tdp { text-align: left ; vertical-align: bottom ; }
3383 table td.tdm { text-align: left ; vertical-align: middle ; }
3384 table td.tdb { text-align: left ; vertical-align: top ; }
3385
3386 table td.tvertbarl { border-left: 1px solid black }
3387 table td.tvertbarldouble { border-left: 4px double black }
3388 table td.tvertbarr { border-right: 1px solid black }
3389 table td.tvertbarrdouble { border-right: 4px double black }
3390
3391 table td.tvertbarldash { border-left: 1px dashed black }
3392 table td.tvertbarldoubledash { border-left: 2px dashed black }
3393 table td.tvertbarrdash { border-right: 1px dashed black }
3394 table td.tvertbarrdoubledash { border-right: 2px dashed black }
3395
3396 table td.tdcenter { text-align: center}
3397 table td.tdleft { text-align: left}
3398 table td.tdright { text-align: right}
3399
3400
3401 /* for cmidrules: */
3402 table td.tdrule {
3403      border-top: 1px solid #A0A0A0 ;
3404 }
3405
3406 table td.tdrulel {
3407      border-top-left-radius:.5em ;
```

```
3408      border-top: 1px solid #A0A0A0 ;
3409 }
3410
3411 table td.tdruler {
3412      border-top-right-radius:.5em ;
3413      border-top: 1px solid #A0A0A0 ;
3414 }
3415
3416 table td.tdrulelr {
3417      border-top-left-radius:.5em ;
3418      border-top-right-radius:.5em ;
3419      border-top: 1px solid #A0A0A0 ;
3420 }
3421
3422
3423 /* Margins of paragraphs inside table cells: */
3424 td.tdp p , td.tdprule p , td.tdP p , td.tdPrule p { padding-top: 1ex ;
3425      padding-bottom: 1ex ; margin: 0ex ; }
3426 td.tdm p , td.tmbrule p , td.tdM p , td.tdMrule p { padding-top: 1ex ;
3427      padding-bottom: 1ex ; margin: 0ex ; }
3428 td.tdb p , td.tdbrule p , td.tdB p , td.tdBrule p { padding-top: 1ex ;
3429      padding-bottom: 1ex ; margin: 0ex ; }
3430
3431 td.tdp , td.tdprule , td.tdP , td.tdPrule
3432      { padding: 0ex .5em 0ex .5em ; }
3433 td.tdm , td.tdmrule , td.tdM , td.tdMrule
3434      { padding: 0ex .5em 0ex .5em ; }
3435 td.tdb , td.tdbrule , td.tdB , td.tdBrule
3436      { padding: 0ex .5em 0ex .5em ; }
3437
3438
3439 /* table notes: */
3440 .tnotes {
3441      margin: 0ex 5% 1ex 5% ;
3442      padding: 0.5ex 1em 0.5ex 1em;
3443      font-size:.80em;
3444      text-align: left ;
3445 }
3446
3447 .minipage .tnotes {
3448      margin: 0pt ;
3449      padding: 0pt ;
3450 }
3451
3452 .tnotes dl dt p {margin-bottom:0px;}
3453
3454 .tnoteitemheader {margin-right: 1em;}
3455
3456
3457 /* for colortbl and cell color */
3458 div.cellcolor {
3459      width: 100% ;
3460      padding: .5ex .5em .5ex .5em ;
3461      margin: -.5ex -.5em -.5ex -.5em ;
3462 }
3463
3464
3465 /* for lyluatex */
3466 span.lyluatex {
3467      display: inline-block ;
```

```
3468 }
3469
3470 div.lyluatex p span.lateximagesource img {
3471     display: block ;
3472     margin-top: 3ex ;
3473     margin-bottom: 3ex ;
3474 }
3475
3476
3477 /* for bigdelim */
3478 .ldelim, .rdelim { font-size: 200% }
3479
3480
3481 /* center, flushleft, flushright environments */
3482 div.center{text-align:center;}
3483 div.center table {margin-left:auto;margin-right:auto;}
3484 div.flushleft{text-align:left;}
3485 div.flushleft table {margin-left:0em ; margin-right:auto;}
3486 div.flushright{text-align:right;}
3487 div.flushright table {margin-left:auto ; margin-right: 0em ;}
3488
3489
3490 /* Fancybox */
3491 div.Btrivlist table tr td {
3492     padding: .2ex 0em ;
3493 }
3494
3495
3496 /* program listing callouts: */
3497 span.callout {
3498     font-family: "DejaVu Sans", "Bitstream Vera Sans",
3499         Geneva, Verdana, sans-serif ;
3500     border-radius: .5em;
3501     background-color:black;
3502     color:white;
3503     padding:0px .25em 0px .25em;
3504     margin: 0 ;
3505     font-weight: bold;
3506     font-size:.72em ;
3507 }
3508
3509 div.programlisting pre.verbatim span.callout{
3510     font-size: .85em ;
3511 }
3512
3513 span.verbatim {
3514     font-family: "DejaVu Mono", "Bitstream Vera Mono", "Lucida Console",
3515         "Nimbus Mono L", "Liberation Mono", "FreeMono", "Andale Mono",
3516         "Courier New", monospace;
3517 }
3518
3519
3520
3521 div.titlehead
3522 {
3523     text-align: left ;
3524     font-style: normal ;
3525     font-weight: normal ;
3526     font-style: normal ;
3527     font-size: .8em ;
```

```
3528        margin: 1ex 0em 1ex 0em ;
3529 }
3530
3531 div.subject
3532 {
3533        text-align: center ;
3534        font-style: normal ;
3535        font-weight: bold ;
3536        font-style: normal ;
3537        font-size: .8em ;
3538        margin: 1ex 0em 1ex 0em ;
3539 }
3540
3541 div.published
3542 {
3543        text-align: center ;
3544        font-variant: normal ;
3545        font-style: italic ;
3546        font-size: 1em ;
3547        margin: 1ex 0em 1ex 0em ;
3548 }
3549
3550 div.subtitle
3551 {
3552        text-align: center ;
3553        font-variant: normal ;
3554        font-style: italic ;
3555        font-size: 1.25em ;
3556        margin: 1ex 0em 1ex 0em ;
3557 }
3558
3559 div.subtitle p { margin: 1ex ; }
3560
3561 div.author
3562 {
3563        text-align: center ;
3564        font-variant: normal ;
3565        font-style: normal ;
3566        font-size: 1em ;
3567        margin: 1ex 0em 1ex 0em ;
3568 }
3569
3570 div.oneauthor {
3571        display: inline-block ;
3572        margin: 0ex 1em 0ex 1em ;
3573 }
3574
3575 /*
3576 div.author table {
3577        margin: 1ex auto 0ex auto ;
3578        background: none ;
3579 }
3580
3581 div.author table tbody tr td { padding: .25ex ; }
3582 */
3583
3584 span.affiliation {font-size: .85em ; font-variant: small-caps; }
3585
3586 div.titledate {
3587        text-align: center ;
```

```
3588      font-size: .85em ;
3589      font-style: italic;
3590      margin: 1ex 0em 1ex 0em ;
3591 }
3592
3593
3594 nav.topnavigation{
3595      text-align: left ;
3596      padding: 0.5ex 1em 0.5ex 1em ;
3597 /*      margin: 2ex 0em 3ex 0em ; */
3598      margin: 0 ;
3599      border-bottom: 1px solid silver ;
3600      border-top: 1px solid silver ;
3601      clear:both ;
3602 }
3603
3604 nav.botnavigation{
3605      text-align: left ;
3606      padding: 0.5ex 1em 0.5ex 1em ;
3607 /*      margin: 3ex 0em 2ex 0em ; */
3608      margin: 0 ;
3609      border-top: 1px solid silver ;
3610      border-bottom: 1px solid silver ;
3611      clear:both ;
3612 }
3613
3614
3615 header {
3616      line-height: 1.2 ;
3617      font-size: 1em ;
3618      border-bottom: 1px solid silver ;
3619      margin: 0px ;
3620      padding: 2ex 1em 2ex 1em ;
3621      text-align:left ;
3622 }
3623
3624
3625 footer {
3626      font-size: .85em ;
3627      line-height: 1.2 ;
3628      margin-top: 1ex ;
3629      border-top: 1px solid silver ;
3630      padding: 2ex 1em 2ex 1em ;
3631      clear:both ;
3632      text-align:left ;
3633 }
3634
3635
3636 /* for \LinkHome, \LinkPrevious, and \LinkNext: */
3637 a.linkhome { font-weight:bold ; font-size: 1em ;}
3638
3639
3640 div.lateximagesource { padding: 0px ; margin: 0px ; display: none; }
3641
3642 img.lateximage{
3643      padding: 0pt ;
3644      margin: 0pt ;
3645      box-shadow: none ;
3646      border: none ;
3647      background: none ;
```

```
3648      max-width: 100% ;
3649      border-radius: 0ex ;
3650      border: none ;
3651 }
3652
3653
3654 div.sidetoccontainer {
3655      font-family: "DejaVu Serif", "Bitstream Vera Serif",
3656          "Lucida Bright", Georgia, serif;
3657      float: left ;
3658      width: 19%; /* room for border-right next to 80% main */
3659      margin: 0pt 0em 3ex 0pt ;
3660      border-right: 1px solid silver;
3661      border-bottom: 1px solid silver;
3662      background: #FAF7F4 ;
3663      font-size:.9em ;
3664      border-radius: 0px 0px 20px 0px ;
3665 }
3666
3667 div.sidetoccontents {
3668      overflow-y: auto ;
3669      width: 100% ;
3670      text-align: left ;
3671 }
3672
3673
3674 nav.sidetoc p {line-height:1.2 ; margin: 1ex .5em 1ex .5em ;
3675      text-indent: 0 ; }
3676
3677 nav.sidetoc p a {color:black ; font-size: .7em ;}
3678
3679 div.sidetoctitle {font-size: 1.2em; font-weight:bold; text-align:center;
3680      border-bottom: 1px solid silver ;      }
3681
3682 nav.sidetoc a:hover {text-decoration: underline ; }
3683
3684
3685
3686 section.textbody { margin: 0ex 1em 0ex 1em ;}
3687
3688
3689 div.multicolsheading { -webkit-column-span: all;
3690      -moz-column-span: all; column-span: all; }
3691 div.multicols {
3692      -webkit-columns: 3 auto ;
3693      -moz-columns: 3 auto ;
3694      columns: 3 auto ;
3695 }
3696 div.multicols p {margin-top: 0ex}
3697
3698
3699 /* Used for xfrac and nicefrac: */
3700 span.numerator {
3701      font-size: 60% ;
3702      vertical-align: .4em ;
3703 }
3704
3705 span.denominator {
3706      font-size: 60%
3707 }
```

```
3708
3709
3710 /* Used for algorithm2e: */
3711 div.alg2evline{
3712     margin-left: 1em ;
3713     padding-left: 1em ;
3714     border-left: 1px solid black ;
3715     border-radius: 0px 0px 0px 1ex ;
3716 }
3717
3718 div.alg2evsline{
3719     margin-left: 1em ;
3720     padding-left: 1em ;
3721     border-left: 1px solid black ;
3722 }
3723
3724 div.alg2enoline{
3725     margin-left: 1em ;
3726     padding-left: 1em ;
3727 }
3728
3729 span.alg2elinenumber{
3730     margin-right: .5em ;
3731     font-size: 60% ;
3732     color: red ;
3733 }
3734
3735
3736 /* Used for algorithmicx: */
3737 span.floatright { float: right ; }
3738
3739
3740 /* keyfloat and tocdata: */
3741 .floatnotes {
3742     margin: 0ex 5% 0ex 5% ;
3743     padding: 0ex 1em 0ex 1em ;
3744     font-size:.80em ;
3745     text-align: left ;
3746 }
3747
3748 .authorartist{
3749     display:block ;
3750     font-size:.70em ;
3751     font-style: italic;
3752 }
3753
3754 nav .authorartist{ display:inline; }
3755
3756
3757
3758 /* Native LaTeX theorems: */
3759
3760 .theoremcontents {
3761     font-style: italic; margin-top: 3ex ; margin-bottom: 3ex ;
3762 }
3763
3764 .theoremlabel {
3765     font-style: normal; font-weight: bold ; margin-right: .5em ;
3766 }
3767
```

```
3768
3769
3770 /* theorem, amsthm, and ntheorem packages */
3771
3772 span.theoremheader,
3773 span.theoremheaderplain,
3774 span.theoremheaderdefinition,
3775 span.theoremheaderbreak,
3776 span.theoremheadermarginbreak,
3777 span.theoremheaderchangebreak,
3778 span.theoremheaderchange,
3779 span.theoremheadermargin
3780 {
3781     font-style:normal ; font-weight: bold ; margin-right: 1em ;
3782 }
3783
3784 span.amsthmnameplain,
3785 span.amsthmnamedefinition,
3786 span.amsthmnumberplain,
3787 span.amsthmnumberdefinition
3788 {
3789     font-style:normal ; font-weight: bold ;
3790 }
3791
3792
3793 span.amsthmnameremark,
3794 span.amsthmnumberremark
3795 {font-style:italic ; font-weight: normal ; }
3796
3797
3798 span.amsthmnoteplain,
3799 span.amsthmnotedefinition
3800 {font-style:normal ;}
3801
3802
3803 span.theoremheaderremark,
3804 span.theoremheaderproof,
3805 span.amsthmproofname
3806 {font-style:italic ; font-weight: normal ; margin-right: 1em ; }
3807
3808 span.theoremheadersc
3809 {
3810     font-style:normal ;
3811     font-variant: small-caps ;
3812     font-weight: normal ;
3813     margin-right: 1em ;
3814 }
3815
3816 .theoremendmark {float:right}
3817
3818 div.amsthmbodyplain, div.theorembodyplain, div.theorembodynonumberplain,
3819 div.theorembodybreak, div.theorembodynonumberbreak,
3820 div.theorembodymarginbreak,
3821 div.theorembodychangebreak,
3822 div.theorembodychange,
3823 div.theorembodymargin
3824 {
3825     font-style:italic;
3826     margin-top: 3ex ; margin-bottom: 3ex ;
3827 }
```

```
3828
3829 div.theorembodydefinition, div.theorembodyremark, div.theorembodyproof,
3830 div.theorembodyplainupright, nonumberplainuprightsc,
3831 div.amsthmbodydefinition, div.amsthmbodyremark,
3832 div.amsthmproof
3833 {
3834     font-style: normal ;
3835     margin-top: 3ex ; margin-bottom: 3ex ;
3836 }
3837
3838 span.amsthmnoteremark {}
3839
3840
3841 /* thmbox */
3842
3843 .thmbox {
3844     font-style: italic; margin-top: 3ex ; margin-bottom: 3ex ;
3845     border: 1px solid gray ;
3846     padding: 1ex ;
3847 }
3848
3849 .thmboxtitle {
3850     font-style: normal; font-weight: bold ; margin-right: .5em ;
3851     border-bottom: 1px solid gray ;
3852 }
3853
3854 span.thmboxproofname, span.thmboxexamplename {
3855     font-weight: bold ;
3856 }
3857
3858 div.thmboxproof, div.thmboxexample {
3859     font-size: 0.85em ;
3860     margin: 2ex ;
3861 }
3862
3863 div.thmboxleftbar {
3864     border-left: 2px solid black ;
3865     padding-left: 1em ;
3866 }
3867
3868
3869
3870 /* For the backnaur package: */
3871 div.backnaur {
3872     display: block ;
3873     margin: 2ex 2em 2ex 2em ;
3874 }
3875
3876 div.backnaur p {
3877     margin: .25ex 0ex .25ex 0ex ;
3878 }
3879
3880 div.backnaurprod {
3881     display: inline-block ;
3882     min-width: 8em ;
3883     text-align:right ;
3884 }
3885
3886 div.backnaurdesc {
3887     display: inline-block ;
```

```
3888 }
3889
3890
3891 /* For the notes package: */
3892 div.notesimportantnote, div.noteswarningnote, div.notesinformationnote {
3893     clear: both ;
3894     margin: 2ex 2em 2ex 2em ;
3895     border: 1px solid silver ;
3896 }
3897
3898 div.notesicon {
3899     float:left ;
3900     display: inline-block ;
3901     background: gold ;
3902     padding: 0ex 1em 0ex 1em ;
3903     margin-right: 1em ;
3904     font-weight: bold ;
3905 }
3906
3907 div.notescontents { font-style: italic }
3908
3909
3910 /* nolbreaks package: */
3911 span.nolbreaks { white-space: nowrap ; }
3912
3913
3914 /*
3915 For CSS LaTeX and related logos:
3916 Based on spacing demonstrated by the metafont package.
3917
3918 The subscripts are shrunk instead of lowered below the baseline,
3919 to avoid browser rendering errors with the line height in lists, etc.
3920 */
3921
3922 .latexlogofont {
3923     font-family: "Linux Libertine O", "Nimbus Roman No 9 L",
3924         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
3925     font-variant: normal ;
3926 }
3927
3928 .latexlogo {
3929     font-family: "Linux Libertine O", "Nimbus Roman No 9 L",
3930         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
3931 }
3932
3933 .latexlogosup {
3934   text-transform: uppercase;
3935   letter-spacing: .03em ;
3936   font-size: 0.7em;
3937   vertical-align: 0.25em;
3938   margin-left: -0.4em;
3939   margin-right: -0.15em;
3940 }
3941
3942 .latexlogosub {
3943   text-transform: uppercase;
3944 /*  vertical-align: -0.27ex; */
3945   margin-left: -0.08em;
3946   margin-right: -0.07em;
3947 /*  font-size: 1em; */
```

```
3948      font-size: .7em ;
3949 }
3950
3951 .latexlogotwoe {
3952    text-transform: none ;
3953    font-variant-numeric: oldstyle-nums ;
3954 }
3955
3956 .latexlogotwoesub {
3957    font-style:italic ;
3958 /*  vertical-align: -0.27ex; */
3959    margin-left: -0.11em;
3960    margin-right: -0.1em;
3961 /*  font-size: 1em; */
3962      font-size: .7em ;
3963 }
3964
3965 .xelatexlogo {
3966      font-family: "Linux Libertine O", "Nimbus Roman No 9 L",
3967          "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
3968      letter-spacing: .03em ;
3969 }
3970
3971 .xelatexlogosub {
3972 /*  vertical-align: -0.27ex; */
3973    margin-left: -0.0667em;
3974    margin-right: -.05em;
3975 /*  font-size: 1em; */
3976      font-size: .7em ;
3977    letter-spacing: .03em ;
3978 }
3979
3980 .amslogo {
3981      font-family: "TeXGyreChorus","URW Chancery L",
3982          "Apple Chancery","ITC Zapf Chancery","Monotype Corsiva",
3983          "Linux Libertine O", "Nimbus Roman No 9 L",  "FreeSerif",
3984          "Hoefler Text", Times, "Times New Roman", serif ;
3985      font-style: italic ;
3986 }
3987
3988 .lyxlogo {
3989      font-family: "URW Classico", Optima, "Linux Biolinum O",
3990          "DejaVu Sans", "Bitstream Vera Sans", Geneva,
3991          Verdana, sans-serif ;
3992 }
3993
3994
3995 /* Only display top and bottom navigation if a small screen: */
3996 /* Hide the sidetoc if a small screen: */
3997 nav.topnavigation { display:none; }
3998 nav.botnavigation { display:none; }
3999
4000 /* Only display the sidetoc's webpage title if a small screen */
4001 span.sidetocthetitle { display: none }
4002
4003 @media screen and (max-width: 100em) {
4004      div.multicols {
4005          -webkit-columns: 2 auto ;
4006          -moz-columns: 2 auto ;
4007          columns: 2 auto ;
```

```
4008        }
4009 }
4010
4011 @media screen and (max-width: 50em) {
4012     div.sidetoccontainer {
4013         float: none ;
4014         width: 100% ;
4015         padding: 0 ;
4016         border-radius: 0 ;
4017         border-bottom: 1px solid black ;
4018         border-top: 1px solid black ;
4019         box-shadow: none ;
4020     }
4021     span.sidetocthetitle { display: inline }
4022     nav.topnavigation { display:block }
4023     nav.botnavigation { display:block }
4024     main.bodycontainer { width: 100% }
4025     .marginpar {
4026         max-width: 100%;
4027         float: none;
4028         display:block ;
4029         margin: 1ex 1em 1ex 1em ;
4030     }
4031     div.multicols {
4032         -webkit-columns: 1 auto ;
4033         -moz-columns: 1 auto ;
4034         columns: 1 auto ;
4035     }
4036 }
4037
4038 @media print {
4039     body {
4040         font-family: "Linux Libertine O",
4041         "DejaVu Serif", "Bitstream Vera Serif",
4042         "Liberation Serif", "Nimbus Roman No 9 L",
4043         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
4044     }
4045     div.sidetoccontainer { display:none; }
4046     nav.topnavigation { display: none; }
4047     nav.botnavigation { display: none; }
4048     main.bodycontainer { width: 100% }
4049 }
4050
4051 @media handheld {
4052     div.sidetoccontainer { display:none; }
4053     nav.topnavigation { display:block }
4054     nav.botnavigation { display:block }
4055     main.bodycontainer { width: 100% }
4056 }
4057
4058 @media projection {
4059     div.sidetoccontainer { display:none; }
4060     nav.topnavigation { display:block }
4061     nav.botnavigation { display:block }
4062     main.bodycontainer { width: 100% }
4063 }
4064 \end{filecontents*}
4065 % \end{Verbatim}% for syntax highlighting
4066 \end{LWRwriteconf}
```

40.5 `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:
```
4067 \begin{LWRwriteconf}
4068 \begin{filecontents*}[overwrite]{lwarp_sagebrush.css}
4069 @import url("lwarp.css") ;
4070
4071
4072 A:link {color:#105030 ; text-decoration: none ; }
4073 A:visited {color:#705030 ; text-shadow:1px 1px 2px #a0a0a0;}
4074 A:hover {color:#006000 ; text-decoration: underline ; text-shadow:0px 0px 2px #a0a0a0;}
4075 A:active {color:#00C000 ; text-shadow:1px 1px 2px #a0a0a0;}
4076
4077
4078
4079 div.book, h1, h2, h3, h4, h5, h6, span.paragraph, span.subparagraph
4080 {
4081     font-family: "URW Classico", Optima, "Linux Biolinum O",
4082         "Linux Libertine O", "Liberation Serif",
4083         "Nimbus Roman No 9 L",  "FreeSerif",
4084         "Hoefler Text", Times, "Times New Roman", serif;
4085     font-variant: small-caps ;
4086     font-weight: normal ;
4087     color: #304070 ;
4088     text-shadow: 2px 2px 3px #808080;
4089 }
4090
4091 h1 {    /* title of the entire website, used on each page */
4092     font-variant: small-caps ;
4093     color: #304070 ;
4094     text-shadow: 2px 2px 3px #808080;
4095     background-color: #F7F7F0 ;
4096     background-image: linear-gradient(to bottom, #F7F7F0, #C0C0C4);
4097 }
4098
4099 h1 {
4100   border-bottom: 1px solid #304070;
4101 /*  border-top: 2px solid #304070; */
4102 }
4103
4104 h2 {
4105   border-bottom: 1px solid #304070;
4106 /*  border-top: 2px solid #304070; */
4107     background-color: #F7F7F0 ;
4108     background-image: linear-gradient(to bottom, #F7F7F0, #DAD0C0);
4109 }
4110
4111
4112
4113 div.abstract {
4114     background: #f5f5eb ;
4115     background-image: linear-gradient(to bottom, #f5f5eb, #C8C8B8);
4116
4117   border: 1px solid silver;
4118     border-radius: 1em ;
4119 }
```

```
4120
4121 div.abstract dl {line-height:1.5;}
4122 div.abstract dt {color:#304070;}
4123
4124 div.abstracttitle{
4125     font-family: "URW Classico", Optima, "Linux Biolinum O",
4126         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
4127         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
4128     font-weight:bold;
4129     font-variant: small-caps ;
4130     font-size:1.5em;
4131     border-bottom: 1px solid silver ;
4132     color: #304070 ;
4133     text-align: center ;
4134     text-shadow: 1px 1px 2px #808080;
4135 }
4136
4137 span.abstractrunintitle{
4138     font-family: "URW Classico", Optima, "Linux Biolinum O",
4139         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
4140         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
4141     font-weight:bold;
4142 }
4143
4144
4145 div.epigraph, div.dictum {
4146     background: #f5f5eb ;
4147     background-image: linear-gradient(to bottom, #f5f5eb, #C8C8B8);
4148
4149     border: 1px solid silver ;
4150     border-radius: 1ex ;
4151     box-shadow: 3px 3px 3px #808080 ;
4152 }
4153
4154
4155 .example {
4156     background-color: #f5f5eb ;
4157     background-image: linear-gradient(to bottom, #f5f5eb, #C8C8B8);
4158
4159 }
4160
4161 div.exampletitle{
4162     font-family: "URW Classico", Optima, "Linux Biolinum O",
4163         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
4164         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
4165     font-weight:bold;
4166     font-variant: small-caps ;
4167     border-bottom: 1px solid silver ;
4168     color: #304070 ;
4169     text-align: center ;
4170     text-shadow: 1px 1px 2px #808080;
4171 }
4172
4173
4174 .sidebar {
4175     background-color: #f5f5eb ;
4176     background-image: linear-gradient(to bottom, #f5f5eb, #C8C8B8);
4177
4178 }
4179
```

```
4180 div.sidebartitle{
4181     font-family: "URW Classico", Optima, "Linux Biolinum O",
4182         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
4183         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
4184     font-weight:bold;
4185     font-variant: small-caps ;
4186     border-bottom: 1px solid silver ;
4187     color: #304070 ;
4188     text-align: center ;
4189     text-shadow: 1px 1px 2px #808080;
4190 }
4191
4192
4193 .fancyvrblabel {
4194     font-family: "URW Classico", Optima, "Linux Biolinum O",
4195         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
4196         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
4197     font-weight:bold;
4198     font-variant: small-caps ;
4199     font-size: 1.5em ;
4200     color: #304070 ;
4201     text-align: center ;
4202     text-shadow: 1px 1px 2px #808080;
4203 }
4204
4205 div.minipage {
4206     background-color: #eeeee7 ;
4207     border: 1px solid silver ;
4208     border-radius: 1ex ;
4209 }
4210
4211 table div.minipage { background: none ; border: none ; }
4212
4213 div.framebox div.minipage {border:none ; background:none}
4214
4215 section.textbody > div.minipage {
4216     box-shadow: 3px 3px 3px #808080 ;
4217 }
4218
4219 div.fboxBlock div.minipage { box-shadow: none ; }
4220
4221 .framed .minipage , .framedleftbar .minipage {
4222     border: none ;
4223     background: none ;
4224     padding: 0ex ;
4225     margin: 0ex ;
4226 }
4227
4228 figure.figure .minipage, div.figurecaption .minipage { border: none; }
4229
4230 div.marginblock div.minipage ,
4231 div.marginparblock div.minipage
4232     { border: none; }
4233
4234 figure , div.marginblock {
4235     background-color: #eeeee7 ;
4236     border: 1px solid silver ;
4237     border-radius: 1ex ;
4238     box-shadow: 3px 3px 3px #808080 ;
4239 }
```

```
4240
4241 figure figure {
4242     border: 1px solid silver ;
4243     margin: 0em ;
4244     box-shadow: none ;
4245 }
4246
4247 /*
4248 div.figurecaption {
4249     border-top: 1px solid silver ;
4250     border-bottom: 1px solid silver ;
4251     background-color: #e8e8e8 ;
4252 }
4253 */
4254
4255
4256 div.table {
4257     box-shadow: 3px 3px 3px #808080 ;
4258 }
4259
4260 /*
4261 .tnotes {
4262     background: #e8e8e8;
4263     border: 1px solid silver;
4264 }
4265 */
4266
4267
4268 nav.topnavigation{
4269     background-color: #b0b8b0 ;
4270     background-image: linear-gradient(to bottom,#e0e0e0,#b0b8b0) ;
4271 }
4272
4273 nav.botnavigation{
4274     background-color: #b0b8b0 ;
4275     background-image: linear-gradient(to top,#e0e0e0,#b0b8b0) ;
4276 }
4277
4278
4279
4280 header{
4281     background-color: #F7F7F0 ;
4282     background-image: linear-gradient(to top, #F7F7F0, #b0b8b0);
4283 }
4284
4285 footer{
4286     background-color: #F7F7F0 ;
4287     background-image: linear-gradient(to bottom, #F7F7F0, #b0b8b0);
4288 }
4289
4290
4291
4292 div.sidetoccontainer {
4293     background-color: #F7F7F0 ;
4294     background-image: linear-gradient(to bottom, #F7F7F0, #C0C0C0);
4295     box-shadow: 3px 3px 3px #808080 ;
4296     }
4297
4298 div.sidetoctitle {color: #304070 ; }
4299
```

```
4300 nav.sidetoc a:hover {
4301     color:#006000 ;
4302     text-decoration: none ;
4303     text-shadow:0px 0px 2px #a0a0a0;
4304 }
4305
4306
4307 @media screen and (max-width: 45em) {
4308     div.sidetoccontainer { border-radius: 0 ; }
4309 }
4310
4311
4312 \end{filecontents*}
4313 % \end{Verbatim}% for syntax highlighting
4314 \end{LWRwriteconf}
```

40.6 `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:
```
4315 \begin{LWRwriteconf}
4316 \begin{filecontents*}[overwrite]{lwarp_formal.css}
4317 @import url("lwarp.css") ;
4318
4319
4320
4321 A:link {color:#802020 ; text-decoration:none; }
4322 A:visited {color:#802020 ; text-shadow:none ;}
4323 A:hover {color:#400000 ; text-shadow:none ;}
4324 A:active {color:#C00000 ; text-shadow:none ;}
4325
4326
4327 body {
4328     font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
4329         "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
4330         "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
4331         "Times New Roman", serif;
4332     background: #fffcf5;
4333 }
4334
4335 span.textrm {
4336     font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
4337         "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
4338         "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
4339         "Times New Roman", serif;
4340 }
4341
4342 span.textsf {
4343      font-family: "DejaVu Sans", "Bitstream Vera Sans",
4344         Geneva, Verdana, sans-serif ;
4345 }
4346
4347
4348
4349 div.book, h1, h2, h3, h4, h5, h6, span.paragraph, span.subparagraph
```

```
4350 {
4351     font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
4352         "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
4353         "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
4354         "Times New Roman", serif;
4355     color: #800000 ;
4356     text-shadow: none ;
4357 }
4358
4359 h1, h2 {
4360     background-color: #fffcf5 ;
4361     background-image: none ;
4362     border-bottom: 1px solid #808080;
4363 /*   border-top: 2px solid #808080; */
4364 }
4365
4366 div.abstracttitle {
4367     font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
4368         "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
4369         "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
4370         "Times New Roman", serif;
4371     color: black ;
4372     text-shadow: none ;
4373 }
4374
4375 span.abstractrunintitle {
4376     font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
4377         "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
4378         "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
4379         "Times New Roman", serif;
4380     color: black ;
4381     text-shadow: none ;
4382 }
4383
4384 div.abstract { font-size: 100% }
4385
4386 .sidebar {
4387     background: #fffcf5;
4388     background-image: none ;
4389   margin: 2em 5% 2em 5%;
4390   padding: 0.5em 1em;
4391   border: none ;
4392   border-top : 1px solid silver;
4393   border-bottom : 1px solid silver;
4394   font-size: 90% ;
4395 }
4396
4397 div.sidebartitle{
4398     font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
4399         "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
4400         "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
4401         "Times New Roman", serif;
4402     color: #800000 ;
4403     text-shadow: none ;
4404     border: none ;
4405 }
4406
4407 .example {
4408     background: #fffcf5;
4409     background-image: none ;
```

```
4410    margin: 2em 5% 2em 5%;
4411    padding: 0.5em 1em;
4412    border: none ;
4413    border-top : 1px solid silver;
4414    border-bottom : 1px solid silver;
4415 }
4416
4417 div.exampletitle{
4418    font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
4419        "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
4420        "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
4421        "Times New Roman", serif;
4422    color: #800000 ;
4423    text-shadow: none ;
4424    border: none ;
4425 }
4426
4427 div.fancyvrblabel{
4428    font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
4429        "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
4430        "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
4431        "Times New Roman", serif;
4432    color: #800000 ;
4433    text-shadow: none ;
4434    border: none ;
4435 }
4436
4437
4438
4439 figure {
4440    margin: 5ex 5% 5ex 5% ;
4441    padding: 1ex 1em 1ex 1em ;
4442    background-color: #fffcf5 ;
4443    overflow-x: auto ;
4444    border: none ;
4445 /*    border-top: 1px solid silver; */
4446 /*    border-bottom: 1px solid silver; */
4447 }
4448
4449
4450 div.figurecaption , .lstlisting {
4451    border: none ;
4452 /*    border-top: 1px solid silver ; */
4453 /*    border-bottom: 1px solid silver ; */
4454    background-color: #fffcf5 ;
4455 }
4456
4457 .tnotes {
4458    background: #fffcf5 ;
4459    border-top: 1px solid silver ;
4460    border-bottom: 1px solid silver ;
4461 }
4462
4463 .theorem {
4464        background: none ;
4465 }
4466
4467 .minipage {
4468    background-color: #fffcf5 ;
4469    border: none ;
```

```
4470 }
4471
4472 div.floatrow figure { border: none ; }
4473
4474 figure figure { border: none ; }
4475
4476
4477 nav.toc, nav.lof, nav.lot, nav.lol {
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 }
4483
4484 div.sidetoccontainer {
4485     font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
4486         "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
4487         "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
4488         "Times New Roman", serif;
4489     background-image: linear-gradient(to bottom, #fffcf5, #C0C0C0);
4490 }
4491
4492 div.sidetoctitle{
4493     color: #800000 ;
4494 }
4495
4496 header{
4497     background-color: #e0e0e0 ;
4498     background-image: linear-gradient(to top, #fffcf5, #b0b0b0);
4499     text-align:center ;
4500 }
4501
4502 footer{
4503     background-color: #e0e0e0 ;
4504     background-image: linear-gradient(to bottom, #fffcf5, #b0b0b0);
4505     padding: 2ex 1em 2ex 1em ;
4506     text-align:left ;
4507 }
4508
4509 nav.botnavigation {
4510     background: #dedcd5 ;
4511     border-top: 1px solid black ;
4512 }
4513 \end{filecontents*}
4514 % \end{Verbatim}% for syntax highlighting
4515 \end{LWRwriteconf}
```

40.7 `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:
```
4516 \begin{LWRwriteconf}
4517 \begin{filecontents*}[overwrite]{sample_project.css}
4518 /* ( --- Start of project.css --- ) */
4519 /* ( --- A sample project-specific CSS file for lwarp --- ) */
```

```
4520
4521 /* Uncomment one of the following: */
4522 @import url("lwarp.css") ;
4523 /* @import url("lwarp_formal.css") ; */
4524 /* @import url("lwarp_sagebrush.css") ; */
4525
4526 /* Project-specific CSS setting follow here. */
4527 /* . . . */
4528
4529 /* ( --- End of project.css --- ) */
4530 \end{filecontents*}
4531 % \end{Verbatim}% for syntax highlighting
4532 \end{LWRwriteconf}
```

40.8 `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:
```
4533 \begin{LWRwriteconf}
4534 \begin{filecontents*}[overwrite]{lwarp.ist}
4535 preamble
4536 "\\begin{theindex}
4537   \\providecommand*\\lettergroupDefault[1]{}
4538   \\providecommand*\\lettergroup[1]{%
4539       \\par\\textbf{#1}\\par
4540       \\nopagebreak
4541   }
4542 "
4543 headings_flag 1
4544 heading_prefix "
4545   \\lettergroup{"
4546 heading_suffix "}"
4547 delim_0 ", \\hyperindexref{"
4548 delim_1 ", \\hyperindexref{"
4549 delim_2 ", \\hyperindexref{"
4550 delim_n "}, \\hyperindexref{"
4551 delim_r "} -- \\hyperindexref{"
4552 delim_t "}"
4553 page_compositor "."
4554 \end{filecontents*}
4555 % \end{Verbatim}% for syntax highlighting
4556 \end{LWRwriteconf}
```

40.9 `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:
```
4557 \begin{LWRwriteconf}
4558 \begin{filecontents*}[overwrite]{lwarp.xdy}
4559 (require "tex/inputenc/latin.xdy")
4560 (merge-rule "\\PS *" "Postscript")
4561 (require "texindy.xdy")
4562 (require "page-ranges.xdy")
4563 (require "book-order.xdy")
4564 (define-location-class "arabic-page-numbers"
4565    ("arabic-numbers") :min-range-length 1)
4566 (require "makeindex.xdy")
4567 (define-attributes (("hyperindexref")))
4568 (markup-locref :open "\hyperindexref{" :close "}")
4569 (markup-locref :open "\hyperindexref{" :close "}" :attr "hyperpage")
4570 (markup-locref :open "\textbf{\hyperindexref{" :close "}}" :attr "textbf")
4571 (markup-locref :open "\textit{\hyperindexref{" :close "}}" :attr "textit")
4572 (define-location-class-order ("roman-page-numbers"
4573                    "arabic-page-numbers"
4574                    "alpha-page-numbers"
4575                    "Roman-page-numbers"
4576                    "Alpha-page-numbers"
4577                    "see"
4578                    "seealso"))
4579 \end{filecontents*}
4580 % \end{Verbatim}% for syntax highlighting
4581 \end{LWRwriteconf}
```

40.10 `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:
```
4582 \begin{LWRwriteconf}
4583 \immediate\openout\LWR@quickfile=lwarp_one_limage.txt
4584 \immediate\write\LWR@quickfile{%
4585    pdfseparate -f \LWRpercent 1 -l \LWRpercent 1 \LWRpercent 4_html.pdf %
4586    \LWR@ImagesDirectory\OSPathSymbol lateximagetemp-\LWRpercent\LWRpercent d.pdf%
4587 }
4588 \immediate\write\LWR@quickfile{%
4589    pdfcrop --hires  --margins \LWRopquote0 1 0 0\LWRopquote\space %
4590    \LWR@ImagesDirectory\OSPathSymbol lateximagetemp-\LWRpercent 1.pdf %
4591    \LWR@ImagesDirectory\OSPathSymbol\LWRpercent 3.pdf%
4592 }
4593 \immediate\write\LWR@quickfile{%
4594    pdftocairo -svg -noshrink \LWR@ImagesDirectory\OSPathSymbol\LWRpercent 3.pdf %
4595    \LWR@ImagesDirectory\OSPathSymbol\LWRpercent 3.svg%
```

```
4596 }
4597 \immediate\write\LWR@quickfile{%
4598     del \LWR@ImagesDirectory\OSPathSymbol\LWRpercent 3.pdf%
4599 }
4600 \immediate\write\LWR@quickfile{%
4601     del \LWR@ImagesDirectory\OSPathSymbol lateximagetemp-\LWRpercent 1.pdf%
4602 }
4603 \immediate\write\LWR@quickfile{exit}
4604 \immediate\closeout\LWR@quickfile
4605 \end{LWRwriteconf}
```

40.11 `lwarp_mathjax.txt`

(*Emulates or patches code by* DAVIDE P. CERVONE.)

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 numbering and starred macros, and use \MathJaxFilename to select the new filename.

Config file:
```
4606 \begin{LWRwriteconf}
4607 \begin{filecontents*}[overwrite]{lwarp_mathjax.txt}
4608 <script>
4609 // Lwarp MathJax emulation code
4610 //
4611 // Based on code by Davide P. Cervone.
4612 // Equation numbering: https://github.com/mathjax/MathJax/issues/2427
4613 // Starred and ifnextchar macros: https://github.com/mathjax/MathJax/issues/2428
4614 // \left, \right delimiters: https://github.com/mathjax/MathJax/issues/2535
4615 //
4616 // Modified by Brian Dunn to adjust equation numbering and add subequations.
4617 //
4618 // LaTeX can use \seteqnumber{subequations?}{section}{number} before each equation.
4619 // subequations? is 0 usually, 1 if inside subequations.
4620 // section is a string printed as-is, or empty.
4621 // number is auto-incremented by MathJax between equations.
4622 //
4623 MathJax = {
4624   subequations: "0",
4625   section: "",
4626   loader: {
4627     load: ['[tex]/tagformat', '[tex]/textmacros'],
4628   },
4629   startup: {
4630     ready() {
4631       //  These would be replaced by import commands if you wanted to make
4632       //  a proper extension.
4633       const Configuration = MathJax._.input.tex.Configuration.Configuration;
4634       const CommandMap = MathJax._.input.tex.SymbolMap.CommandMap;
4635       const Macro = MathJax._.input.tex.Symbol.Macro;
```

```
4636        const TexError = MathJax._.input.tex.TexError.default;
4637        const ParseUtil = MathJax._.input.tex.ParseUtil.default;
4638        const expandable = MathJax._.util.Options.expandable;
4639
4640        //   Insert the replacement string into the TeX string, and check
4641        //   that there haven't been too many maxro substitutions (prevents
4642        //   infinite loops).
4643        const useArgument = (parser, text) => {
4644      parser.string = ParseUtil.addArgs(parser, text, parser.string.slice(parser.i));
4645          parser.i = 0;
4646          if (++parser.macroCount > parser.configuration.options.maxMacros) {
4647            throw new TexError('MaxMacroSub1',
4648            'MathJax maximum macro substitution count exceeded; ' +
4649            'is there a recursive macro call?');
4650          }
4651        }
4652
4653        //   Create the command map for:
4654        //      \ifstar, \ifnextchar, \ifblank, \ifstrequal, \gsub, \seteqnumber
4655        new CommandMap('Lwarp-macros', {
4656          ifstar: 'IfstarFunction',
4657          ifnextchar: 'IfnextcharFunction',
4658          ifblank: 'IfblankFunction',
4659          ifstrequal: 'IfstrequalFunction',
4660          gsubstitute: 'GsubstituteFunction',
4661          seteqnumber: 'SeteqnumberFunction'
4662        }, {
4663          //   This function implements an ifstar macro.
4664          IfstarFunction(parser, name) {
4665            const resultstar = parser.GetArgument(name);
4666            const resultnostar = parser.GetArgument(name);
4667           const star = parser.GetStar();                    // true if there is a *
4668            useArgument(parser, star ? resultstar : resultnostar);
4669          },
4670
4671          //   This function implements an ifnextchar macro.
4672          IfnextcharFunction(parser, name) {
4673            let whichchar = parser.GetArgument(name);
4674            if (whichchar.match(/^(?:0x[0-9A-F]+|[0-9]+)$/i)) {
4675              // $ syntax highlighting
4676              whichchar = String.fromCodePoint(parseInt(whichchar));
4677            }
4678            const resultnextchar = parser.GetArgument(name);
4679            const resultnotnextchar = parser.GetArgument(name);
4680            const gotchar = (parser.GetNext() === whichchar);
4681            useArgument(parser, gotchar ? resultnextchar : resultnotnextchar);
4682          },
4683
4684          // This function implements an ifblank macro.
4685          IfblankFunction(parser, name) {
4686            const blankarg = parser.GetArgument(name);
4687            const resultblank = parser.GetArgument(name);
4688            const resultnotblank = parser.GetArgument(name);
4689            const isblank = (blankarg.trim() == "");
4690            useArgument(parser, isblank ? resultblank : resultnotblank);
4691          },
4692
4693          // This function implements an ifstrequal macro.
4694          IfstrequalFunction(parser, name) {
4695            const strequalfirst = parser.GetArgument(name);
```

```
4696              const strequalsecond = parser.GetArgument(name);
4697              const resultequal = parser.GetArgument(name);
4698              const resultnotequal = parser.GetArgument(name);
4699              const isequal = (strequalfirst == strequalsecond);
4700              useArgument(parser, isequal ? resultequal : resultnotequal);
4701          },
4702
4703          // This function implements a gsub macro.
4704          GsubstituteFunction(parser, name) {
4705              const gsubfirst = parser.GetArgument(name);
4706              const gsubsecond = parser.GetArgument(name);
4707              const gsubthird = parser.GetArgument(name);
4708              let gsubresult=gsubfirst.replace(gsubsecond, gsubthird);
4709              useArgument(parser, gsubresult);
4710          },
4711
4712          //  This function modifies the equation numbers.
4713          SeteqnumberFunction(parser, name) {
4714              //  Get the macro parameters
4715            const star = parser.GetStar();                    // true if there is a *
4716        const optBrackets = parser.GetBrackets(name);  // contents of optional brackets
4717        const newsubequations = parser.GetArgument(name); // the subequations argument
4718        const neweqsection = parser.GetArgument(name);  // the eq section argument
4719        const neweqnumber = parser.GetArgument(name);   // the eq number argument
4720        MathJax.config.subequations=newsubequations ;   // a string with boolean meaning
4721        MathJax.config.section=neweqsection ;           // a string with numeric meaning
4722            parser.tags.counter = parser.tags.allCounter = neweqnumber ;
4723          }
4724
4725      });
4726
4727      //  Create the Lwarp-macros package
4728      Configuration.create('Lwarp-macros', {
4729        handler: {macro: ['Lwarp-macros']}
4730      });
4731
4732      MathJax.startup.defaultReady();
4733
4734      // For forward references:
4735      MathJax.startup.input[0].preFilters.add(({math}) => {
4736        if (math.inputData.recompile){
4737        MathJax.config.subequations = math.inputData.recompile.subequations;
4738            MathJax.config.section = math.inputData.recompile.section;
4739        }
4740      });
4741      MathJax.startup.input[0].postFilters.add(({math}) => {
4742        if (math.inputData.recompile){
4743        math.inputData.recompile.subequations = MathJax.config.subequations;
4744            math.inputData.recompile.section = MathJax.config.section;
4745        }
4746      });
4747
4748      // For \left, \right with unicode-math:
4749      const {DelimiterMap} = MathJax._.input.tex.SymbolMap;
4750      const {Symbol} = MathJax._.input.tex.Symbol;
4751      const {MapHandler} = MathJax._.input.tex.MapHandler;
4752      const delimiter = MapHandler.getMap('delimiter');
4753      delimiter.add('\\lBrack', new Symbol('\\lBrack', '\u27E6'));
4754      delimiter.add('\\rBrack', new Symbol('\\rBrack', '\u27E7'));
4755      delimiter.add('\\lAngle', new Symbol('\\lAngle', '\u27EA'));
```

```
4756          delimiter.add('\\rAngle', new Symbol('\\rAngle', '\u27EB'));
4757          delimiter.add('\\lbrbrak', new Symbol('\\lbrbrak', '\u2772'));
4758          delimiter.add('\\rbrbrak', new Symbol('\\rbrbrak', '\u2773'));
4759          delimiter.add('\\lbag', new Symbol('\\lbag', '\u27C5'));
4760          delimiter.add('\\rbag', new Symbol('\\rbag', '\u27C6'));
4761       delimiter.add('\\llparenthesis', new Symbol('\\llparenthesis', '\u2987'));
4762       delimiter.add('\\rrparenthesis', new Symbol('\\rrparenthesis', '\u2988'));
4763          delimiter.add('\\llangle', new Symbol('\\llangle', '\u2989'));
4764          delimiter.add('\\rrangle', new Symbol('\\rrangle', '\u298A'));
4765          delimiter.add('\\Lbrbrak', new Symbol('\\Lbrbrak', '\u27EC'));
4766          delimiter.add('\\Rbrbrak', new Symbol('\\Rbrbrak', '\u27ED'));
4767          delimiter.add('\\lBrace', new Symbol('\\lBrace', '\u2983'));
4768          delimiter.add('\\rBrace', new Symbol('\\rBrace', '\u2984'));
4769          delimiter.add('\\lParen', new Symbol('\\lParen', '\u2985'));
4770          delimiter.add('\\rParen', new Symbol('\\rParen', '\u2986'));
4771          delimiter.add('\\lbrackubar', new Symbol('\\lbrackubar', '\u298B'));
4772          delimiter.add('\\rbrackubar', new Symbol('\\rbrackubar', '\u298C'));
4773       delimiter.add('\\lbrackultick', new Symbol('\\lbrackultick', '\u298D'));
4774       delimiter.add('\\rbracklrtick', new Symbol('\\rbracklrtick', '\u298E'));
4775       delimiter.add('\\lbracklltick', new Symbol('\\lbracklltick', '\u298F'));
4776       delimiter.add('\\rbrackurtick', new Symbol('\\rbrackurtick', '\u2990'));
4777          delimiter.add('\\langledot', new Symbol('\\langledot', '\u2991'));
4778          delimiter.add('\\rangledot', new Symbol('\\rangledot', '\u2992'));
4779          delimiter.add('\\lparenless', new Symbol('\\lparenless', '\u2993'));
4780          delimiter.add('\\rparengtr', new Symbol('\\rparengtr', '\u2994'));
4781          delimiter.add('\\Lparengtr', new Symbol('\\Lparengtr', '\u2995'));
4782          delimiter.add('\\Rparenless', new Symbol('\\Rparenless', '\u2996'));
4783          delimiter.add('\\lblkbrbrak', new Symbol('\\lblkbrbrak', '\u2997'));
4784          delimiter.add('\\rblkbrbrak', new Symbol('\\rblkbrbrak', '\u2998'));
4785          delimiter.add('\\lvzigzag', new Symbol('\\lvzigzag', '\u29D8'));
4786          delimiter.add('\\rvzigzag', new Symbol('\\rvzigzag', '\u29D9'));
4787          delimiter.add('\\Lvzigzag', new Symbol('\\Lvzigzag', '\u29DA'));
4788          delimiter.add('\\Rvzigzag', new Symbol('\\Rvzigzag', '\u29DB'));
4789       delimiter.add('\\lcurvyangle', new Symbol('\\lcurvyangle', '\u29FC'));
4790       delimiter.add('\\rcurvyangle', new Symbol('\\rcurvyangle', '\u29FD'));
4791          delimiter.add('\\Vvert', new Symbol('\\Vvert', '\u2980'));
4792       }   // ready
4793    },    // startup
4794
4795    tex: {
4796      packages: {'[+]': ['tagformat', 'Lwarp-macros', 'textmacros']},
4797      tags: "ams",
4798          tagformat: {
4799              number: function (n) {
4800                  if(MathJax.config.subequations==0)
4801                      return(MathJax.config.section + n);
4802                  else
4803                       return(MathJax.config.section + String.fromCharCode(96+n));
4804              },
4805          },
4806    }
4807 }
4808 </script>
4809
4810 <script
4811     id="MathJax-script"
4812     src="https://cdn.jsdelivr.net/npm/mathjax@3/es5/tex-svg.js"
4813 ></script>
4814 \end{filecontents*}
4815 % \end{Verbatim}% for syntax highlighting
```

4816 \end{LWRwriteconf}

40.12 `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.

```
4817 \begin{LWRcreatelwarpmk}

4818 \begin{filecontents*}[overwrite]{lwarpmk.lua}
4819 #!/usr/bin/env texlua
4820
4821 -- Copyright 2016-2022 Brian Dunn
4822
4823
4824 printversion = "v0.910"
4825 requiredconfversion = "2" -- also at *lwarpmk.conf
4826
4827 function printhelp ()
4828 print ("lwarpmk: Use lwarpmk -h or lwarpmk --help for help.") ;
4829 end
4830
4831
4832 function printusage ()
4833 --
4834 -- Print the usage of the lwarpmk command:
4835 --
4836 print ( [[
4837
4838 lwarpmk print [-p project]: Compile the print version if necessary.
4839 lwarpmk print1 [-p project]: Forced single compile of the print version.
4840 lwarpmk printindex [-p project]: Process print indexes.
4841 lwarpmk printglossary [-p project]: Process the glossary for the print version.
4842 lwarpmk html [-p project]: Compile the HTML version if necessary.
4843 lwarpmk html1 [-p project]: Forced single compile of the HTML version.
4844 lwarpmk htmlindex [-p project]: Process HTML indexes.
4845 lwarpmk htmlglossary [-p project]: Process the glossary for the html version.
4846 lwarpmk again [-p project]: Touch the source code to trigger recompiles.
4847 lwarpmk limages [-p project]: Process the "lateximages" created by lwarp.sty.
4848 lwarpmk pdftohtml [-p project]:
4849     For use with latexmk or a Makefile:
4850     Converts project_html.pdf to project_html.html and individual HTML files.
4851     Finishes the HTML conversion even if there was a compile error.
4852 lwarpmk pdftosvg <list of file names>: Converts each PDF file to SVG.
```

```
4853 lwarpmk epstopdf <list of file names>: Converts each EPS file to PDF.
4854 lwarpmk clean [-p project]: Remove *.aux, *.toc, *.lof/t,
4855     *.idx, *.ind, *.bbl, *.log, *_html_inc.*, .gl*,
4856     *_html.pdf, *_html.html, *_html.sidetoc
4857 lwarpmk cleanall [-p project]: Remove auxiliary files, project.pdf, *.html
4858 lwarpmk cleanlimages: Removes all images from the "lateximages" directory.
4859 lwarpmk -v: Print the version number.
4860 lwarpmk -h: Print this help message.
4861 lwarpmk --help: Print this help message.
4862
4863 ]] )
4864 -- printconf ()
4865 end
4866
4867
4868 function splitfilename ( pathandfilename )
4869 --
4870 -- Separates out the path and extension from a filename.
4871 -- Returns path, filename with extension, and extension.
4872 -- Ex: thispath, thisfilename, thisextension = splitfilename ("path/to/filename.ext")
4873 --
4874 -- https://www.fhug.org.uk/wiki/wiki/doku.php?id=plugins:code_snippets:
4875 --       split_filename_in_to_path_filename_and_extension
4876 --
4877     if lfs.attributes(pathandfilename,"mode") == "directory" then
4878      local strPath = pathandfilename:gsub("[\\/]$","") -- $ (syntax highlighting)
4879         return strPath.."\\","",""
4880     end
4881     pathandfilename = pathandfilename.."."
4882     return pathandfilename:match("^(.-)([^\\/]-)%.([^\\/%.]-)%.?$")
4883 end
4884
4885
4886 function splitfile (destfile,sourcefile)
4887 --
4888 -- Split one large sourcefile into a number of files,
4889 -- starting with destfile.
4890 -- The file is split at each occurance of <!--|Start file|newfilename|*
4891 -- If lwarp is in use, sets usinglwarp.
4892 --
4893 usinglwarp = false ;
4894 print ("lwarpmk: Splitting " .. sourcefile .. " into " .. destfile) ;
4895 local sfile = io.open(sourcefile)
4896 io.output(destfile)
4897 for line in sfile:lines() do
4898 i,j,copen,cstart,newfilename = string.find (line,"(.*)|(.*)|(.*)|") ;
4899 if ( (i~= nil) and (copen == "<!--") and (cstart == "Start file")) then
4900     -- split the file
4901     io.output(newfilename) ;
4902 else
4903 if ( (i~= nil) and (copen == "<!--") and (cstart == "Using lwarp")) then
4904     -- verified the use of \usepackage{lwarp}
4905     usinglwarp = true ;
4906 else
4907     -- not a splitpoint
4908     io.write (line .. "\n") ;
4909 end end
4910 end -- do
4911 io.close(sfile)
4912 if ( usinglwarp == false ) then
```

```
4913     print ("lwarpmk: ===")
4914     print ("lwarpmk: \\usepackage{lwarp} was not detected.")
4915     print ("lwarpmk: The HTML output will not be correct.")
4916     print ("lwarpmk: Ensured that \\usepackage{lwarp} is enabled,")
4917     print ("lwarpmk: then lwarpmk print and lwarpmk html again.")
4918     print ("lwarpmk: ===")
4919 end
4920 end -- function
4921
4922
4923 function cvalueerror ( line, linenum , cvalue )
4924 --
4925 -- Incorrect value, so print an error and exit.
4926 --
4927     print ("lwarpmk: ===")
4928     print ("lwarpmk: " .. linenum .. " : " .. line ) ;
4929     print (
4930         "lwarpmk: incorrect variable value \"" .. cvalue ..
4931         "\" in lwarpmk.conf.\n"
4932     ) ;
4933     print ("lwarpmk: ===")
4934 --    printconf () ;
4935     os.exit(1) ;
4936 end
4937
4938
4939 function printhowtorecompile ()
4940 -- Tells the user how to recompile to regenerate the configuration files.
4941   print ("lwarpmk: The configuration files lwarpmk.conf and "..sourcename..".lwarpmkconf" )
4942     print ("lwarpmk:   must be updated.  To do so, recompile" )
4943     print ("lwarpmk:   " , sourcename..".tex" )
4944     if ( printlatexcmd == "" ) then
4945         print ("lwarpmk:   using xe/lua/pdflatex," )
4946     else
4947         print ("lwarpmk:   using the command:")
4948         print ("lwarpmk:   " , printlatexcmd )
4949     end
4950     print ("lwarpmk:   then use lwarpmk again.")
4951 end -- printhowtorecompile
4952
4953
4954 function ignoreconf ()
4955 -- Global argument index
4956 argindex = 2
4957 end
4958
4959 function loadconf ()
4960 --
4961 -- Load settings from the project's "lwarpmk.conf" file:
4962 --
4963 -- Default configuration filename:
4964 local conffile = "lwarpmk.conf"
4965 local confroot = "lwarpmk"
4966 -- Global argument index
4967 argindex = 2
4968 -- Optional configuration filename:
4969 if ( arg[argindex] == "-p" ) then
4970     argindex = argindex + 1
4971     confroot = arg[argindex]
4972     conffile = confroot..".lwarpmkconf"
```

```
4973     argindex = argindex + 1
4974 end
4975 -- Additional defaults:
4976 confversion = "0"
4977 opsystem = "Unix"
4978 imagesdirectory = "lateximages"
4979 imagesname = "image-"
4980 latexmk = "false"
4981 printlatexcmd = ""
4982 HTMLlatexcmd = ""
4983 printindexcmd = ""
4984 HTMLindexcmd = ""
4985 latexmkindexcmd = ""
4986 -- to be removed:
4987 -- indexprog = "makeindex"
4988 -- makeindexstyle = "lwarp.ist"
4989 -- xindylanguage = "english"
4990 -- xindycodepage = "utf8"
4991 -- xindystyle = "lwarp.xdy"
4992 -- pdftotextenc = "UTF-8"
4993 glossarycmd = "makeglossaries"
4994 -- Verify the file exists:
4995 if (lfs.attributes(conffile,"mode")==nil) then
4996     -- file not exists
4997     print ("lwarpmk: ===")
4998     print ("lwarpmk: File \"" .. conffile .."\" does not exist.")
4999     print ("lwarpmk: Move to the project's source directory,")
5000     print ("lwarpmk: recompile using pdflatex, xelatex, or lualatex,")
5001     print ("lwarpmk: then try using lwarpmk again.")
5002     if ( arg[argindex] ~= nil ) then
5003         print (
5004             "lwarpmk: (\"" .. confroot ..
5005             "\" does not appear to be a project name.)"
5006         )
5007     end
5008     print ("lwarpmk: ===")
5009     printhelp () ;
5010     os.exit(1) -- exit the entire lwarpmk script
5011 else -- file exists
5012 -- Read the file:
5013 print ("lwarpmk: Reading " .. conffile ..".")
5014 local cfile = io.open(conffile)
5015 -- Scan each line, parsing each line as: name = [[string]]
5016 local linenum = 0
5017 for line in cfile:lines() do -- scan lines
5018 linenum = linenum + 1
5019 i,j,cvarname,cvalue = string.find (line,"([%w-_]*)%s*=%s*%[%[([^%]]*)%]%]") ;
5020 -- Error if incorrect enclosing characters:
5021 if ( i == nil ) then
5022     print ("lwarpmk: ===")
5023     print ("lwarpmk: " ..  linenum .. " : " .. line ) ;
5024     print ("lwarpmk: Incorrect entry in " .. conffile ..".\n" ) ;
5025     print ("lwarpmk: ===")
5026 --    printconf () ;
5027     os.exit(1) ;
5028 end -- nil
5029 if ( cvarname == "confversion" ) then
5030     confversion = cvalue
5031 elseif ( cvarname == "opsystem" ) then
5032     -- Verify choice of opsystem:
```

```
5033     if ( (cvalue == "Unix") or (cvalue == "Windows") ) then
5034         opsystem = cvalue
5035     else
5036         cvalueerror ( line, linenum , cvalue )
5037     end
5038 elseif ( cvarname == "sourcename" ) then sourcename = cvalue
5039 elseif ( cvarname == "homehtmlfilename" ) then homehtmlfilename = cvalue
5040 elseif ( cvarname == "htmlfilename" ) then htmlfilename = cvalue
5041 elseif ( cvarname == "imagesdirectory" ) then imagesdirectory = cvalue
5042 elseif ( cvarname == "imagesname" ) then imagesname = cvalue
5043 elseif ( cvarname == "latexmk" ) then latexmk = cvalue
5044 elseif ( cvarname == "printlatexcmd" ) then printlatexcmd = cvalue
5045 elseif ( cvarname == "HTMLlatexcmd" ) then HTMLlatexcmd = cvalue
5046 elseif ( cvarname == "printindexcmd" ) then printindexcmd = cvalue
5047 elseif ( cvarname == "HTMLindexcmd" ) then HTMLindexcmd = cvalue
5048 elseif ( cvarname == "latexmkindexcmd" ) then latexmkindexcmd = cvalue
5049 elseif ( cvarname == "glossarycmd" ) then glossarycmd = cvalue
5050 elseif ( cvarname == "pdftotextenc" ) then pdftotextenc = cvalue
5051 else
5052     print ("lwarpmk: ===")
5053     print ("lwarpmk: " .. linenum .. " : " .. line ) ;
5054     print (
5055         "lwarpmk: Incorrect variable name \"" .. cvarname .. "\" in " ..
5056         conffile ..".\n"
5057     ) ;
5058     print ("lwarpmk: ===")
5059 --    printconf () ;
5060 os.exit(1) ;
5061 end -- cvarname
5062 end -- do scan lines
5063 io.close(cfile)
5064 end -- file exists
5065 -- Error if sourcename is "lwarp".
5066 -- This could happen if a local copy of lwarp has recently been recompiled.
5067 if sourcename=="lwarp" then
5068     print ("lwarpmk: ===")
5069   print ("lwarpmk: lwarp.sty has recently been recompiled in this directory,")
5070   print ("lwarpmk: and \"lwarpmk.conf\" is no longer set for your own project.")
5071    print ("lwarpmk:   (Perhaps you are not in your project's directory?)")
5072    print ("lwarpmk: In your project directory, recompile your project")
5073    print ("lwarpmk: using pdf/lua/xelatex <projectname>.")
5074   print ("lwarpmk: After a recompile, \"lwarpmk.conf\" will be set for your project,")
5075    print ("lwarpmk: and you may again use lwarpmk.")
5076    print ("lwarpmk: ===")
5077    os.exit(1)
5078 end -- sourcename of "lwarp"
5079 -- Select some operating-system commands:
5080 if opsystem=="Unix" then  -- For Unix / Linux / Mac OS:
5081     rmname = "rm"
5082     mvname = "mv"
5083     cpname = "cp"
5084     touchnamepre = "touch"
5085     touchnamepost = ""
5086     newtouchname = "touch"
5087     dirslash = "/"
5088     opquote= "\'"
5089     cmdgroupopenname = " ( "
5090     cmdgroupclosename = " ) "
5091     seqname = " && "
5092     bgname = " &"
```

```
5093 elseif opsystem=="Windows" then -- For Windows
5094     rmname = "DEL"
5095     mvname = "MOVE"
5096     cpname = "COPY"
5097     touchnamepre = "COPY /b"
5098     touchnamepost = "+,,"
5099     newtouchname = "echo empty >"
5100     dirslash = "\\"
5101     opquote= "\""
5102     cmdgroupopenname = ""
5103     cmdgroupclosename = ""
5104     seqname = " & "
5105     bgname = ""
5106 else
5107     print ("lwarpmk: ===")
5108     print ("lwarpmk: Select Unix or Windows for opsystem." )
5109     print ("lwarpmk: ===")
5110     os.exit(1)
5111 end --- for Windows
5112 -- Warning if the operating system does not appear to be correct,
5113 -- in case files were transferred to another system.
5114 if ( (package.config:sub(1,1)) ~= dirslash ) then
5115     print ("lwarpmk: ===")
5116   print ("lwarpmk: It appears that lwarpmk.conf is for a different operating system." )
5117     printhowtorecompile ()
5118     print ("lwarpmk: ===")
5119     os.exit(1)
5120 end
5121 -- Error if the configuration file's version is not current:
5122 if ( confversion ~= requiredconfversion ) then
5123     print ("lwarpmk: ===")
5124     printhowtorecompile ()
5125     print ("lwarpmk: ===")
5126     os.exit(1)
5127 end
5128 end -- loadconf
5129
5130
5131 function executecheckerror ( executecommands , errormessage )
5132 --
5133 -- Execute an operating system call,
5134 -- and maybe exit with an error message.
5135 --
5136 local err
5137 err = os.execute ( executecommands )
5138 if ( err ~= 0 ) then
5139     print ("lwarpmk: ===")
5140     print ("lwarpmk: " .. errormessage )
5141     print ("lwarpmk: ===")
5142     os.exit(1)
5143 end
5144 end -- executecheckerror
5145
5146
5147 function refreshdate ()
5148 os.execute(touchnamepre .. " " .. sourcename .. ".tex " .. touchnamepost)
5149 end
5150
5151
5152
```

```
5153 function reruntoget (filesource)
5154 --
5155 -- Scan the LaTeX log file for the phrase "Rerun to get",
5156 -- indicating that the file should be compiled again.
5157 -- Return true if found.
5158 --
5159 local fsource = io.open(filesource)
5160 for line in fsource:lines() do
5161 if ( string.find(line,"Rerun to get") ~= nil ) then
5162     io.close(fsource)
5163     return true
5164 end -- if
5165 end -- do
5166 io.close(fsource)
5167 return false
5168 end
5169
5170
5171
5172 function onetime (latexcmd, fsuffix)
5173 --
5174 -- Compile one time, return true if should compile again.
5175 -- fsuffix is "" for print, "_html" for HTML output.
5176 --
5177 print("lwarpmk: Compiling with: " .. latexcmd)
5178 executecheckerror (
5179     latexcmd ,
5180     "Compile error."
5181 )
5182 return (reruntoget(sourcename .. fsuffix .. ".log") ) ;
5183 end
5184
5185
5186 function manytimes (latexcmd, fsuffix)
5187 --
5188 -- Compile up to five times.
5189 -- fsuffix is "" for print, "_html" for HTML output
5190 --
5191 if onetime(latexcmd, fsuffix) == true then
5192 if onetime(latexcmd, fsuffix) == true then
5193 if onetime(latexcmd, fsuffix) == true then
5194 if onetime(latexcmd, fsuffix) == true then
5195 if onetime(latexcmd, fsuffix) == true then
5196 end end end end end
5197 end
5198
5199
5200 function verifyfileexists (filename)
5201 --
5202 -- Exit if the given file does not exist.
5203 --
5204 if (lfs.attributes ( filename , "modification" ) == nil ) then
5205     print ("lwarpmk: ===")
5206     print ("lwarpmk: " .. filename .. " not found." ) ;
5207     print ("lwarpmk: ===")
5208     os.exit (1) ;
5209 end
5210 end
5211
5212
```

```
5213
5214 function pdftohtml ()
5215 --
5216 -- Convert <project>_html.pdf into HTML files:
5217 --
5218 -- Convert to text:
5219 print ("lwarpmk: Converting " .. sourcename
5220     .."_html.pdf to " .. sourcename .. "_html.html")
5221 err = os.execute("pdftotext  -enc " .. pdftotextenc .. "  -nopgbrk  -layout "
5222     .. sourcename .. "_html.pdf " .. sourcename .. "_html.html")
5223 if ( err ~= 0 ) then
5224     print ("lwarpmk: ===")
5225     print ("lwarpmk: Ensure that the Poppler utilities are installed." )
5226   print ("lwarpmk: See the Lwarp manual: 'Installing additional utilities'." )
5227     print ("lwarpmk: ===")
5228     os.exit(1)
5229 end
5230 -- Split the result into individual HTML files:
5231 splitfile (homehtmlfilename .. ".html" , sourcename .. "_html.html")
5232 end
5233
5234
5235 function removeaux ()
5236 --
5237 -- Remove auxiliary files:
5238 -- All .aux files are removed since there may be many bbl*.aux files.
5239 -- Also removes sourcename_html.pdf, sourcename_html.html,
5240 -- and sourcename_html.sidetoc, plus comment_*.cut.
5241 --
5242 os.execute ( rmname .. " *.aux " ..
5243     sourcename .."".toc " .. sourcename .. "_html.toc " ..
5244     sourcename .."".lof " .. sourcename .. "_html.lof " ..
5245     sourcename .."".lot " .. sourcename .. "_html.lot " ..
5246     sourcename .."".bbl " .. sourcename .. "_html.bbl " ..
5247     " *.idx " ..
5248     " *.ind " ..
5249     sourcename .."".ps " .. sourcename .."_html.ps " ..
5250     sourcename .."".log " .. sourcename .. "_html.log " ..
5251     sourcename .."".gl* " .. sourcename .. "_html.gl* " ..
5252     sourcename .. "_html.pdf " ..
5253     sourcename .. "_html.html " ..
5254     sourcename .. "_html.sidetoc " ..
5255     " *_html_inc.* " ..
5256     " comment_*.cut"
5257     )
5258 end
5259
5260 function checkhtmlpdfexists ()
5261 --
5262 -- Error if the HTML document does not exist.
5263 -- The lateximages are drawn from the HTML PDF version of the document,
5264 -- so "lwarpmk html" must be done before "lwarpmk limages".
5265 --
5266 local htmlpdffile = io.open(sourcename .. "_html.pdf", "r")
5267 if ( htmlpdffile == nil ) then
5268     print ("")
5269     print ("lwarpmk: ===")
5270     print ("lwarpmk: The HTML version of the document does not exist.")
5271     print ("lwarpmk: Enter \"lwarpmk html\" to compile the HTML version.")
5272     print ("lwarpmk: ===")
```

```
5273     os.exit(1)
5274 end
5275 io.close (htmlpdffile)
5276 end -- checkhtmlpdfexists
5277
5278
5279 function warnlimages ()
5280 --
5281 -- Warning of a missing <sourcename>-images.txt file:
5282     print ("lwarpmk: ===")
5283     print ("lwarpmk: \"" .. sourcename .. "-images.txt\" does not exist.")
5284   print ("lwarpmk: Your project does not use SVG math or other lateximages,")
5285     print ("lwarpmk: or the file has been deleted somehow.")
5286     print ("lwarpmk: Use \"lwarpmk html1\" to recompile your project")
5287     print ("lwarpmk: and recreate \"" .. sourcename .. "-images.txt\".")
5288   print ("lwarpmk: If your project does not use SVG math or other lateximages,")
5289   print ("lwarpmk: then \"" .. sourcename .. "-images.txt\" will never exist, and")
5290     print ("lwarpmk: \"lwarpmk limages\" will not be necessary.")
5291     print ("lwarpmk: ===")
5292 end -- warnlimages
5293
5294
5295 function warnlimagesrecompile ()
5296 -- Warning if must recompile before creating limages:
5297     print ("")
5298     print ("lwarpmk: ===")
5299     print ("lwarpmk: Cross-references are not yet correct.")
5300   print ("lwarpmk: The document must be recompiled before creating the lateximages.")
5301   print ("lwarpmk: Enter \"lwarpmk html1\" again, then try \"lwarpmk limages\" again.")
5302     print ("lwarpmk: ===")
5303 end --warnlimagesrecompile
5304
5305
5306 function checklimages ()
5307 --
5308 -- Check <sourcename>.txt to see if need to recompile first.
5309 -- If any entry has a page number of zero, then there were incorrect images.
5310 --
5311 print ("lwarpmk: Checking for a valid " .. sourcename .. "-images.txt file.")
5312 local limagesfile = io.open(sourcename .. "-images.txt", "r")
5313 if ( limagesfile == nil ) then
5314     warnlimages ()
5315     os.exit(1)
5316 end
5317 -- Track warning to recompile if find a page 0
5318 local pagezerowarning = false
5319 -- Scan <sourcename>.txt
5320 for line in limagesfile:lines() do
5321     -- lwimgpage is the page number in the PDF which has the image
5322     -- lwimghash is true if this filename is a hash
5323     -- lwimgname is the lateximage filename root to assign for the image
5324     i,j,lwimgpage,lwimghash,lwimgname = string.find (line,"|(.*)|(.*)|(.*)|")
5325     -- For each entry:
5326     if ( (i~=nil) ) then
5327         -- If the page number is 0, image references are incorrect
5328         --  and must recompile the soure document:
5329         if ( lwimgpage == "0" ) then
5330             pagezerowarning = true
5331         end
5332     end -- if i~=nil
```

```
5333 end -- do
5334 -- The last line should be |end|end|end|.
5335 -- If not, the compile must have aborted, and the images are incomplete.
5336 if ( lwimgpage ~= "end" ) then
5337     warnlimagesrecompile()
5338     os.exit(1) ;
5339 end
5340 if ( pagezerowarning ) then
5341     warnlimagesrecompile()
5342     os.exit(1) ;
5343 end -- pagezerowarning
5344 end -- checklimages
5345
5346
5347 function createuniximage ( lwimgfullname )
5348 --
5349 -- Create one lateximage for Unix / Linux / Mac OS.
5350 --
5351 executecheckerror (
5352     cmdgroupopenname ..
5353     "pdfseparate -f " .. lwimgpage .. " -l " .. lwimgpage .. " " ..
5354         sourcename .."_html.pdf " ..
5355         imagesdirectory .. dirslash .."lateximagetemp-%d" .. ".pdf" ..
5356         seqname ..
5357     -- Crop the image:
5358     "pdfcrop --hires --margins \"0 1 0 0\" " .. imagesdirectory .. dirslash .. "lateximagetemp-" ..
5359         lwimgpage .. ".pdf " ..
5360         imagesdirectory .. dirslash .. lwimgname .. ".pdf" ..
5361         seqname ..
5362     -- Convert the image to svg:
5363     "pdftocairo -svg -noshrink " .. imagesdirectory .. dirslash .. lwimgname .. ".pdf " ..
5364         imagesdirectory .. dirslash .. lwimgname .."svg" ..
5365         seqname ..
5366     -- Remove the temporary files:
5367     rmname .. " " .. imagesdirectory .. dirslash .. lwimgname .. ".pdf" .. seqname ..
5368     rmname .. " " .. imagesdirectory .. dirslash .. "lateximagetemp-" .. lwimgpage .. ".pdf" ..
5369     cmdgroupclosename .. " >/dev/null " .. bgname
5370     ,
5371     "File error trying to convert " .. lwimgfullname
5372 )
5373 -- Every 32 images, wait for completion at below normal priority,
5374 --   allowing other image tasks to catch up.
5375 numimageprocesses = numimageprocesses + 1
5376 if ( numimageprocesses > 32 ) then
5377     numimageprocesses = 0
5378     print ( "lwarpmk: waiting" )
5379     executecheckerror ( "wait" , "File error trying to wait.")
5380 end
5381 end -- createuniximage
5382
5383
5384 function createwindowsimage ( lwimgfullname )
5385 --
5386 -- Create one lateximage for Windows.
5387 --
5388 -- Every 32 images, wait for completion at below normal priority,
5389 --   allowing other image tasks to catch up.
5390 numimageprocesses = numimageprocesses + 1
5391 if ( numimageprocesses > 32 ) then
5392     numimageprocesses = 0
```

```
5393    thiswaitcommand = "/WAIT /BELOWNORMAL"
5394    print ( "lwarpmk: waiting" )
5395 else
5396    thiswaitcommand = ""
5397 end
5398 -- Execute the image generation command
5399 executecheckerror (
5400    "start /B " .. thiswaitcommand .. " \"\" lwarp_one_limage " ..
5401    lwimgpage .. " " ..
5402    lwimghash .. " " ..
5403    lwimgname .. " " ..
5404    sourcename .. " <nul >nul"
5405    ,
5406    "File error trying to create image."
5407 )
5408 end -- createwindowsimage
5409
5410
5411 function createonelateximage ( line )
5412 --
5413 -- Given the next line of <sourcename>.txt, convert a single image.
5414 --
5415 -- lwimgpage is the page number in the PDF which has the image
5416 -- lwimghash is true if this filename is a hash
5417 -- lwimgname is the lateximage filename root to assign for the image
5418 i,j,lwimgpage,lwimghash,lwimgname = string.find (line,"|(.*)|(.*)|(.*)|")
5419 -- For each entry:
5420 if ( (i~=nil) ) then
5421    -- Skip if the page number is 0:
5422    if ( lwimgpage == "0" ) then
5423        pagezerowarning = true
5424    -- Skip if the page number is "end":
5425    else if ( lwimgpage == "end" ) then
5426    else
5427        -- Skip is this image is hashed and already exists:
5428      local lwimgfullname = imagesdirectory .. dirslash .. lwimgname .. ".svg"
5429        if (
5430            (lwimghash ~= "true") or
5431            (lfs.attributes(lwimgfullname,"mode")==nil) -- file not exists
5432        )
5433        then -- not hashed or not exists:
5434            -- Print the name of the file being generated:
5435            print ( "lwarpmk: " .. lwimgname )
5436       -- Touch/create the dest so that only once instance tries to build it:
5437            executecheckerror (
5438                newtouchname .. " " .. lwimgfullname ,
5439                "File error trying to touch " .. lwimgfullname
5440            )
5441            -- Separate out the image into its own single-page pdf:
5442            if opsystem=="Unix" then
5443                createuniximage (lwimgfullname)
5444            elseif opsystem=="Windows" then
5445                createwindowsimage (lwimgfullname)
5446            end
5447        end -- not hashed or not exists
5448    end -- not page "end"
5449    end -- not page 0
5450 end -- not nil
5451 end -- createonelateximage
5452
```

```
5453
5454 function createlateximages ()
5455 --
5456 -- Create lateximages based on <sourcename>-images.txt:
5457 --
5458 -- See if the document must be recompiled first:
5459 checklimages ()
5460 -- See if the HTML version exists:
5461 checkhtmlpdfexists ()
5462 -- Attempt to create the lateximages:
5463 print ("lwarpmk: Creating lateximages.")
5464 local limagesfile = io.open(sourcename .. "-images.txt", "r")
5465 if ( limagesfile == nil ) then
5466     warnlimages ()
5467     os.exit(1)
5468 end
5469 -- Create the lateximages directory, ignore error if already exists
5470 err = os.execute("mkdir " .. imagesdirectory)
5471 -- For Windows, create lwarp_one_limage.cmd from lwarp_one_limage.txt:
5472 if opsystem=="Windows" then
5473     executecheckerror (
5474         cpname .. " lwarp_one_limage.txt lwarp_one_limage.cmd" ,
5475       "File error trying to copy lwarp_one_limage.txt to lwarp_one_limage.cmd"
5476     )
5477 end -- create lwarp_one_limage.cmd
5478 -- Track the number of parallel processes
5479 numimageprocesses = 0
5480 -- Track warning to recompile if find a page 0
5481 pagezerowarning = false
5482 -- Scan <sourcename>.txt
5483 for line in limagesfile:lines() do
5484     createonelateximage ( line )
5485 end -- do
5486 io.close(limagesfile)
5487 print ( "lwarpmk limages: ===")
5488 print ( "lwarpmk limages: Wait a moment for the images to complete" )
5489 print ( "lwarpmk limages:   before reloading the page." )
5490 print ( "lwarpmk limages: ===")
5491 print ( "lwarpmk limages: Done." )
5492 if ( pagezerowarning == true ) then
5493     print ( "lwarpmk limages: WARNING: Images will be incorrect." )
5494     print ( "lwarpmk limages:   Enter \"lwarpmk cleanlimages\", then" )
5495     print ( "lwarpmk limages:   recompile the document one more time, then" )
5496     print ( "lwarpmk limages:   repeat \"lwarpmk images\" again." )
5497 end -- pagezerowarning
5498 end -- function
5499
5500
5501 function convertepstopdf ()
5502 --
5503 -- Converts EPS files to PDF files.
5504 -- The filenames are arg[argindex] and up.
5505 -- arg[1] is the command "epstopdf".
5506 --
5507 ignoreconf ()
5508 for i = argindex , #arg do
5509     if (lfs.attributes(arg[i],"mode")==nil) then
5510         print ("lwarpmk: File \"" .. arg[i] .. "\" does not exist.")
5511     else
5512         print ("lwarpmk: Converting \"" .. arg[i] .. "\"")
```

```
5513        thispath, thisfilename, thisextension = splitfilename(arg[i])
5514        if ( thispath == nil ) then
5515            os.execute ( "epstopdf " .. arg[i] )
5516        else
5517            os.execute (
5518                "epstopdf " ..
5519                thispath .. thisfilename .. "." .. thisextension .. " " ..
5520                thispath .. thisfilename .. ".pdf"
5521            )
5522        end
5523    end -- if
5524 end -- do
5525 end --function
5526
5527
5528 function convertpdftosvg ()
5529 --
5530 -- Converts PDF files to SVG files.
5531 -- The filenames are arg[argindex] and up.
5532 -- arg[1] is the command "pdftosvg".
5533 --
5534 ignoreconf ()
5535 for i = argindex , #arg do
5536    if (lfs.attributes(arg[i],"mode")==nil) then
5537        print ("lwarpmk: File \"" .. arg[i] .. "\" does not exist.")
5538    else
5539        print ("lwarpmk: Converting \"" .. arg[i] .. "\"")
5540        thispath, thisfilename, thisextension = splitfilename(arg[i])
5541        if ( thispath == nil ) then
5542            os.execute ( "pdftocairo -svg " .. arg[i] )
5543        else
5544            os.execute (
5545                "pdftocairo -svg " ..
5546                thispath .. thisfilename .. "." .. thisextension .. " " ..
5547                thispath .. thisfilename .. ".svg"
5548            )
5549        end
5550    end -- if
5551 end -- do
5552 end --function
5553
5554
5555 -- Force an update and conclude processing:
5556 function updateanddone ()
5557 print ("lwarpmk: Forcing an update of " .. sourcename ..".tex.")
5558 refreshdate ()
5559 print ("lwarpmk: " .. sourcename ..".tex is ready to be recompiled.")
5560 print ("lwarpmk: Done.")
5561 end -- function
5562
5563
5564 -- Start of the main code: --
5565
5566
5567 -- lwarpmk --version :
5568
5569 if (arg[1] == "--version") then
5570 print ( "lwarpmk: " .. printversion )
5571
5572 else -- not --version
```

```
5573
5574
5575 -- print intro:
5576
5577 print ("lwarpmk: " .. printversion .. " Automated make for the LaTeX Lwarp package.")
5578
5579
5580 -- lwarpmk print:
5581
5582 if arg[1] == "print" then
5583 loadconf ()
5584 if ( latexmk == "true" ) then
5585     print ("lwarpmk: Compiling with: " .. printlatexcmd)
5586     executecheckerror (
5587         printlatexcmd ,
5588         "Compile error."
5589     )
5590     print ("lwarpmk: Done.")
5591 else -- not latexmk
5592     verifyfileexists (sourcename .. ".tex") ;
5593     -- See if up to date:
5594     if (
5595         ( lfs.attributes ( sourcename .. ".pdf" , "modification" ) == nil ) or
5596         (
5597             lfs.attributes ( sourcename .. ".tex" , "modification" ) >
5598             lfs.attributes ( sourcename .. ".pdf" , "modification" )
5599         )
5600     ) then
5601         -- Recompile if not yet up to date:
5602         manytimes(printlatexcmd, "")
5603         print ("lwarpmk: Done.") ;
5604     else
5605         print ("lwarpmk: " .. sourcename .. ".pdf is up to date.") ;
5606     end
5607 end -- not latexmk
5608
5609
5610 -- lwarpmk print1:
5611
5612 elseif arg[1] == "print1" then
5613     loadconf ()
5614     verifyfileexists (sourcename .. ".tex") ;
5615     onetime(printlatexcmd, "")
5616     print ("lwarpmk: Done.") ;
5617
5618
5619 -- lwarpmk printindex:
5620 -- Compile the index then touch the source
5621 -- to trigger a recompile of the document:
5622
5623 elseif arg[1] == "printindex" then
5624 loadconf ()
5625 os.execute ( printindexcmd )
5626 print ("lwarpmk: -------")
5627 updateanddone ()
5628
5629
5630 -- lwarpmk printglossary:
5631 -- Compile the glossary then touch the source
5632 -- to trigger a recompile of the document:
```

```
5633
5634 elseif arg[1] == "printglossary" then
5635 loadconf ()
5636 print ("lwarpmk: Processing the glossary.")
5637
5638 os.execute(glossarycmd .. " " .. sourcename)
5639 updateanddone ()
5640
5641
5642 -- lwarpmk html:
5643
5644 elseif arg[1] == "html" then
5645 loadconf ()
5646 if ( latexmk == "true" ) then
5647     print ("lwarpmk: Compiling with: " .. HTMLlatexcmd)
5648     executecheckerror (
5649         HTMLlatexcmd ,
5650         "Compile error."
5651     )
5652     pdftohtml ()
5653     print ("lwarpmk: Done.")
5654 else -- not latexmk
5655     verifyfileexists ( sourcename .. ".tex" ) ;
5656     -- See if exists and is up to date:
5657     if (
5658      ( lfs.attributes ( homehtmlfilename .. ".html" , "modification" ) == nil ) or
5659         (
5660             lfs.attributes ( sourcename .. ".tex" , "modification" ) >
5661             lfs.attributes ( homehtmlfilename .. ".html" , "modification" )
5662         )
5663     ) then
5664         -- Recompile if not yet up to date:
5665         manytimes(HTMLlatexcmd, "_html")
5666         pdftohtml ()
5667         print ("lwarpmk: Done.")
5668     else
5669         print ("lwarpmk: " .. homehtmlfilename .. ".html is up to date.")
5670     end
5671 end -- not latexmk
5672
5673
5674 -- lwarpmk html1:
5675
5676 elseif arg[1] == "html1" then
5677     loadconf ()
5678     verifyfileexists ( sourcename .. ".tex" ) ;
5679     onetime(HTMLlatexcmd, "_html")
5680     pdftohtml ()
5681     print ("lwarpmk: Done.")
5682
5683
5684 -- lwarpmk pdftohtml:
5685 elseif arg[1] == "pdftohtml" then
5686     loadconf ()
5687     pdftohtml ()
5688
5689
5690 -- lwarpmk htmlindex:
5691 -- Compile the index then touch the source
5692 -- to trigger a recompile of the document:
```

```
5693
5694 elseif arg[1] == "htmlindex" then
5695 loadconf ()
5696 os.execute ( HTMLindexcmd )
5697 print ("lwarpmk: -------")
5698 updateanddone ()
5699
5700
5701 -- lwarpmk htmlglossary:
5702 -- Compile the glossary then touch the source
5703 -- to trigger a recompile of the document.
5704 -- The <sourcename>.xdy file is created by the glossaries package.
5705
5706 elseif arg[1] == "htmlglossary" then
5707 loadconf ()
5708 print ("lwarpmk: Processing the glossary.")
5709 os.execute(glossarycmd .. " " .. sourcename .. "_html")
5710 updateanddone ()
5711
5712
5713 -- lwarpmk limages:
5714 -- Scan the <sourcename>.txt file to create lateximages.
5715
5716 elseif arg[1] == "limages" then
5717 loadconf ()
5718 print ("lwarpmk: Processing images.")
5719 createlateximages ()
5720 print ("lwarpmk: Done.")
5721
5722
5723 -- lwarpmk again:
5724 -- Touch the source to trigger a recompile.
5725
5726 elseif arg[1] == "again" then
5727 loadconf ()
5728 updateanddone ()
5729
5730
5731 -- lwarpmk clean:
5732 -- Remove project.aux, .toc, .lof, .lot, .log, *.idx, *.ind, *_html_inc.*, .gl*
5733
5734 elseif arg[1] == "clean" then
5735 loadconf ()
5736 removeaux ()
5737 print ("lwarpmk: Done.")
5738
5739
5740 -- lwarpmk cleanall
5741 -- Remove project.aux, .toc, .lof, .lot, .log, *.idx, *.ind, *_html_inc.*, .gl*
5742 --    and also project.pdf, project.dvi, *.html
5743
5744 elseif arg[1] == "cleanall" then
5745 loadconf ()
5746 removeaux ()
5747 os.execute ( rmname .. " " ..
5748     sourcename .. ".pdf " .. sourcename .. "_html.pdf " ..
5749     sourcename .. ".dvi " .. sourcename .. "_html.dvi " ..
5750     "*.html"
5751     )
5752 print ("lwarpmk: Done.")
```

```
5753
5754
5755 -- lwarpmk cleanlimages
5756 -- Remove images from the imagesdirectory.
5757
5758 elseif arg[1] == "cleanlimages" then
5759 loadconf ()
5760 os.execute ( rmname .. " " .. imagesdirectory .. dirslash .. "*" )
5761 print ("lwarpmk: Done.")
5762
5763 -- lwarpmk epstopdf <list of file names>
5764 -- Convert EPS files to PDF using epstopdf
5765 elseif arg[1] == "epstopdf" then
5766 convertepstopdf ()
5767 print ("lwarpmk: Done.")
5768
5769
5770 -- lwarpmk pdftosvg <list of file names>
5771 -- Convert PDF files to SVG using pdftocairo
5772 elseif arg[1] == "pdftosvg" then
5773 convertpdftosvg ()
5774 print ("lwarpmk: Done.")
5775
5776
5777 -- lwarpmk with no argument :
5778
5779 elseif (arg[1] == nil) then
5780 printhelp ()
5781
5782
5783 -- lwarpmk -v:
5784
5785 elseif (arg[1] == "-v" ) then
5786 -- The version number has already been printed
5787 -- by the lwarpmk intro.
5788
5789 -- lwarpmk -h or lwarpmk --help :
5790
5791 elseif (arg[1] == "-h" ) or (arg[1] == "--help") then
5792 printusage ()
5793
5794
5795 -- Unknown command:
5796
5797 else
5798 printhelp ()
5799 print ("\nlwarpmk: ****** Unknown command \""..arg[1].."\". ******\n")
5800 end
5801
5802 end -- not --version
5803 \end{filecontents*}
5804 % \end{Verbatim}% for syntax highlighting

5805 \end{LWRcreatelwarpmk}
```

41 Stacks

5806 `\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.

41.1 Assigning depths

initial depths for empty stack entries:

5807 `\newcommand*{\LWR@depthnone}{-5}`

All sectioning depths are deeper than `LWR@depthfinished`:

5808 `\newcommand*{\LWR@depthfinished}{-4}`
5809 `\newcommand*{\LWR@depthbook}{-2}`
5810 `\newcommand*{\LWR@depthpart}{-1}`
5811 `\newcommand*{\LWR@depthchapter}{0}`
5812 `\newcommand*{\LWR@depthsection}{1}`
5813 `\newcommand*{\LWR@depthsubsection}{2}`
5814 `\newcommand*{\LWR@depthsubsubsection}{3}`
5815 `\newcommand*{\LWR@depthparagraph}{4}`
5816 `\newcommand*{\LWR@depthsubparagraph}{5}`

Used by `\itemize`, `\enumerate`, `\description`:

5817 `\newcommand*{\LWR@depthlist}{6}`

Used by `\item`:

5818 `\newcommand*{\LWR@depthlistitem}{7}`
5819 `\let\LWR@depthdescitem\LWR@depthlistitem`

41.2 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.

5820 `\newcommand*{\LWR@closeone}{}% top of the stack`
5821 `\newcommand*{\LWR@closetwo}{}`
5822 `\newcommand*{\LWR@closethree}{}`
5823 `\newcommand*{\LWR@closefour}{}`

```
5824 \newcommand*{\LWR@closefive}{}
5825 \newcommand*{\LWR@closesix}{}
5826 \newcommand*{\LWR@closeseven}{}
5827 \newcommand*{\LWR@closeeight}{}
5828 \newcommand*{\LWR@closenine}{}
5829 \newcommand*{\LWR@closeten}{}
5830 \newcommand*{\LWR@closeeleven}{}
5831 \newcommand*{\LWR@closetwelve}{}
5832 \newcommand*{\LWR@closethirteen}{}
5833 \newcommand*{\LWR@closefourteen}{}
5834 \newcommand*{\LWR@closefifteen}{}
5835 \newcommand*{\LWR@closesixteen}{}
5836 \newcommand*{\LWR@closeseventeen}{}
5837 \newcommand*{\LWR@closeeighteen}{}
5838 \newcommand*{\LWR@closenineteen}{}
```

41.3   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}
```

```
5839 \newcommand*{\LWR@closedepthone}{\LWR@depthnone}% top of the stack
5840 \newcommand*{\LWR@closedepthtwo}{\LWR@depthnone}
5841 \newcommand*{\LWR@closedepththree}{\LWR@depthnone}
5842 \newcommand*{\LWR@closedepthfour}{\LWR@depthnone}
5843 \newcommand*{\LWR@closedepthfive}{\LWR@depthnone}
5844 \newcommand*{\LWR@closedepthsix}{\LWR@depthnone}
5845 \newcommand*{\LWR@closedepthseven}{\LWR@depthnone}
5846 \newcommand*{\LWR@closedeptheight}{\LWR@depthnone}
5847 \newcommand*{\LWR@closedepthnine}{\LWR@depthnone}
5848 \newcommand*{\LWR@closedepthten}{\LWR@depthnone}
5849 \newcommand*{\LWR@closedeptheleven}{\LWR@depthnone}
5850 \newcommand*{\LWR@closedepthtwelve}{\LWR@depthnone}
5851 \newcommand*{\LWR@closedepththirteen}{\LWR@depthnone}
5852 \newcommand*{\LWR@closedepthfourteen}{\LWR@depthnone}
5853 \newcommand*{\LWR@closedepthfifteen}{\LWR@depthnone}
5854 \newcommand*{\LWR@closedepthsixteen}{\LWR@depthnone}
5855 \newcommand*{\LWR@closedepthseventeen}{\LWR@depthnone}
5856 \newcommand*{\LWR@closedeptheighteen}{\LWR@depthnone}
5857 \newcommand*{\LWR@closedepthnineteen}{\LWR@depthnone}
```

41.4   Pushing and popping the stack

\LWR@pushclose {⟨*sectiontype*⟩}

Pushes one return action and its LATEX depth onto the stacks.

```
5858 \NewDocumentCommand{\LWR@pushclose}{m}
5859 {%
5860 \global\let\LWR@closenineteen\LWR@closeeighteen%
5861 \global\let\LWR@closeeighteen\LWR@closeseventeen%
5862 \global\let\LWR@closeseventeen\LWR@closesixteen%
5863 \global\let\LWR@closesixteen\LWR@closefifteen%
5864 \global\let\LWR@closefifteen\LWR@closefourteen%
5865 \global\let\LWR@closefourteen\LWR@closethirteen%
5866 \global\let\LWR@closethirteen\LWR@closetwelve%
5867 \global\let\LWR@closetwelve\LWR@closeeleven%
5868 \global\let\LWR@closeeleven\LWR@closeten%
5869 \global\let\LWR@closeten\LWR@closenine%
5870 \global\let\LWR@closenine\LWR@closeeight%
5871 \global\let\LWR@closeeight\LWR@closeseven%
5872 \global\let\LWR@closeseven\LWR@closesix%
5873 \global\let\LWR@closesix\LWR@closefive%
5874 \global\let\LWR@closefive\LWR@closefour%
5875 \global\let\LWR@closefour\LWR@closethree%
5876 \global\let\LWR@closethree\LWR@closetwo%
5877 \global\let\LWR@closetwo\LWR@closeone%
5878 \global\csletcs{LWR@closeone}{LWR@printclose#1}%
5879 \global\let\LWR@closedepthnineteen\LWR@closedeptheighteen%
5880 \global\let\LWR@closedeptheighteen\LWR@closedepthseventeen%
5881 \global\let\LWR@closedepthseventeen\LWR@closedepthsixteen%
5882 \global\let\LWR@closedepthsixteen\LWR@closedepthfifteen%
5883 \global\let\LWR@closedepthfifteen\LWR@closedepthfourteen%
5884 \global\let\LWR@closedepthfourteen\LWR@closedepththirteen%
5885 \global\let\LWR@closedepththirteen\LWR@closedepthtwelve%
5886 \global\let\LWR@closedepthtwelve\LWR@closedeptheleven%
5887 \global\let\LWR@closedeptheleven\LWR@closedepthten%
5888 \global\let\LWR@closedepthten\LWR@closedepthnine%
5889 \global\let\LWR@closedepthnine\LWR@closedeptheight%
5890 \global\let\LWR@closedeptheight\LWR@closedepthseven%
5891 \global\let\LWR@closedepthseven\LWR@closedepthsix%
5892 \global\let\LWR@closedepthsix\LWR@closedepthfive%
5893 \global\let\LWR@closedepthfive\LWR@closedepthfour%
5894 \global\let\LWR@closedepthfour\LWR@closedepththree%
5895 \global\let\LWR@closedepththree\LWR@closedepthtwo%
5896 \global\let\LWR@closedepthtwo\LWR@closedepthone%
5897 \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.

```
5898 \ifdefstring{\LWR@closedepthnineteen}{\LWR@depthnone}%
5899     {}%
5900     {%
5901         \PackageError{lwarp}%
5902             {The document is nested too deeply for Lwarp}%
5903             {PLEASE inform the Lwarp maintainer!}%
5904     }%
5905 }
```

\LWR@popclose Pops one action and its depth off the stacks.

```
5906 \newcommand*{\LWR@popclose}
5907 {%
```

```
5908 \global\let\LWR@closeone\LWR@closetwo%
5909 \global\let\LWR@closetwo\LWR@closethree%
5910 \global\let\LWR@closethree\LWR@closefour%
5911 \global\let\LWR@closefour\LWR@closefive%
5912 \global\let\LWR@closefive\LWR@closesix%
5913 \global\let\LWR@closesix\LWR@closeseven%
5914 \global\let\LWR@closeseven\LWR@closeeight%
5915 \global\let\LWR@closeeight\LWR@closenine%
5916 \global\let\LWR@closenine\LWR@closeten%
5917 \global\let\LWR@closeten\LWR@closeeleven%
5918 \global\let\LWR@closeeleven\LWR@closetwelve%
5919 \global\let\LWR@closetwelve\LWR@closethirteen%
5920 \global\let\LWR@closethirteen\LWR@closefourteen%
5921 \global\let\LWR@closefourteen\LWR@closefifteen%
5922 \global\let\LWR@closefifteen\LWR@closesixteen%
5923 \global\let\LWR@closesixteen\LWR@closeseventeen%
5924 \global\let\LWR@closeseventeen\LWR@closeeighteen%
5925 \global\let\LWR@closeeighteen\LWR@closenineteen%
5926 \global\let\LWR@closedepthone\LWR@closedepthtwo%
5927 \global\let\LWR@closedepthtwo\LWR@closedepththree%
5928 \global\let\LWR@closedepththree\LWR@closedepthfour%
5929 \global\let\LWR@closedepthfour\LWR@closedepthfive%
5930 \global\let\LWR@closedepthfive\LWR@closedepthsix%
5931 \global\let\LWR@closedepthsix\LWR@closedepthseven%
5932 \global\let\LWR@closedepthseven\LWR@closedeptheight%
5933 \global\let\LWR@closedeptheight\LWR@closedepthnine%
5934 \global\let\LWR@closedepthnine\LWR@closedepthten%
5935 \global\let\LWR@closedepthten\LWR@closedeptheleven%
5936 \global\let\LWR@closedeptheleven\LWR@closedepthtwelve%
5937 \global\let\LWR@closedepthtwelve\LWR@closedepththirteen%
5938 \global\let\LWR@closedepththirteen\LWR@closedepthfourteen%
5939 \global\let\LWR@closedepthfourteen\LWR@closedepthfifteen%
5940 \global\let\LWR@closedepthfifteen\LWR@closedepthsixteen%
5941 \global\let\LWR@closedepthsixteen\LWR@closedepthseventeen%
5942 \global\let\LWR@closedepthseventeen\LWR@closedeptheighteen%
5943 \global\let\LWR@closedeptheighteen\LWR@closedepthnineteen%
5944 }
```

```
5945 \end{warpHTML}
```

42   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}{}

`5946 \begin{warpHTML}`

\LWR@setexparray {⟨*name*⟩} {⟨*index*⟩} {⟨*contents*⟩}

```
5947 \newbool{LWR@setexparray@doingparhooks}
5948
5949 \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.

```
5950     \let\ifLWR@setexparray@doingparhooks\ifLWR@doingparhooks%
5951     \setbool{LWR@doingparhooks}{false}%
5952     \let\LWR@setexparray@par\par%
5953     \let\par\relax%
```

The name of the control sequence is the given name with the index appended.

```
5954     \xdef\LWR@thisexparrayname{#1#2}%
```

Locally assign the value to the control sequence:

```
5955     \ifstrempty{#3}%
5956         {\csdef{\LWR@thisexparrayname}{}}%
5957         {\csedef{\LWR@thisexparrayname}{#3}}%
```

Restore the paragraph handling:

```
5958     \let\ifLWR@doingparhooks\ifLWR@setexparray@doingparhooks%
5959     \let\par\LWR@setexparray@par%
5960 }
```

\LWR@getexparray {⟨*name*⟩} {⟨*index*⟩}

```
5961 \newcommand*{\LWR@getexparray}[2]{%
5962     \@nameuse{#1#2}%
5963 }
```

```
5964 \end{warpHTML}
```

43   Localizing catcodes

`5965 \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.

```
5966 \newcommand{\StartDefiningTabulars}{%
5967     \LWR@traceinfo{StartDefiningTabulars}%
5968     \warpHTMLonly{\catcode`\&=\active}%
5969 }
```

\StopDefiningTabulars Place after defining something with & in it.

```
5970 \newcommand{\StopDefiningTabulars}{%
5971     \LWR@traceinfo{StopDefiningTabulars}%
5972     \warpHTMLonly{\catcode`\&=4}%
5973 }
```

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.

```
5974 \newbool{LWR@mathmacro}
5975 \boolfalse{LWR@mathmacro}
```

\StartDefiningMath Place before defining something with $ in it.

```
5976 \newcommand{\StartDefiningMath}{%
5977     \LWR@traceinfo{StartDefiningMath}%
5978     \warpHTMLonly{\catcode`\$=\active}%
5979 }
```

\StopDefiningMath Place after defining something with $ in it.

```
5980 \newcommand{\StopDefiningMath}{%
5981     \LWR@traceinfo{StopDefiningMath}%
5982     \warpHTMLonly{\catcode`\$=3}% math shift
5983 }
```

```
5984 \end{warpall}
```

5985 \begin{warpHTML}

A definition for & in case it is referred to after \StartDefiningTabulars but outside
a tabular.

```
5986 \StartDefiningTabulars
5987 \protected\gdef&{%
5988     \PackageWarning{lwarp}{%
5989         An ampersand is being used inside a tabular\MessageBreak
5990     }%
5991 }%
5992 \StopDefiningTabulars
```

```
5993 \end{warpHTML}
```

44   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.}}
```

5994 \begin{warpall}

LWR@dynamicmath (*bool*)   True to mark inline math which is dynamic in nature, thus should not be hashed
Default: false   for reuse.

5995 \newbool{LWR@dynamicmath}
5996 \boolfalse{LWR@dynamicmath}

\inlinemathother Place before using $ ... $ or \(... \) if the contents of the math are not static,
depending on counters or dynamic macros.

5997 \newcommand{\inlinemathother}{%
5998 \LWR@traceinfo{inlinemathother}%
5999 \booltrue{LWR@dynamicmath}%
6000 }

\inlinemathnormal Place after using $ ... $ or \(... \) with dynamic contents.

6001 \newcommand{\inlinemathnormal}{%
6002 \LWR@traceinfo{inlinemathnormal}%
6003 \boolfalse{LWR@dynamicmath}%
6004 }

6005 \end{warpall}

45   HTML entities

6006 \begin{warpHTML}

HTML Unicode entities:

6007 \let\LWR@origampersand\&

\LWR@fontfortags {⟨*macro name*⟩} {⟨*argument*⟩}

Forces roman TT font for HTML tags.

6008 \newrobustcmd*{\LWR@fontfortags}[2]{%

```
6009        \ifmmode%
6010            \PackageError{lwarp}%
6011                {%
6012                    An HTML tag was generated inside math.\MessageBreak
6013                    This should never occur.\MessageBreak
6014                    Something is broken in Lwarp.\MessageBreak
6015                    Enter 'h' for details%
6016                }%
6017                {(Using #1{#2}.)}%
6018        \else%
```

Used by ltjtbook, platex, and related.

```
6019            \ifdef{\romanencoding}%
6020                {%
6021                    \romanencoding{\encodingdefault}%
6022                }%
6023                {%
```

Used by babel:

```
6024                    \ifdef{\latintext}
6025                        {\latintext}
6026                        {\fontencoding\encodingdefault}%
6027                }%
6028            \LWR@print@normalfont%
6029            \LWR@origttfamily%
6030        \fi%
6031 }
```

\HTMLentity {⟨*entitytag*⟩}

> \protect is in case the tag appears in TOC, LOF, LOT.

```
6032 \newcommand*{\HTMLentity}[1]{%
6033 % \LWR@traceinfo{HTMLentity \detokenize{#1}}%
6034     \begingroup%
6035     \LWR@hook@processingtags%
6036     \LWR@fontfortags{HTMLentity}{\detokenize{#1}}%
6037     \protect\LWR@origampersand\LWR@isolate{#1};%
6038     \endgroup%
6039 % \LWR@traceinfo{HTMLentity done}%
6040 }
```

\HTMLunicode {⟨*hex_unicode*⟩}

```
6041 \newcommand*{\HTMLunicode}[1]{\HTMLentity{\LWR@origpound{}x#1}}
```

\&

```
6042 \renewrobustcmd*{\&}{\HTMLentity{amp}}
```

\textless

```
6043 \let\LWR@origtextless\textless
6044 \renewrobustcmd*{\textless}{\HTMLentity{lt}}
```

\textgreater

```
6045 \let\LWR@origtextgreater\textgreater
6046 \renewrobustcmd*{\textgreater}{\HTMLentity{gt}}
```

```
6047 \end{warpHTML}
```

46 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: 6048 \begin{warpall}

\BaseJobname   The \jobname of the printed version, even if currently compiling the HTML ver-
sion. I.e. this is the \jobname without _html appended. This is used to set
\HomeHTMLFilename if the user did not provide one.

```
6049 \providecommand*{\BaseJobname}{\jobname}
```

\HTMLFilename   The prefix for all generated HTML files other than the home page, defaulting to
empty. See section 7.6.1.

```
6050 \providecommand*{\HTMLFilename}{}
```

\HomeHTMLFilename   The filename of the home page, defaulting to the \BaseJobname. See section 7.6.1.

```
6051 \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.

```
6052 \newcommand*{\SetHTMLFileNumber}[1]{%
6053     \setcounter{LWR@htmlfilenumber}{#1}%
6054 }
```

FileSectionNames (*bool*)   Selects how to create HTML file names.

Defaults to use section names in the filenames.

```
6055 \newbool{FileSectionNames}
6056 \booltrue{FileSectionNames}
```

```
6057 \end{warpall}
```

for HTML output: 6058 \begin{warpHTML}

Updated each time a new HTML file is begun. Used to provide HTML previous/next web page links.

```
6059 \newcounter{LWR@HTMLpagenum}
6060 \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.

```
6061 \newcounter{LWR@htmlseqfilenumber}
6062 \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.

```
6063 \newbool{LWR@setseqfilelabel}
6064 \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.

```
6065 \newcounter{LWR@htmlfilenumber}
6066 \setcounter{LWR@htmlfilenumber}{0}
```

\LWR@htmlsectionfilename {⟨*htmlfilenumber or name*⟩}

Prints the filename for a given section: \HTMLFilename{}filenumber/name.html

```
6067 \newcommand*{\LWR@htmlsectionfilename}[1]{%
6068 \LWR@traceinfo{LWR@htmlsectionfilename A !\detokenize{#1}!}%
6069 \begingroup%
```

Disable CJK xpinyin while generating file names.

```
6070 \LWR@disablepinyin%
```

Section 0 or empty is given the home filename. The filename must be detokenized for underscores.

```
6071 % \LWR@traceinfo{about to assign temp}%
6072 \LWR@sanitize{#1}%
6073 \LWR@traceinfo{about to compare with ??}%
6074 \ifdefstring{\LWR@sanitized}{??}
6075     {\LWR@traceinfo{found ??}}%
6076     {\LWR@traceinfo{not found ??}}%
6077 \LWR@traceinfo{about to compare with zero or empty}%
6078 \ifboolexpr{
6079     test {\ifdefstring{\LWR@sanitized}{0}} or
6080     test {\ifdefstring{\LWR@sanitized}{}} or
6081     test {\ifdefstring{\LWR@sanitized}{??}}
6082 }
6083 {%
```

```
6084        \LWR@traceinfo{LWR@htmlsectionfilename B \HomeHTMLFilename.html}%
6085        \HomeHTMLFilename.html%
6086 }%
```

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:

```
6087 {%
6088        \LWR@traceinfo{LWR@htmlsectionfilename C \LWR@sanitized}%
6089        \ifboolexpr{
6090              test{\ifdefvoid{\HTMLFilename}} and
6091              (
6092                    test{\ifdefstring{\LWR@sanitized}{Index}} or
6093                    test{\ifdefstring{\LWR@sanitized}{index}}
6094              )
6095        }%
6096        {%
6097              \LWR@traceinfo{Adding a zero to the index filename.}%
6098              \LWR@sanitized-0.html%
6099        }%
```

Otherwise, create a filename with the chosen prefix:

```
6100        {%
6101              \HTMLFilename\LWR@isolate{\LWR@sanitized}.html%
6102        }%
6103 }%
6104 \LWR@traceinfo{LWR@htmlsectionfilename Z}%
6105 \endgroup%
6106 }
```

\LWR@htmlrefsectionfilename {⟨*label*⟩}

Prints the filename for the given label

```
6107 \newcommand*{\LWR@htmlrefsectionfilename}[1]{%
6108        \LWR@traceinfo{LWR@htmlrefsectionfilename: !\detokenize{#1}!}%
```

\LWR@nullfonts to allow math in a section name.

```
6109        \begingroup%
6110        \LWR@nullfonts%
6111        \LWR@htmlsectionfilename{\LWR@htmlfileref{#1}}%
6112        \endgroup%
6113        \LWR@traceinfo{LWR@htmlrefsectionfilename: done}%
6114 }
```

```
6115 \end{warpHTML}
```

47   Homepage link

6116 \begin{warpall}

\linkhomename Holds the default name for the home link.

```
6117 \newcommand{\linkhomename}{Home}
```

6118 `\end{warpall}`

6119 `\begin{warpHTML}`

\LinkHome May be used wherever you wish to place a link back to the homepage. The filename must be detokenized for underscores.

6120 `\newcommand*{\LinkHome}{%`
6121 `    \LWR@subhyperrefclass{\HomeHTMLFilename.html}{\linkhomename}{linkhome}%`
6122 `}`

6123 `\end{warpHTML}`

6124 `\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.

6125 `\AtBeginDocument{`
6126 `\@ifundefined{hyperref}{`
6127 `    \newcommand*{\LinkHome}{%`
6128 `        \linkhomename\ --- page \pageref{\BaseJobname-page-LWRfirstpage}%`
6129 `    }`
6130 `}{`
6131 `    \newcommand*{\LinkHome}{%`
6132 `        \hyperref[\BaseJobname-page-LWRfirstpage]{\linkhomename}%`
6133 `    }`
6134 `}`
6135 `}`
6136
6137 `\AfterEndPreamble{\label{\BaseJobname-page-LWRfirstpage}}`

6138 `\end{warpprint}`

6139 `\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 sideTOC.

6140 `\newcommand*{\LWR@topnavigation}{%`
6141 `    \LWR@htmlelementclassline{nav}{topnavigation}{\LinkHome}`
6142 `}`

\LWR@botnavigation Creates a link to the homepage at the bottom of the page for use when the window is too narrow for the sideTOC.

6143 `\newcommand*{\LWR@botnavigation}{%`
6144 `    \LWR@htmlelementclassline{nav}{botnavigation}{\LinkHome}`
6145 `}`

6146 `\end{warpHTML}`

48   Previous/next navigation links

6147 `\begin{warpall}`

`\linkpreviousname`   What to call the link to the previous web page.

6148 `\newcommand*{\linkpreviousname}{Previous}`

`\linknextname`   What to call the link to the next web page.

6149 `\newcommand*{\linknextname}{Next}`

6150 `\end{warpall}`

6151 `\begin{warpprint}`

`\LinkPrevious`   Creates a link to the previous web page if there is one.

6152 `\newcommand*{\LinkPrevious}{}`

`\LinkNext`   Creates a link to the next web page if there is one.

6153 `\newcommand*{\LinkNext}{}`

6154 `\end{warpprint}`

6155 `\begin{warpHTML}`

`\LinkPrevious`   Creates a link to the previous web page if there is one.

The links refer to the LaTeX labels `\Basejobname-autofile-*`

```
6156 \newcommand*{\LinkPrevious}{%
6157     \ifnumless{\value{LWR@htmlseqfilenumber}}{1}{}{%
6158         \setcounter{LWR@tempcountone}{\value{LWR@htmlseqfilenumber}-1}%
6159         \LWR@subhyperrefclass{%
6160             \LWR@htmlrefsectionfilename{%
6161                 \BaseJobname-autofile-\arabic{LWR@tempcountone}%
6162             }%
6163         }{\linkpreviousname}{linkhome}%
6164     }%
6165 }
```

`\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`

```
6166 \newcommand*{\LinkNext}{%
6167     \ifcsdef{r@\BaseJobname-autofile-last@lwarp}{%
6168         \edef\LWR@tempone{%
6169         \LWR@htmlfileref{\BaseJobname-autofile-\arabic{LWR@htmlseqfilenumber}}}%
6170         }%
```

```
6171        \edef\LWR@temptwo{%
6172            \LWR@htmlfileref{\BaseJobname-autofile-last}%
6173        }%
6174        \ifdefequal{\LWR@tempone}{\LWR@temptwo}{}{%
6175            \setcounter{LWR@tempcountone}{\value{LWR@htmlseqfilenumber}+1}%
6176            \LWR@subhyperrefclass{%
6177                \LWR@htmlrefsectionfilename{%
6178                    \BaseJobname-autofile-\arabic{LWR@tempcountone}%
6179                }%
6180            }{\linknextname}{linkhome}%
6181        }%
6182    }{}%
6183 }
```

```
6184 \end{warpHTML}
```

49 \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: 6185 \begin{warpHTML}

\LWRPrintStack Prints the closedepth stack.

```
6186 \newcommand*{\LWR@subprintstack}{
6187 \LWR@closedepthone\ \LWR@closedepthtwo\ \LWR@closedepththree\
6188 \LWR@closedepthfour\ \LWR@closedepthfive\ \LWR@closedepthsix\
6189 \LWR@closedepthseven\ \LWR@closedeptheight\ \LWR@closedepthnine\
6190 \LWR@closedepthten\ \LWR@closedeptheleven\ \LWR@closedepthtwelve\
6191 \LWR@closedeptththirteen\ \LWR@closedepthfourteen\ \LWR@closedepthfifteen\
6192 \LWR@closedepthsixteen\ \LWR@closedepthseventeen\ \LWR@closedeptheighteen\
6193 \LWR@closedepthnineteen\
6194 }
6195
6196 \newcommand*{\LWRPrintStack}{
6197 \LWR@startpars
6198 \LWR@subprintstack
6199 }
```

```
6200 \end{warpHTML}
```

for PRINT output: 6201 \begin{warpprint}

```
6202 \newcommand*{\LWRPrintStack}{}
```

```
6203 \end{warpprint}
```

50 Closing stack levels

for HTML output: 6204 \begin{warpHTML}

Close one nested level:

```
6205 \newcommand*{\LWR@closeoneprevious}{%
6206
6207 \LWR@closeone
6208
6209 \LWR@popclose
6210 }
```

\LWR@closeprevious {⟨*sectintype*⟩} Close everything up to the given depth:

```
6211 \newcommand*{\LWR@closeprevious}[1]{%
6212 \LWR@traceinfo{%
6213     LWR@closeprevious to depth \csuse{LWR@depth#1}, %
6214     depths are \LWR@subprintstack%
6215 }%
```

Close any pending paragraph:

```
6216 \LWR@stoppars%
```

Close anything nested deeper than the desired depth. First close anything deeper, then at most one of the same level.

```
6217 \whileboolexpr{test{\ifnumcomp{\LWR@closedepthone}{>}{\csuse{LWR@depth#1}}}}%
6218 {%
6219     \LWR@traceinfo{LWR@closeprevious: closing out depth \LWR@closedepthone}%
6220     \LWR@closeoneprevious%
6221 }%
6222 \ifboolexpr{test{\ifnumcomp{\LWR@closedepthone}{=}{\csuse{LWR@depth#1}}}}%
6223 {%
6224     \LWR@traceinfo{LWR@closeprevious: closing out depth \LWR@closedepthone}%
6225     \LWR@closeoneprevious%
6226 }{}%
6227 \LWR@traceinfo{LWR@closeprevious: done, depths are \LWR@subprintstack}%
6228 }
```

```
6229 \end{warpHTML}
```

51   PDF pages and styles

```6230 \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.

```
6231 \newcommand{\LWR@forcenewpage}{%
6232 \LWR@traceinfo{LWR@forcenewpage}%
6233 \ifinner\else%
6234     \LWR@traceinfo{LWR@forcenewpage A}%
6235     \LWR@stoppars%
6236     \LWR@traceinfo{LWR@forcenewpage B}%
6237     \LWR@maybe@orignewpage%
```

```
6238    \LWR@traceinfo{LWR@forcenewpage C}%
6239    \LWR@startpars%
6240 \fi%
6241 \LWR@traceinfo{LWR@forcenewpage done}%
6242 }
```

\pagestyle, etc. are nullified for HTML output.

\pagestyle {⟨*style*⟩}

```
6243 \renewcommand*{\pagestyle}[1]{}
```

\thispagestyle {⟨*style*⟩}

```
6244 \renewcommand*{\thispagestyle}[1]{}
```

\markboth {⟨*left*⟩} {⟨*right*⟩}

```
6245 \renewcommand*{\markboth}[2]{}
```

\markright {⟨*right*⟩}

```
6246 \renewcommand*{\markright}[1]{}
```

\raggedbottom

```
6247 \renewcommand*{\raggedbottom}{}
```

\flushbottom

```
6248 \renewcommand*{\flushbottom}{}
```

\sloppy

```
6249 \renewcommand*{\sloppy}{}
```

\fussy

```
6250 \renewcommand*{\fussy}{}
```

\pagenumbering * {⟨*commands*⟩}

```
6251 \RenewDocumentCommand{\pagenumbering}{s m}{}
```

```
6252 \end{warpHTML}
```

52   HTML tags, spans, divs, elements

for HTML output: 6253 \begin{warpHTML}

52.1   Mapping LaTeX sections to HTML sections

```
6254 \newcommand*{\LWR@tagtitle}{h1}
6255 \newcommand*{\LWR@tagtitleend}{/h1}
6256 \newcommand*{\LWR@tagbook}{div class=\textquotedbl{}book\textquotedbl}
6257 \newcommand*{\LWR@tagbookend}{/div}
6258 \newcommand*{\LWR@tagpart}{h2}
6259 \newcommand*{\LWR@tagpartend}{/h2}
6260 \newcommand*{\LWR@tagchapter}{h3}
6261 \newcommand*{\LWR@tagchapterend}{/h3}
6262 \newcommand*{\LWR@tagsection}{h4}
6263 \newcommand*{\LWR@tagsectionend}{/h4}
6264 \newcommand*{\LWR@tagsubsection}{h5}
6265 \newcommand*{\LWR@tagsubsectionend}{/h5}
6266 \newcommand*{\LWR@tagsubsubsection}{h6}
6267 \newcommand*{\LWR@tagsubsubsectionend}{/h6}
6268 \newcommand*{\LWR@tagparagraph}{span class=\textquotedbl{}paragraph\textquotedbl}
6269 \newcommand*{\LWR@tagparagraphend}{/span}
6270 \newcommand*{\LWR@tagsubparagraph}{span class=\textquotedbl{}subparagraph\textquotedbl}
6271 \newcommand*{\LWR@tagsubparagraphend}{/span}
6272
6273 \newcommand*{\LWR@tagregularparagraph}{p}
```

52.2   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.

```
6274 \newcommand*{\LWR@hook@processingtags}{}
```

52.3   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.)

```
6275 \providecommand*{\LWR@FBcancel}{}
6276
6277 \AtBeginDocument{%
```

In some circumstances, \NoAutoSpacing may be defined when \frenchbsetup is
not.

```
6278 \@ifundefined{NoAutoSpacing}%
6279     {}%
6280     {%
6281         \LetLtxMacro\LWR@FBcancel\NoAutoSpacing%
6282         \appto{\LWR@hook@processingtags}{\LWR@FBcancel}%
6283     }%
6284
6285 \@ifundefined{frenchbsetup}%
```

```
6286 {}%
6287 {%
6288     \frenchbsetup{FrenchFootnotes=false}%

6289 %
6290     \renewrobustcmd*{\FBcolonspace}{%
6291         \begingroup%
6292         \LWR@hook@processingtags%
6293         \LWR@origampersand{}nbsp;%
6294         \endgroup%
6295     }%
6296     \renewrobustcmd*{\FBthinspace}{%
6297         \begingroup%
6298         \LWR@hook@processingtags%
6299         \LWR@origampersand\LWR@origpound{}x202f;% \,
6300         \endgroup%
6301     }%
6302     \renewrobustcmd*{\FBguillspace}{%
6303         \begingroup%
6304         \LWR@hook@processingtags%
6305         \LWR@origampersand{}nbsp;% ~,  for \og xyz \fg{}
6306         \endgroup%
6307     }%
6308     \DeclareDocumentCommand{\FBmedkern}{}{%
6309         \begingroup%
6310         \LWR@hook@processingtags%
6311         \LWR@origampersand\LWR@origpound{}x202f;% \,
6312         \endgroup%
6313     }%
6314     \DeclareDocumentCommand{\FBthickkern}{}{%
6315         \begingroup%
6316         \LWR@hook@processingtags%
6317         \LWR@origampersand{}nbsp;% ~
6318         \endgroup%
6319     }%
6320     \renewrobustcmd*{~}{\HTMLentity{nbsp}}% was overwritten by babel-french
6321     \ifFBunicode%
6322     \else%
6323         \DeclareTextSymbol{\FBtextellipsis}{LY1}{133}%
6324         \DeclareTextCommandDefault{\FBtextellipsis}{\textellipsis\xspace}%
6325     \fi%
6326 }%
6327 }
```

52.4   HTML output formatting

Helps format the output HTML code for human readability.

\LWR@indentHTML Newline and indent the output HTML code.

```
6328 \newcommand*{\LWR@indentHTML}{%
6329     \LWR@orignewline\LWR@origrule{2em}{0pt}%
6330 }
```

\LWR@indentHTMLtwo Newline and indent the output HTML code.

```
6331 \newcommand*{\LWR@indentHTMLtwo}{%
6332     \LWR@orignewline\LWR@origrule{4em}{0pt}%
6333 }
```

52.5 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.

```
6334 \newcommand*{\LWR@htmltagc}[1]{%
6335     \LWR@traceinfo{LWR@htmltagc !\detokenize{#1}!}%
6336     \begingroup%
6337     \LWR@hook@processingtags%
6338     \LWR@fontfortags{LWR@htmltagc}{\detokenize{#1}}%
6339     \protect\LWR@origtextless%
6340     \LWR@isolate{#1}%
6341     \protect\LWR@origtextgreater%
6342     \endgroup%
6343 }
```

\LWR@spanwarnformat {⟨*object*⟩}

Warns if the given object is used inside a span.

```
6344 \newcommand*{\LWR@spanwarnformat}[1]{%
6345     \ifnumcomp{\value{LWR@spandepth}}{>}{0}{%
6346         \PackageWarning{lwarp}{%
6347             A #1 is being used inside a span.\MessageBreak
6348             Formatting may be lost,%
6349         }%
6350     }{}%
6351 }
```

\LWR@spanwarninvalid {⟨*object*⟩}

Warns if the given object is used inside a span.

```
6352 \newcommand*{\LWR@spanwarninvalid}[1]{%
6353     \ifnumcomp{\value{LWR@spandepth}}{>}{0}{%
6354         \PackageWarning{lwarp}{%
6355             A #1 is being used inside a span.\MessageBreak
6356             This generates invalid HTML,%
6357         }%
6358     }{}%
6359 }
```

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.

```
6360 \newcommand*{\LWR@nestspanitem}{%
6361     \if@newlist\else{
6362         \LWR@htmltagc{br /}%
6363         \LWR@orignewline%
6364     }\fi%
6365     \LWR@origitem%
6366 }
6367
6368 \newenvironment*{LWR@nestspan}
6369 {%
6370     \LWR@traceinfo{LWR@nestspan starting}%
6371     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
6372     {%
6373         \LWR@traceinfo{LWR@nestspan: inside a lateximage}%
6374     }%
6375     {% not in a lateximage
6376         \LWR@traceinfo{LWR@nestspan: NOT inside a lateximage}%
6377         \addtocounter{LWR@spandepth}{1}%
```

Nullify several objects inside the span:

```
6378         \RenewDocumentEnvironment{minipage}{O{t} o O{t} m}%
6379             {\LWR@spanwarnformat{minipage or \protect\parbox}}%
6380             {}%
6381         \RenewDocumentEnvironment{BlockClass}{o m}%
6382             {\LWR@spanwarnformat{multi-paragraph object}}%
6383             {}%
6384         \RenewDocumentEnvironment{LWR@BlockClassWP}{m m D(){} m}%
6385             {\LWR@spanwarnformat{multi-paragraph object}}%
6386             {}%
6387         \renewcommand{\BlockClassSingle}[2]{%
6388             {\LWR@spanwarnformat{multi-paragraph object}}%
6389             ##2%
6390         }%
6391         \renewcommand{\LWR@forcenewpage}{}%
6392         \renewcommand{\LWR@liststart}{\LetLtxMacro\item\LWR@nestspanitem}%
6393         \renewcommand{\LWR@listend}{\leavevmode}%
6394         \renewenvironment{quote}{\LWR@htmltagc{br /}}{\LWR@htmltagc{br /}}%
6395       \renewenvironment{quotation}{\LWR@htmltagc{br /}}{\LWR@htmltagc{br /}}%
6396     }% not in a lateximage
6397     \LWR@traceinfo{LWR@nestspan starting: done}%
6398 }% starting env
6399 {% ending env
6400     \LWR@traceinfo{LWR@nestspan ending}%
6401     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
6402     {}%
6403     {\addtocounter{LWR@spandepth}{-1}}%
6404     \LWR@traceinfo{LWR@nestspan ending: done}%
6405 }
```

\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.

```
6406 \NewDocumentCommand{\LWR@htmlspan}{m +m}{%
6407     \LWR@ensuredoingapar%
6408     \LWR@htmltagc{#1}%
```

```
6409        \begin{LWR@nestspan}%
6410        #2%
6411        \LWR@htmltagc{/#1}%
6412        \end{LWR@nestspan}%
6413 }
```

\LWR@htmlspanclass [⟨*style*⟩] (⟨*aria role*⟩) {⟨*class*⟩} {⟨*text*⟩}

```
6414 \NewDocumentCommand{\LWR@htmlspanclass}{o D(){} m +m}{%
6415        \LWR@traceinfo{LWR@htmlspanclass |#1|#2|#3|}%
6416        \LWR@ensuredoingapar%
6417        \ifblank{#2}%
6418            {\LWR@subhtmlelementclass{span}[#1]{#3}}%
6419            {\LWR@subhtmlelementclass{span}[#1](#2){#3}}%
6420        \begin{LWR@nestspan}%
6421        #4%
6422        \LWR@htmltagc{/span}%
6423        \LWR@traceinfo{LWR@htmlspanclass done}%
6424        \end{LWR@nestspan}%
6425 }
```

\LWR@htmltag {⟨*tag*⟩}

> Print an HTML tag: <tag>

```
6426 \newcommand*{\LWR@htmltag}[1]{%
6427        \LWR@htmltagc{#1}%
6428 }
```

52.6   Block tags and comments

In the following, \origttfamily breaks ligatures, which may not be used for HTML codes:

\LWR@htmlopencomment
\LWR@htmlclosecomment
```
6429 \newcommand*{\LWR@htmlopencomment}{%
6430 % \LWR@traceinfo{LWR@htmlopencomment}%
6431        \begingroup%
6432        \LWR@hook@processingtags%
6433        \LWR@fontfortags{LWR@htmlopencomment}{}%
6434        \LWR@print@mbox{\LWR@origtextless{}!-\/-}%
6435        \endgroup%
6436 }
6437
6438 \newcommand*{\LWR@htmlclosecomment}{%
6439 % \LWR@traceinfo{LWR@htmlclosecomment}%
6440        \begingroup%
6441        \LWR@hook@processingtags%
6442        \LWR@fontfortags{LWr@htmlclosecomment}{}%
6443        \LWR@print@mbox{-\/-\LWR@origtextgreater}%
6444        \endgroup%
6445 }
```

\LWR@htmlcomment {⟨*comment*⟩}

```
6446 \newcommand{\LWR@htmlcomment}[1]{%
6447     \ifmmode%
6448     \else%
6449         \LWR@htmlopencomment{}%
6450         {%
6451             \LWR@print@normalfont%
6452             \LWR@origttfamily% break ligatures
6453             #1%
6454         }%
6455         \LWR@htmlclosecomment{}%
6456     \fi%
6457 }
```

\LWR@htmlblockcomment {⟨*comment*⟩}

```
6458 \newcommand{\LWR@htmlblockcomment}[1]
6459     {\LWR@stoppars\LWR@htmlcomment{#1}\LWR@startpars}
```

\LWR@htmlblocktag {⟨*tag*⟩} print a stand-alone HTML tag

```
6460 \newcommand*{\LWR@htmlblocktag}[1]{%
6461     \LWR@stoppars%
6462     \LWR@htmltag{#1}%
6463     \LWR@startpars%
6464 }
```

52.7   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.

```
6465 \NewDocumentCommand{\LWR@subhtmlelementclass}{m O{} D(){} m}{%
6466     \LWR@traceinfo{LWR@subhtmlelementclass !#1!#2!#3!#4!}%
6467     \ifblank{#2}%
6468     {% empty style
6469         \LWR@htmltag{%
6470             #1%
6471             \ifblank{#3}{}{ role=\textquotedbl#3\textquotedbl}% spaces
6472             \ifblank{#4}{}{ class=\textquotedbl#4\textquotedbl}% spaces
6473         }%
6474     }%
6475     {% non-empty style
6476         \LWR@htmltag{%
6477             #1\LWR@indentHTML%
6478          \ifblank{#3}{}{role=\textquotedbl#3\textquotedbl\LWR@indentHTML}%
6479          \ifblank{#4}{}{class=\textquotedbl#4\textquotedbl\LWR@indentHTML}%
6480             style=\textquotedbl#2\textquotedbl\LWR@orignewline%
6481         }%
6482     }%
6483     \LWR@traceinfo{LWR@subhtmlelementclass done}%
6484 }
```

\LWR@htmlelementclass {⟨*element*⟩} [⟨*style*⟩] {⟨*class*⟩}

```
6485 \NewDocumentCommand{\LWR@htmlelementclass}{m o D(){} m}{%
6486     \LWR@stoppars%
6487     \LWR@forceemptyline%
6488     \ifblank{#3}%
6489         {\LWR@subhtmlelementclass{#1}[#2]{#4}}%
6490         {\LWR@subhtmlelementclass{#1}[#2](#3){#4}}%
6491     \LWR@startpars%
6492 }
```

\LWR@htmlelementclassend {⟨*element*⟩} {⟨*class*⟩}

```
6493 \newcommand*{\LWR@htmlelementclassend}[2]{%
6494     \LWR@stoppars%
6495     \LWR@htmltag{/#1}%
6496     \ifbool{HTMLDebugComments}{%
6497         \LWR@htmlcomment{End of #1 ''#2''}%
6498     }{}%
6499     \LWR@startpars%
6500 }
```

\LWR@htmldivclass [⟨*style*⟩] (⟨*aria role*⟩) {⟨*class*⟩}

```
6501 \NewDocumentCommand{\LWR@htmldivclass}{o D(){} m}{%
6502     \ifblank{#2}
6503         {\LWR@htmlelementclass{div}[#1]{#3}}%
6504         {\LWR@htmlelementclass{div}[#1](#2){#3}}%
6505 }
```

\LWR@htmldivclassend {⟨*class*⟩}

```
6506 \newcommand*{\LWR@htmldivclassend}[1]{%
6507     \LWR@htmlelementclassend{div}{#1}%
6508 }
```

52.8   Single-line elements

A single-line element, without a paragraph tag for the line of text:

\LWR@htmlelementclassline {⟨*element*⟩} [⟨*style*⟩] {⟨*class*⟩} {⟨*text*⟩}

```
6509 \NewDocumentCommand{\LWR@htmlelementclassline}{m o m +m}{%
6510     \LWR@stoppars
6511     \LWR@forceemptyline%
6512     \LWR@subhtmlelementclass{#1}[#2]{#3}%
6513     #4%
6514     \LWR@htmltag{/#1}
6515     \LWR@startpars
6516 }
```

52.9 HTML5 semantic elements

\LWR@htmlelement {⟨element⟩}

```
6517 \newcommand*{\LWR@htmlelement}[1]{%
6518     \LWR@htmlblocktag{#1}
6519 }
```

\LWR@htmlelementend {⟨element⟩}

```
6520 \newcommand*{\LWR@htmlelementend}[1]{%
6521     \LWR@stoppars
6522     \LWR@htmltag{/#1}
6523     \LWR@startpars
6524 }
6525
6526 \end{warpHTML}
```

52.10 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 95.

BlockClass (env.) [⟨style⟩] (⟨aria role⟩) {⟨class⟩} High-level interface for <div> classes.

Ex: \begin{BlockClass}{class} text \end{BlockClass}

for HTML & PRINT:
```
6527 \begin{warpall}
6528 \NewDocumentEnvironment{BlockClass}{o D(){} m}{}{}
6529 \end{warpall}
```

for HTML output:
```
6530 \begin{warpHTML}
6531
6532 \NewDocumentEnvironment{LWR@HTML@BlockClass}{o D(){} m}%
6533     {\LWR@htmldivclass[#1](#2){#3}}%
6534     {\LWR@htmldivclassend{#3}}
6535
6536 \LWR@formattedenv{BlockClass}
6537 \end{warpHTML}
```

\BlockClassSingle {⟨class⟩} {⟨text⟩} A single-line <div>, without a paragraph tag for the line of text.

for HTML & PRINT:
```
6538 \begin{warpall}
6539 \newcommand{\BlockClassSingle}[2]{#2}
6540 \end{warpall}
```

for HTML output:
```
6541 \begin{warpHTML}
6542 \newcommand{\LWR@HTML@BlockClassSingle}[2]{%
6543     \LWR@htmlelementclassline{div}{#1}{#2}%
6544 }
```

6545
6546 \LWR@formatted{BlockClassSingle}
6547 \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: 6548 \begin{warpall}
6549 \NewDocumentCommand{\InlineClass}{D(}{)}{} o m +m}{#4}%
6550 \end{warpall}

for HTML output: 6551 \begin{warpHTML}
6552 \NewDocumentCommand{\LWR@HTML@InlineClass}{D(}{)}{} o m +m}{%
6553   \LWR@traceinfo{LWR@HTML@InlineClass #3}%
6554   \ifbool{FormatWP}{%
6555   \LWR@traceinfo{LWR@HTML@InlineClass: FormatWP}%
6556   \LWR@htmlspanclass[#1]{#3}{#4}%
6557   }{%
6558   \LWR@traceinfo{LWR@HTML@InlineClass: not FormatWP}%
6559   \LWR@htmlspanclass[#2]{#3}{#4}%
6560   }%
6561   \LWR@traceinfo{LWR@HTML@InlineClass: done}%
6562 }
6563
6564 \LWR@formatted{InlineClass}
6565 \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: 6566 \begin{warpall}
6567 \NewDocumentEnvironment{LWR@BlockClassWP}{m m D(}{)}{} m}{}{}
6568 \end{warpall}

for HTML output: 6569 \begin{warpHTML}
6570 \NewDocumentEnvironment{LWR@HTML@LWR@BlockClassWP}{m m D(}{)}{} m}%
6571   {%
6572   \LWR@stoppars%
6573   \ifbool{FormatWP}%
6574   {%
6575   \addtocounter{LWR@thisautoidWP}{1}%

6576   \LWR@htmltag{%
6577   div class=\textquotedbl#4\textquotedbl\ % space
6578   id=\textquotedbl%
6579   \LWR@print@mbox{autoidWP-\arabic{LWR@thisautoidWP}}%
6580   \textquotedbl%
6581   \ifblank{#3}{}{ role=\textquotedbl#3\textquotedbl}%
6582   \ifblank{#1}{}{ style=\textquotedbl#1\textquotedbl}%
6583   }%

```
6584        }% FormatWP
6585        {% not FormatWP
6586          \LWR@htmltag{%
6587              div class=\textquotedbl#4\textquotedbl%
6588              \ifblank{#3}{}{ role=\textquotedbl#3\textquotedbl}%
6589              \ifblank{#2}{}{ style=\textquotedbl#2\textquotedbl}%
6590          }%
6591        }% not FormatWP
6592        \LWR@startpars%
6593      }
6594      {\LWR@htmldivclassend{#4}}
6595
6596 \LWR@formattedenv{LWR@BlockClassWP}
6597 \end{warpHTML}
```

52.11   Closing HTML tags

6598 \begin{warpHTML}

Sections H1, H2, etc. do not need a closing HTML tag, but we add a comment for readability:

```
6599 \newcommand*{\LWR@printclosebook}
6600     {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing book}}{}}
6601 \newcommand*{\LWR@printclosepart}
6602     {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing part}}{}}
6603 \newcommand*{\LWR@printclosechapter}
6604     {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing chapter}}{}}
6605 \newcommand*{\LWR@printclosesection}
6606     {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing section}}{}}
6607 \newcommand*{\LWR@printclosesubsection}
6608     {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing subsection}}{}}
6609 \newcommand*{\LWR@printclosesubsubsection}
6610     {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing subsubsection}}{}}
6611 \newcommand*{\LWR@printcloseparagraph}
6612     {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing paragraph}}{}}
6613 \newcommand*{\LWR@printclosesubparagraph}
6614     {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing subparagraph}}{}}
```

Lists require closing HTML tags:

```
6615 \newcommand*{\LWR@printcloselistitem}
6616     {\LWR@htmltag{/li}}
6617 \newcommand*{\LWR@printclosedescitem}
6618     {\LWR@htmltag{/dd}}
6619 \newcommand*{\LWR@printcloseitemize}
6620     {\LWR@htmltag{/ul}}
6621 \newcommand*{\LWR@printcloseenumerate}
6622     {\LWR@htmltag{/ol}}
6623 \newcommand*{\LWR@printclosedescription}
6624     {\LWR@htmltag{/dl}}
```

```
6625 \end{warpHTML}
```

53 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 54 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@laateximagedepth 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: 6626 `\begin{warpHTML}`

LWR@spandepth (*Ctr*)   Do not create paragraph tags inside of an HTML span.

```
6627 \newcounter{LWR@spandepth}
6628 \setcounter{LWR@spandepth}{0}
```

LWR@doingparhooks (*bool*)   Tells whether the lwarp paragraph hooks are to be active.

```
6629 \newbool{LWR@doingparhooks}
6630 \boolfalse{LWR@doingparhooks}
```

LWR@in@multirow@par (*bool*)   Tells whether to generate break instead of paragraph tags inside a \multirow.

```
6631 \newbool{LWR@in@multirow@par}
6632 \boolfalse{LWR@in@multirow@par}
```

LWR@starting@fancybox (*bool*)   Suppresses <br> if beginning a fancybox environment.

```
6633 \newbool{LWR@starting@fancybox}
6634 \boolfalse{LWR@starting@fancybox}
```

LWR@doingstartpars (*bool*)   Tells whether paragraphs may be generated.

```
6635 \newbool{LWR@doingstartpars}
6636 \boolfalse{LWR@doingstartpars}
```

LWR@doingapar (*bool*)   Tells whether have actually generated and are currently processing paragraph text.

```
6637 \newbool{LWR@doingapar}
6638 \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.

```
6639 \newbool{LWR@algocf@dopars}
6640 \boolfalse{LWR@algocf@dopars}
```

\PN@parnotes@auto   Redefined by parnotes to print paragraph notes at the end of each paragraph.

```
6641 \def\PN@parnotes@auto{}%
```

\LWR@ensuredoingapar These were different in older versions of lwarp, but are now the same thing.
  \LWR@openparagraph

```
6642 \newcommand*{\LWR@openparagraph}
6643 {%
```

See if paragraph handling is enabled:

```
6644     \ifboolexpr{
6645         bool{LWR@doingparhooks} and
6646         bool{LWR@doingstartpars}
6647     }%
6648     {% handling pars
```

See if have already started a lateximage or a <span>. If so, do not generate nested paragraph tags.

```
6649         \ifboolexpr{
6650             test {\ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}} or
6651             test {\ifnumcomp{\value{LWR@spandepth}}{>}{0}}
6652         }% 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>.

```
6653         {}% no nested par tags
```

Else: No lateximage or <span> has been started yet, so it's OK to generate paragraph tags.

```
6654         {% yes nest par tags
6655             \ifbool{LWR@doingapar}{}{%
```

If parnotes is used, paragraph notes are inserted before starting the next paragraph:

```
6656                 \PN@parnotes@auto%
```

Set flag before creating the tag, so that the tag itself does not trigger a new paragraph:

```
6657                 \global\booltrue{LWR@doingapar}%
```

The opening paragraph tag. Do not create tag if doing algorithm2e handling instead:

```
6658                 \ifbool{LWR@algocf@dopars}{}{%
6659                     \ifbool{LWR@in@multirow@par}%
6660                         {}%
6661                     {\LWR@htmltagc{\LWR@tagregularparagraph}\LWR@orignewline}%
6662                 }%
6663             }%
6664         }% end of yes nest par tags
6665     }% end of handling pars
6666     {}% not handling pars
6667 }
6668
6669 \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.

```
6670 \newcommand*{\LWR@closeparagraph@br}
6671 {%
6672     \ifboolexpr{
6673         test {\ifnumcomp{\value{LWR@spandepth}}{>}{0}} and
6674         test {\ifnumcomp{\value{LWR@lateximagedepth}}{=}{0}} and
6675         not bool {LWR@starting@fancybox} and
6676         not bool {LWR@intabularmetadata} or
6677         bool {LWR@in@multirow@par}
6678     }%
6679         {\unskip\LWR@htmltagc{br /}}%
6680         {}%
6681 }
```

\LWR@closeparagraph

```
6682 \newcommand*{\LWR@closeparagraph}
6683 {%
6684 % \LWR@traceinfo{LWR@closeparagraph}%
```

See if paragraph handling is enabled:

```
6685     \ifbool{LWR@doingparhooks}{%
6686         \ifbool{LWR@doingapar}%
```

If currently in paragraph mode:

```
6687         {% handling pars
```

See if already started a lateximage or a <span>:

```
6688         \ifboolexpr{
6689             test {\ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}} or
6690             test {\ifnumcomp{\value{LWR@spandepth}}{>}{0}} or
6691             bool{LWR@in@multirow@par}
6692         }%
```

Add a parbreak if in a span, not in a lateximage, and not in table metadata.

```
6693         {% no nested par tags
6694             \LWR@closeparagraph@br%
6695         }% no nested par tags
```

If have not already started a lateximage or a <span>:

```
6696         {% 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.)

```
6697             \@hspacer{\fill}% \hspace*{\fill}
6698             \leavevmode\LWR@orignewline%
6699             \LWR@htmltagc{/\LWR@tagregularparagraph}%
```

No longer doing a paragraph:

```
6700                    \global\boolfalse{LWR@doingapar}%
```

Disable the special minipage & \hspace interaction until a new minipage is found:

```
6701                    \global\boolfalse{LWR@minipagethispar}%
```

If parnotes is used, paragraph notes are inserted after ending the previous paragraph:

```
6702                    \PN@parnotes@auto%
```

```
6703                }% end of yes nest par tags
6704            }% LWR@doingapar: end of handling pars
```

Add a parbreak if in a span, not in a lateximage, and not in table metadata.

```
6705            {% not LWR@doingapar: not handling pars
6706                \LWR@closeparagraph@br%
6707            }% 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.

```
6708            \ifboolexpr{%
6709                not bool {LWR@doingapar} and
6710                test {\ifnumcomp{\value{LWR@tabulardepth}}{>}{0}} and
6711                test {
6712              \ifnumcomp{\value{LWR@tabulardepth}}{=}{\value{LWR@tabularpardepth}}
6713                } and
6714                bool {LWR@intabularmetadata} and
6715                not bool {LWR@tableparcell} and
6716                test {\ifnumcomp{\value{LWR@lateximagedepth}}{=}{0}}
6717            }%
6718                {\LWR@getmynexttoken}%
6719                {}%
6720        }% LWR@doingparhooks
6721        {}% not LWR@doingparhooks
6722 % Do not place anything here, due to the above \LWR@getmynexttoken.
6723 }
```

53.1   Paragraph Hooks

para/begin (*Hook*) [LaTeX]

```
6724 \AddToHook{para/begin}[lwarp]{\LWR@openparagraph}
```

para/end (*Hook*) [LaTeX]

```
6725 \AddToHook{para/end}[lwarp]{\LWR@closeparagraph}
```

```
6726 \end{warpHTML}
```

54 Paragraph start/stop handling

These commands allow/disallow the generation of HTML paragraph tags.

Section 53 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: 6727 `\begin{warpHTML}`

\LWR@startpars Begin handling HTML paragraphs. This allows an HTML paragraph to start, but one has not yet begun.

```
6728 \newcommand*{\LWR@startpars}%
6729 {%
```

Ignore if inside a lateximage or <span>:

```
6730     \ifboolexpr{
6731         test {\ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}} or
6732         test {\ifnumcomp{\value{LWR@spandepth}}{>}{0}}
6733     }%
6734         {}% nesting
6735         {% not nesting
```

The LaTeX paragraph hook controls tag generation for the start and end of paragraphs.

See if currently handling HTML paragraphs:

```
6736         \ifboolexpr {bool{LWR@doingparhooks} and bool{LWR@doingstartpars}}%
```

If already in paragraph mode, do nothing.

```
6737             {}%
```

If not currently in paragraph mode:

```
6738             {\par}%
```

Are now handling paragraphs, but have not yet actually started one:

```
6739             \global\booltrue{LWR@doingstartpars}%
```

No <par> tag yet to undo:

```
6740             \global\boolfalse{LWR@doingapar}%
6741         }% not nesting
6742 }
```

\LWR@stoppars Stop handling HTML paragraphs. Any currently open HTML paragraph is closed, and no more will be opened.

```
6743 \newcommand*{\LWR@stoppars}%
6744 {%
```

Ignore if inside a lateximage or <span>:

```
6745    \ifboolexpr{
6746        test {\ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}} or
6747        test {\ifnumcomp{\value{LWR@spandepth}}{>}{0}}
6748    }%
6749        {}% nesting
6750        {% not nesting
```

See if currently handling HTML paragraphs:

```
6751            \ifboolexpr{bool{LWR@doingparhooks} and bool{LWR@doingapar}}%
```

if currently in an HTML paragraph:

```
6752            {%
```

Print a closing tag:

```
6753                \leavevmode\LWR@orignewline%
6754                \LWR@htmltagc{/\LWR@tagregularparagraph}%
6755                \LWR@orignewline%
```

No longer have an open HTML paragraph:

```
6756                \global\boolfalse{LWR@doingapar}%
```

Disable the special minipage & \hspace interaction until a new minipage is found:

```
6757                \global\boolfalse{LWR@minipagethispar}%
6758            }%
```

If was not in an HTML paragraph:

```
6759            {}%
```

No longer in paragraph mode:

```
6760            \global\setbool{LWR@doingstartpars}{false}%
```

No <p> tag to undo:

```
6761            \global\boolfalse{LWR@doingapar}%
6762        }% not nesting
6763 }
```

```
6764 \end{warpHTML}
```

55 Indentfirst

indentfirst (*Pkg*)   indentfirst redefines \@afterindentfalse to be \@afterindenttrue. This is re-
versed \AtBeginDocument here.

6765 \begin{warpHTML}

```
6766 \AtBeginDocument{
6767     \def\@afterindentfalse{\let\if@afterindent\iffalse}
6768     \@afterindentfalse
6769 }
6770 \let\LWR@afterindent@syntaxhighlight\fi% syntax highlighting


6771 \end{warpHTML}
```

56 Page headers and footers

for HTML & PRINT: `6772 \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.

```
6773 \newcommand{\LWR@firstpagetop}{} % for the home page alone
6774 \newcommand{\LWR@firstpagebottom}{} % for the home page alone
6775 \newcommand{\LWR@pagetop}{} % for all other pages
6776 \newcommand{\LWR@pagebottom}{}
```

\HTMLFirstPageTop {⟨*text and logos*⟩}

```
6777 \newcommand{\HTMLFirstPageTop}[1]{%
6778     \renewcommand{\LWR@firstpagetop}{#1}%
6779 }
```

\HTMLFirstPageBottom {⟨*text and logos*⟩}

```
6780 \newcommand{\HTMLFirstPageBottom}[1]{%
6781     \renewcommand{\LWR@firstpagebottom}{#1}%
6782 }
```

\HTMLPageTop {⟨*text and logos*⟩}

```
6783 \newcommand{\HTMLPageTop}[1]{%
6784     \renewcommand{\LWR@pagetop}{#1}%
6785 }
```

\HTMLPageBottom {⟨*text and logos*⟩}

```
6786 \newcommand{\HTMLPageBottom}[1]{%
6787     \renewcommand{\LWR@pagebottom}{#1}%
6788 }


6789 \end{warpall}
```

57 css

for HTML output: `6790 \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.

```
6791 \newcommand*{\LWR@currentcss}{lwarp.css}
```

`\CSSFilename` {⟨*new-css-filename.css*⟩}   Assigns the css file to be used by the following HTML pages.

```
6792 \newcommand*{\CSSFilename}[1]{%
6793     \renewcommand*{\LWR@currentcss}{#1}%
6794     \@onelevel@sanitize\LWR@currentcss%
6795 }
6796
6797 \end{warpHTML}
```

for PRINT output:
```
6798 \begin{warpprint}
6799 \newcommand*{\CSSFilename}[1]{}
6800 \end{warpprint}
```

58   MATHJAX script

for HTML output:
```
6801 \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.

```
6802 \newcommand*{\LWR@mathjaxfilename}{lwarp_mathjax.txt}
```

`\MathJaxFilename` {⟨*filename*⟩}   Assigns the MATHJAX script file to be used by the following HTML pages.

```
6803 \newcommand*{\MathJaxFilename}[1]{%
6804     \renewcommand*{\LWR@mathjaxfilename}{#1}%
6805     \@onelevel@sanitize\LWR@mathjaxfilename%
6806 }
6807
6808 \end{warpHTML}
```

for PRINT output:
```
6809 \begin{warpprint}
6810 \newcommand*{\MathJaxFilename}[1]{}
6811 \end{warpprint}
```

59   Title, HTML meta author, HTML meta description

for HTML output:
```
6812 \begin{warpHTML}
```

\title {⟨*title*⟩} Modified to remember \thetitle, which is used to set the HTML page titles.

```
6813 \let\LWR@origtitle\title
6814
6815 \renewcommand*{\title}[1]{%
6816     \LWR@origtitle{#1}%
6817     \begingroup%
6818         \renewcommand{\thanks}[1]{}%
6819         \protected@xdef\thetitle{#1}%
6820     \endgroup%
6821 }
6822 \end{warpHTML}
```

for HTML & PRINT: `6823 \begin{warpall}`

\HTMLTitle {⟨*Titlename*⟩}   The Title to place into an HTML meta tag. The default is to use the document \title's setting.

```
6824 \providecommand{\thetitle}{\BaseJobname}
6825
6826 \newcommand{\theHTMLTitle}{\thetitle}
6827
6828 \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.

```
6829 \providecommand{\theauthor}{}
6830
6831 \newcommand{\theHTMLAuthor}{\theauthor}
6832
6833 \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 \HTMLDescription, allowing different HTML descriptions 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.

```
6834 \newcommand{\LWR@currentHTMLDescription}{}
6835
6836 \newcommand{\HTMLDescription}[1]{%
6837 \renewcommand{\LWR@currentHTMLDescription}{#1}
6838 }
6839
6840 \end{warpall}
```

60 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   If using **memoir** class, with which lwarp preloads **footmisc**, the stable option must
⚠ **memoir**   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.

60.1   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 60.4 for the implementation.

60.2 Minipage footnotes

See section 60.5 for how minipage footnotes are gathered. See section 94.4 for how minipage footnotes are placed into the document.

60.3 Titlepage thanks

See section 69.7 for titlepage footnotes.

60.4 Regular page footnote implementation

for HTML & PRINT: 6841 `\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.

```
6842 \newcounter{FootnoteDepth}
6843 \setcounter{FootnoteDepth}{3}
```

footnoteReset (*Ctr*)   If non-zero, the `footnote` counter is reset to this value each time the foot-
Default: 0   notes 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.)

```
6844 \newcounter{footnoteReset}
6845 \setcounter{footnoteReset}{0}
```

```
6846 \end{warpall}
```

for HTML output: 6847 `\begin{warpHTML}`

Required for footnotes inside `description` or **amstheorem** square braces:

```
6848 \AtBeginDocument{
6849 \robustify{\footnote}
6850 \robustify{\footnotemark}
6851 }
```

\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.

6852 \newbox\LWR@footnotebox

LWR@spewingnotes (*bool*)   Used with the footnote package to suppress paragraph tags before and after
\spewnotes.

6853 \newbool{LWR@spewingnotes}% For the footnote package.

Much of the following has unneeded print-mode formatting removed.

\@makefntext {⟨*text*⟩}

6854 \long\def\@makefntext#1{^{\@thefnmark}~{#1}}

\@makefnmark

6855 \def\@makefnmark{%
6856   ^{\@thefnmark}%
6857 }

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.

6858 \long\def\LWR@@footnotetext#1#2{%
6859 \LWR@traceinfo{LWR@footnotetext}%

Perhaps generate an autopage in the text to link a citation backreference closer to
its usage.

6860 \LWR@newautopagelabel{page}%
6861 \LWR@ensuredoingapar%

Locally disable auto page labels inside the footnote text. Footnotes are accumu-
lated in the current page before finally being placed in a potentially later page, so
the autopages would be incorrect.

6862 \begingroup%
6863 \let\LWR@newautopagelabel\LWR@null@newautopagelabel%

Take the existing footnote box and add the new content:

```
6864 \global\setbox\csname #2\endcsname=\vbox{%
6865     \unvbox\csname #2\endcsname%
```

Remember the footnote number for \ref:

```
6866     \protected@edef\@currentlabel{%
6867         \csname p@footnote\endcsname\@thefnmark%
6868     }% @currentlabel
```

Open a group:

```
6869     \color@begingroup%
```

Disable CJK xpinyin while generating footnotes.

```
6870     \LWR@disablepinyin%
```

Use HTML superscripts in the footnote even when the main text is inside a lateximage, because the footnote will be in HTML:

```
6871     \renewrobustcmd{\textsuperscript}[1]{\LWR@htmlspan{sup}{##1}}%
```

Use paragraph tags if in a tabular data cell or a lateximage:

```
6872     \ifbool{LWR@spewingnotes}{}{%
6873         \LWR@htmltagc{\LWR@tagregularparagraph}\LWR@orignewline%
6874     }%
```

Append the footnote to the list:

```
6875     \@makefntext{#1}%
```

Closing paragraph tag:

```
6876     \ifbool{LWR@spewingnotes}{}{%
6877         \LWR@origtilde\LWR@orignewline%
6878         \LWR@htmltagc{/\LWR@tagregularparagraph}%
6879         \LWR@orignewline%
6880     }%
```

Close the group:

```
6881     \color@endgroup%
6882 }% vbox
6883 \endgroup%
6884 }%
```

\LWR@footnotetext {⟨*text*⟩}

```
6885 \long\def\LWR@footnotetext#1{\LWR@@footnotetext{#1}{LWR@footnotebox}}%
```

\@footnotetext {⟨*text*⟩}

```
6886 \LetLtxMacro\@footnotetext\LWR@footnotetext
```

60.5   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.

6887 \newbox\LWR@mpfootnotes

\@mpfootnotetext {⟨text⟩}

6888 \long\def\@mpfootnotetext#1{%
6889 \LWR@traceinfo{@mpfootnotetext}%
6890 \LWR@ensuredoingapar%
6891 \global\setbox\LWR@mpfootnotes\vbox{%
6892   \unvbox\LWR@mpfootnotes%
6893   \reset@font\footnotesize%
6894   \hsize\columnwidth%
6895   \@parboxrestore%
6896   \protected@edef\@currentlabel%
6897   {\csname p@mpfootnote\endcsname\@thefnmark}%
6898   \color@begingroup%

Add paragraph tag:

6899   \LWR@htmltagc{\LWR@tagregularparagraph}\LWR@orignewline%

6900   \@makefntext{%
6901   \ignorespaces#1%
6902   }%

Add the closing paragraph tag:

6903   \leavevmode\LWR@orignewline%
6904   \LWR@htmltagc{/\LWR@tagregularparagraph}%

6905   \color@endgroup%
6906 }% vbox

Paragraph handling:

6907 \LWR@ensuredoingapar%
6908 \LWR@traceinfo{@mpfootnotetext: done}%
6909 }

\thempfootnote Redefined to remove the \itshape, which caused an obscure compiling error in
some situations.

6910 \AtBeginDocument{
6911   \def\thempfootnote{\@alph\c@mpfootnote}
6912 }

60.6   Printing pending footnotes

\LWR@@printpendingfootnotes {⟨footnote counter name⟩}

```
6913 \newcommand*{\LWR@@printpendingfootnotes}[1]{%
6914 \expandafter\ifvoid\csname LWR@#1box\endcsname\else
6915     \LWR@forcenewpage
6916     \begin{BlockClass}(note){footnotes}%
```

Create a new autopage in case citation back references occur inside the footnotes:

```
6917     \LWR@newautopagelabel{page}%

6918     \null
6919     \unvbox\csuse{LWR@#1box}
6920     \setbox\csuse{LWR@#1box}=\vbox{}
6921     \end{BlockClass}
6922     \ifltxcounter{#1Reset}{%
6923         \ifnumgreater{\value{#1Reset}}{0}{%
6924             \setcounter{#1}{\value{#1Reset}}%
6925             \addtocounter{#1}{-1}%
6926         }{}%
6927     }{}%
6928 \fi
6929 }
```

`\LWR@printpendingfootnotes` Enclose the footnotes in a class, print, then clear. For manynotes, new footnotes may be added via \appto.

```
6930 \newcommand*{\LWR@printpendingfootnotes}{%
6931     \LWR@@printpendingfootnotes{footnote}%
6932 }
```

`\LWR@maybeprintpendingfootnotes` {⟨*depth*⟩} Used to print footnotes before sections only if formatting for an EPUB or word processor:

```
6933 \newcommand*{\LWR@maybeprintpendingfootnotes}[1]{%
6934 \ifboolexpr{
6935     not test{\ifnumcomp{#1}{>}{\value{FootnoteDepth}}} or
6936     bool{FormatEPUB} or
6937     bool{FormatWP}
6938 }%
6939 {\LWR@printpendingfootnotes}%
6940 {}%
6941 }
```

`\LWR@printpendingmpfootnotes` Enclose the minipage footnotes in a class, print, then clear.

```
6942 \newcommand*{\LWR@printpendingmpfootnotes}{%
6943 \ifvoid\LWR@mpfootnotes\else
6944     \LWR@forcenewpage
6945     \begin{BlockClass}(note){footnotes}%
6946     \null
6947     \unvbox\LWR@mpfootnotes
6948     \setbox\LWR@mpfootnotes=\vbox{}
6949     \end{BlockClass}
6950 \fi
6951 }
```

`\LWR@nullifyfootnotes` Cancels footnotes, such as inside an HTML comment or a \nameref.

```
6952 \newcommand*{\LWR@nullifyfootnotes}{%
6953     \renewcommand{\footnote}[2][]{}%
6954     \renewcommand{\footnotemark}[1][]{}%
6955 }
```

```
6956 \end{warpHTML}
```

61   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.

6957 \begin{warpHTML}

\marginpar [⟨*left*⟩] {⟨*right*⟩}

```
6958 \renewcommand{\marginpar}[2][]{%
6959 \ifbool{FormatWP}%
6960 {%
6961     \begin{LWR@BlockClassWP}%
6962         {width:2in; float:right; margin:10pt}{}(note){marginblock}%
6963     #2
6964     \end{LWR@BlockClassWP}%
6965 }%
6966 {%
6967     \LWR@htmlspanclass(note){marginpar}{#2}%
6968 }%
6969 }
```

\marginparBlock [⟨*left*⟩] {⟨*right*⟩}

For use when the marginpar will be more than one paragraph, and/or contains
more than simple text.

HTML version.

```
6970 \newcommand{\marginparBlock}[2][]{%
6971     \LWR@stoppars%
6972     \ifbool{FormatWP}%
6973     {%
6974         \begin{LWR@BlockClassWP}%
6975             {width:2in; float:right; margin:10pt}{}%
6976             (note){marginblock}%
6977         #2
6978         \end{LWR@BlockClassWP}
6979     }%
6980     {%
6981         \begin{BlockClass}[width:2in; float:right; margin:10pt]
6982             (note){marginparblock}%
6983         #2
```

```
6984        \end{BlockClass}
6985    }%
6986    \LWR@startpars%
6987 }
```

\reversemarginpar

```
6988 \renewcommand*{\reversemarginpar}{}
```

\normalmarginpar

```
6989 \renewcommand*{\normalmarginpar}{}
```

```
6990 \end{warpHTML}
```

for PRINT output: `6991 \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.

```
6992 \LetLtxMacro\marginparBlock\marginpar
```

```
6993 \end{warpprint}
```

62   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.

```
6994 \newcounter{LWR@currentautosecpage}
6995 \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.

```
6996 \newcounter{LWR@currentautosecfloatpage}
6997 \setcounter{LWR@currentautosecfloatpage}{1}
```

LWR@previousautopagelabel   Remembers which autopage label was most recently generated. Used to avoid
(*Ctr*)   duplicates.

```
6998 \newcounter{LWR@previousautopagelabel}
6999 \setcounter{LWR@previousautopagelabel}{-1}
```

\LWR@newautopagelabel {⟨*pagenumber counter*⟩}

\BaseJobname is added to the label in case xr or xr-hyper are used.

```
7000 \newcommand*{\LWR@newautopagelabel}[1]{%
```

No action if this autopage label has already been defined:

```
7001 \ifnumequal{\value{LWR@previousautopagelabel}}{\value{page}}%
7002     {}%
```

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.

```
7003     {%
7004         \label{\BaseJobname-autopage-\csuse{the#1}}%
```

If there are intervening pages, such as an SVG image, define another label for the new page:

```
7005         \ifnumequal{\value{#1}}{\value{page}}%
7006             {}%
7007             {\label{\BaseJobname-autopage-\csuse{thepage}}}%
```

Remember the latest autopage label:

```
7008         \setcounter{LWR@previousautopagelabel}{\value{page}}%
7009     }%
7010 }
```

\LWR@null@newautopagelabel {⟨*pagenumber counter*⟩}

Inside a footnote, the page numbers will be incorrect, so this is nullified.

```
7011 \newcommand*{\LWR@null@newautopagelabel}[1]{}
```

63   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: 7012 `\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.

```
7013 \newcounter{FileDepth}
7014 \setcounter{FileDepth}{-5}
```

CombineHigherDepths (*bool*)   Combile higher-level sections together into one file?

```
7015 \newbool{CombineHigherDepths}
7016 \booltrue{CombineHigherDepths}
```

\FilenameLimit   Maximum length of the generated filenames.

```
7017 \newcommand*{\FilenameLimit}{80}
```

```
7018 \end{warpall}
```

for HTML output: 7019 `\begin{warpHTML}`

\LWR@thisfilename   The currently-active filename or number. At first, this is the homepage.

```
7020 \AtBeginDocument{
7021 \ifbool{FileSectionNames}%
7022     {\newcommand*{\LWR@thisfilename}{\HomeHTMLFilename}}
7023     {\newcommand*{\LWR@thisfilename}{0}}
7024 }
```

\LWR@thisnewfilename   The filename being sanitized.

```
7025 \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.

```
7026 \NewDocumentCommand{\LWR@simplifyname}{s m}{%
7027 \IfBooleanTF{#1}{%
7028     \StrSubstitute{\LWR@thisnewfilename}%
7029         {\detokenize{#2}}%
7030         {\detokenize{-}}[\LWR@thisnewfilename]%
7031 }{%
7032     \StrSubstitute{\LWR@thisnewfilename}%
```

```
7033        {#2}%
7034        {\detokenize{-}}[\LWR@thisnewfilename]%
7035 }
7036 }
```

\LWR@simplifycustom User-defined filename simplifications. Redefine with \newcommand.

```
7037 \newcommand*{\LWR@simplifycustom}{}
```

\FilenameSimplify *{⟨phrase⟩} Assign a user-defined filename simplification. Appends to \LWR@simplifycustom.

```
7038 \NewDocumentCommand{\FilenameSimplify}{s m}{%
7039 \IfBooleanTF{#1}{%
7040     \appto{\LWR@simplifycustom}{%
7041         \LWR@simplifyname*{#2}%
7042     }%
7043 }{%
7044     \appto{\LWR@simplifycustom}{%
7045         \LWR@simplifyname{#2}%
7046     }%
7047 }%
7048 }
```

\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.

```
7049 \newcommand*{\LWR@avoiddupfilenames}{%
7050    To avoid duplicate filenames, use the optional\MessageBreak
7051    short Table of Contents entry:\MessageBreak
7052    \space\space\protect\section[Unique name, no math]{Name with math}%
7053        \MessageBreak
7054    or use \protect\texorpdfstring, from the hyperref package:\MessageBreak
7055    \space\space%
7056        \protect\section{\MessageBreak
7057            \space\space\space\space\protect\texorpdfstring\MessageBreak
7058                \space\space\space\space\space\space%
7059                {Name with math}{Unique name, no math}\MessageBreak
7060        \space\space}
7061 }
```

\LWR@filenamenoblanks {⟨filename⟩}

Convert blanks into dashes, removes short words, store result in
\LWR@thisfilename.

Also see \LWR@nullfonts for nullified macros.

```
7062 \newcommand*{\LWR@filenamenoblanks}[1]{%
7063 \begingroup
```

Locally temporarily disable direct-formatting commands, not used in filenames:

```
7064 \LWR@nullfonts%
7065 \renewcommand*{\LWR@htmltagc}[1]{}%
```

```
7066 \edef\LWR@thisnewfilename{#1}%
```

Replaces common macros with hyphens. (\& is done by \LWR@nullfonts.)

```
7067 \RenewDocumentCommand{\LWR@subsingledollar}{s m m m}{}%
7068 \LWR@simplifyname{\_}
7069 \LWR@simplifyname{\#}
7070 \LWR@simplifyname{\textbackslash}
7071 \LWR@simplifyname{\protect}
7072 \LWR@simplifyname{\ }
7073 \LWR@simplifyname{\textless}
7074 \LWR@simplifyname{\textgreater}
```

```
7075 \edef\LWR@thisnewfilename{\detokenize\expandafter{\LWR@thisnewfilename}}%
```

Warn if there is dollar math in the section name:

```
7076 \ifbool{FileSectionNames}{%
7077     \IfSubStr{\LWR@thisnewfilename}{\LWRdollar}{%
7078         \PackageWarning{lwarp}
7079         {%
7080             This section name:\MessageBreak
7081             \space\space``\detokenize\expandafter{#1}''\MessageBreak
7082             at the line number listed below,\MessageBreak
7083             is using $dollar-delimited math$,
7084             which generates\MessageBreak
7085             complicated file names.  It is better to use\MessageBreak
7086             \space\space%
7087             \protect\section{Name with \protect\(parenthesis math\protect\)}%
7088                 \MessageBreak
7089             The math then will be removed from the file name.\MessageBreak
7090             \MessageBreak
7091             \LWR@avoiddupfilenames%
7092             \MessageBreak
7093             This section is found before or%
7094         }
7095     }{}%
7096 }{}
```

```
7097 \LWR@traceinfo{LWR@filenamenoblanks edef: !\LWR@thisnewfilename!}%
7098 \fullexpandarg%
```

Convert spaces into hyphens:

```
7099 \LWR@simplifyname*{ }
```

Convert punctutation into hyphens:

```
7100 \LWR@simplifyname*{*}
7101 \LWR@simplifyname*{(}
7102 \LWR@simplifyname*{)}
7103 \LWR@simplifyname*{.}
7104 \LWR@simplifyname*{!}
7105 \LWR@simplifyname*{,}
7106 \LWR@simplifyname*{'}
7107 \LWR@simplifyname*{+}
7108 \LWR@simplifyname*{/}
7109 \LWR@simplifyname*{:}
7110 \LWR@simplifyname*{;}
7111 \LWR@simplifyname*{=}
```

```
7112 \LWR@simplifyname*{?}
7113 \LWR@simplifyname*{@}
7114 \LWR@simplifyname*{^}
7115 \LWR@simplifyname*{&}
7116 \LWR@simplifyname*{"}
7117 \LWR@simplifyname*{<}
7118 \LWR@simplifyname*{>}
```

```
7119 \LWR@simplifyname{\LWRbackslash}
```

Braces are removed entirely to avoid extra dashes in the result.

```
7120 \StrSubstitute{\LWR@thisnewfilename}%
7121     {\LWRleftbrace}{}[\LWR@thisnewfilename]%
7122 \StrSubstitute{\LWR@thisnewfilename}%
7123     {\LWRrightbrace}{}[\LWR@thisnewfilename]%
```

```
7124 \LWR@simplifyname{\LWRpercent}
7125 \LWR@simplifyname{\LWRdollar}
```

```
7126 \LWR@simplifyname*{|}
7127 \LWR@simplifyname*{^}
7128 \LWR@simplifyname*{~}
7129 \LWR@simplifyname*{[}
7130 \LWR@simplifyname*{]}
7131 \LWR@simplifyname*{`}
```

Convert short words:

```
7132 \LWR@simplifyname*{-s-}
7133 \LWR@simplifyname*{-S-}
7134 \LWR@simplifyname*{-a-}
7135 \LWR@simplifyname*{-A-}
7136 \LWR@simplifyname*{-an-}
7137 \LWR@simplifyname*{-AN-}
7138 \LWR@simplifyname*{-to-}
7139 \LWR@simplifyname*{-TO-}
7140 \LWR@simplifyname*{-by-}
7141 \LWR@simplifyname*{-BY-}
7142 \LWR@simplifyname*{-of-}
7143 \LWR@simplifyname*{-OF-}
7144 \LWR@simplifyname*{-and-}
7145 \LWR@simplifyname*{-AND-}
7146 \LWR@simplifyname*{-for-}
7147 \LWR@simplifyname*{-FOR-}
7148 \LWR@simplifyname*{-the-}
7149 \LWR@simplifyname*{-THE-}
```

Convert custom words:

```
7150 \LWR@simplifycustom%
```

If PDF LATEX and not utf8 encoding, don't try to convert emdash, endash:

```
7151 \ifPDFTeX% pdflatex or dvi latex
7152 \ifdefstring{\inputencodingname}{utf8}{%
7153     \LWR@simplifyname*{—}
```

```
7154 %      emdash
7155    \LWR@simplifyname*{–}
7156 %      endash
7157 }{}%
7158 \else% not PDFTeX
7159    \LWR@simplifyname*{–}
7160    \LWR@simplifyname*{–}
7161 \fi%
```

Convert multiple hyphens:

```
7162 \LWR@simplifyname*{-----}
7163 \LWR@simplifyname*{----}
7164 \LWR@simplifyname*{---}
7165 \LWR@simplifyname*{--}
```

If starts with a dash, remove the leading dash:

```
7166 \IfBeginWith{\LWR@thisnewfilename}{\detokenize{-}}{%
7167    \StrGobbleLeft{\LWR@thisnewfilename}{1}[\LWR@thisnewfilename]%
7168 }{}%
```

If ends with a dash, remove the trailing dash:

```
7169 \IfEndWith{\LWR@thisnewfilename}{\detokenize{-}}{%
7170    \StrGobbleRight{\LWR@thisnewfilename}{1}[\LWR@thisnewfilename]%
7171 }{}%
```

Limits the length of the filename:

```
7172 \StrLeft{\LWR@thisnewfilename}{\FilenameLimit}[\LWR@thisnewfilename]%
```

Return the global result:

```
7173 \global\let\LWR@thisfilename\LWR@thisnewfilename%
7174 \endgroup%
7175 \LWR@traceinfo{LWR@filenamenoblanks: result is \LWR@thisfilename}%
7176 }
```

63.1   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.

```
7177 \bgroup
7178 \catcode`!=3 \catcode`?=3
7179
7180 \long\gdef\LWR@replacestrings@addto#1#2{%
7181    \expandafter\def\expandafter#1\expandafter{#1#2}%
```

```
7182 }
7183
7184 \gdef\LWR@replacestrings#1#2{%
7185     \long\def\LWR@replacestringsA##1#1{\def\tmpb{##1}\LWR@replacestringsB}%
7186     \long\def\LWR@replacestringsB##1#1{%
7187         \ifx!##1\relax \else\LWR@replacestrings@addto\tmpb{#2##1}%
7188         \expandafter\LWR@replacestringsB\fi%
7189     }%                                              improved version <May 2016> inspired
7190     \expandafter\LWR@replacestringsA\tmpb?#1!#1% from pysyntax.tex by Petr Krajnik
7191     \long\def\LWR@replacestringsA##1?{%
7192         \def\tmpb{##1}%
7193     }\expandafter\LWR@replacestringsA\tmpb%
7194 }
7195 \egroup
```

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.

```
7196 \newbool{LWR@MathJax@silentquotes}
7197 \boolfalse{LWR@MathJax@silentquotes}
```

\LWR@subHTMLsanitize   \LWR@strresult must first be set by \LWR@HTMLsanitize, \LWR@HTMLsanitizeexpand,
            or \CustomizeMathJax.

```
7198 \catcode'\#=12
7199 \catcode'\&=12
7200 \newcommand{\LWR@subHTMLsanitize}{%
```

The &, <, and > may be interpreted by the browser:

```
7201     \edef\tmpb{\detokenize\expandafter{\LWR@strresult}}%
7202     \LWR@replacestrings{&}{&amp;}%
7203     \LWR@replacestrings{<}{&lt;}%
7204     \LWR@replacestrings{>}{&gt;}%
```

The quotes occasionally causes problems. For mathspec, also allow neutralization
of \" and the " character.

```
7205     \ifbool{LWR@MathJax@silentquotes}
7206         {%
7207             \expandafter\LWR@replacestrings\expandafter{\LWRbackslash"}{}%
7208             \LWR@replacestrings{"}{}%
7209         }%
7210         {\LWR@replacestrings{"}{&quot;}}%
7211     \LWR@replacestrings{'}{&apos;}%
7212     \LWR@replacestrings{`}{&grave;}%
```

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.

```
7213     \LWR@replacestrings{##}{#}%
7214     \edef\LWR@strresult{\detokenize\expandafter{\tmpb}}%
7215 }
```

```
7216 \catcode'\#=6
7217 \catcode'\&=4
```

`\LWR@HTMLsanitizedetokenized` {⟨*detokenized text*⟩}

Prints the sanitized text, already detokenized.

```
7218 \newrobustcmd{\LWR@HTMLsanitizedetokenized}[1]{%
7219     \LWR@traceinfo{LWR@HTMLsanitizedetokenized !#1!}%
```

Cancel French babel character handling, and fully expand the strings:

```
7220     \begingroup%
7221     \LWR@hook@processingtags%
7222     \edef\LWR@strresult{#1}%
7223     \LWR@subHTMLsanitize%
7224     \LWR@strresult%
7225     \endgroup%
7226     \LWR@traceinfo{LWR@HTMLsanitize done}%
7227 }
```

`\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.

```
7228 \edef\LWR@beginspaceleftbrace{begin \LWRleftbrace}
7229 \edef\LWR@beginspaceleftbrace{\detokenize\expandafter{\LWR@beginspaceleftbrace}}
7230 \edef\LWR@beginleftbrace{begin\LWRleftbrace}
7231 \edef\LWR@beginleftbrace{\detokenize\expandafter{\LWR@beginleftbrace}}
7232
7233 \edef\LWR@endspacerightbrace{end \LWRrightbrace}
7234 \edef\LWR@endspacerightbrace{\detokenize\expandafter{\LWR@endspacerightbrace}}
7235 \edef\LWR@endrightbrace{end\LWRrightbrace}
7236 \edef\LWR@endrightbrace{\detokenize\expandafter{\LWR@endrightbrace}}
7237
7238 \newrobustcmd{\LWR@HTMLsanitizeexpanded}[1]{%
```

Cancel French babel character handling, and fully expand the strings:

```
7239     \begingroup%
7240     \LWR@hook@processingtags%
```

```
7241     \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.

```
7242     \protect\StrSubstitute{\LWR@strresult}%
7243         {\LWR@beginspaceleftbrace}{\LWR@beginleftbrace}[\LWR@strresult]%
7244     \protect\StrSubstitute{\LWR@strresult}%
7245         {\LWR@endspacerightbrace}{\LWR@endrightbrace}[\LWR@strresult]%
```

```
7246     \LWR@subHTMLsanitize%
7247     \LWR@strresult%
7248     \endgroup%
7249 }
```

63.2 Customizing MATHJAX

\LWR@customizedMathJax Additional MATHJAX definitions to be added to the start of each HTML page.

```
7250 \newcommand*{\LWR@customizedMathJax}{}
```

LWR@warnedcustomizemathjax   Used to issue only one warning about using a \CustomizeMathJax per macro.
(*bool*)
```
7251 \newbool{LWR@warnedcustomizemathjax}
7252 \boolfalse{LWR@warnedcustomizemathjax}
```

\LWR@subcustomizedmathjax {⟨*macro definition*⟩}

```
7253 \newcommand*{\LWR@subcustomizedmathjax}[1]{%
7254     \begingroup%
7255     \LWR@hook@processingtags%
7256     \edef\LWR@strresult{\detokenize{#1}}%
7257     \LWR@subHTMLsanitize%
7258     \xdef\LWR@customizedMathJax{%
7259         \LWR@customizedMathJax%
7260             \LWR@strresult%
7261     }%
7262     \endgroup%
7263 }
7264 \@onlypreamble\LWR@subcustomizedmathjax
```

\CustomizeMathJax {⟨*macro definition*⟩}

A warning is issued if a very long argument is given.

```
7265 \newcommand*{\CustomizeMathJax}[1]{%
7266     \ifbool{LWR@warnedcustomizemathjax}{}{%
7267         \StrLen{\detokenize{#1}}[\LWR@tempone]%
7268         \ifnumgreater{\LWR@tempone}{350}{%
7269             \AtEndDocument{%
7270                 \PackageNoteNoLine{lwarp}{%
7271                 To ensure faster MathJax compilation, place each\MessageBreak
7272                 custom macro in its own \protect\CustomizeMathJax.\MessageBreak
7273                 See the Lwarp documentation regarding customizing\MessageBreak
7274                     MathJax%
7275                 }%
7276             }%
7277             \booltrue{LWR@warnedcustomizemathjax}%
7278         }{}%
7279     }%
7280     \appto\LWR@customizedMathJax{\LWRbackslash(}%
7281     \LWR@subcustomizedmathjax{#1}%
7282     \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
7283 }
7284 \@onlypreamble\CustomizeMathJax
```

\LWR@infoprocessingmathjax {⟨*package name*⟩}

```
7285 \newcommand*{\LWR@infoprocessingmathjax}[1]{%
7286 \typeout{---}
7287 \typeout{Package lwarp: Processing MathJax customizations for #1.}
```

```
7288 \typeout{\space\space This may take a moment.}
7289 \typeout{---}
7290 }
```

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.

```
7291 \providecommand{\footnotename}{footnote}
7292
7293 % due to warpMathJax:
7294 \end{warpHTML}
7295
7296 \begin{warpMathJax}
7297 \xdef\LWR@customizedMathJax{\LWR@customizedMathJax%
7298     \LWRbackslash(%
7299     \LWRbackslash{}newcommand%
7300     \{\LWRbackslash{}footnotename\}%
7301     \{\footnotename\}%
7302     \LWRbackslash)\par%
7303 }
7304 \end{warpMathJax}
```

\LWRfootnote is set per equation if a footnote is detected in the equation's math expression, otherwise it defaults to \footnotename.

```
7305 \begin{warpMathJax}
7306 \CustomizeMathJax{\def\LWRfootnote{1}}
7307 \CustomizeMathJax{\newcommand{\footnote}[2][\LWRfootnote]{{}^{\mathrm{#1}}}}
7308 \CustomizeMathJax{\newcommand{\footnotemark}[1][\LWRfootnote]{{}^{\mathrm{#1}}}}
```

\hspace is modified to accept and ignore a star:

```
7309 \CustomizeMathJax{\let\LWRorighspace\hspace}
7310 \CustomizeMathJax{\renewcommand{\hspace}{\ifstar\LWRorighspace\LWRorighspace}}
```

Various other customizations:

```
7311 \CustomizeMathJax{\newcommand{\mathnormal}[1]{{#1}}}
7312 \CustomizeMathJax{\newcommand\ensuremath[1]{#1}}
7313 \CustomizeMathJax{% absorb two optional arguments
7314     \newcommand{\LWRframebox}[2][]{\fbox{#2}}
7315     \newcommand{\framebox}[1][]{\LWRframebox}
7316 }
7317 \CustomizeMathJax{\newcommand{\setlength}[2]{}}
7318 \CustomizeMathJax{\newcommand{\addtolength}[2]{}}
7319 \CustomizeMathJax{\newcommand{\setcounter}[2]{}}
7320 \CustomizeMathJax{\newcommand{\addtocounter}[2]{}}
7321 \CustomizeMathJax{\newcommand{\arabic}[1]{}}
7322 \CustomizeMathJax{\newcommand{\number}[1]{}}
```

```
7323 \CustomizeMathJax{\newcommand{\noalign}[1]{\text{#1}\notag \\}}
7324 \CustomizeMathJax{\newcommand{\cline}[1]{}}
7325 \CustomizeMathJax{\newcommand{\directlua}[1]{\text{(directlua)}}}
7326 \CustomizeMathJax{\newcommand{\luatexdirectlua}[1]{\text{(directlua)}}}
```

\protect, \mathchar, and \delimiter are silently discarded; and \mathcode and \delcode are ignored.

```
7327 \CustomizeMathJax{\newcommand{\protect}{}}
7328 \CustomizeMathJax{\def\LWRabsorbnumber#1 {}}
7329 \CustomizeMathJax{\def\LWRabsorbquotenumber"#1 {}}
7330 \CustomizeMathJax{\newcommand{\LWRabsorboption}[1][]{}}
7331 \CustomizeMathJax{\newcommand{\LWRabsorbtwooptions}[1][]{\LWRabsorboption}}
7332 \CustomizeMathJax{\def\mathchar{\ifnextchar"\LWRabsorbquotenumber\LWRabsorbnumber}}
7333 \CustomizeMathJax{\def\mathcode#1={\mathchar}}
7334 \CustomizeMathJax{\let\delcode\mathcode}
7335 \CustomizeMathJax{\let\delimiter\mathchar}
```

Some text symbols missing from MATHJAX:

```
7336 \CustomizeMathJax{\def\oe{\unicode{x0153}}}
7337 \CustomizeMathJax{\def\OE{\unicode{x0152}}}
7338 \CustomizeMathJax{\def\ae{\unicode{x00E6}}}
7339 \CustomizeMathJax{\def\AE{\unicode{x00C6}}}
7340 \CustomizeMathJax{\def\aa{\unicode{x00E5}}}
7341 \CustomizeMathJax{\def\AA{\unicode{x00C5}}}
7342 \CustomizeMathJax{\def\o{\unicode{x00F8}}}
7343 \CustomizeMathJax{\def\O{\unicode{x00D8}}}
7344 \CustomizeMathJax{\def\l{\unicode{x0142}}}
7345 \CustomizeMathJax{\def\L{\unicode{x0141}}}
7346 \CustomizeMathJax{\def\ss{\unicode{x00DF}}}
7347 \CustomizeMathJax{\def\SS{\unicode{x1E9E}}}
7348 \CustomizeMathJax{\def\dag{\unicode{x2020}}}
7349 \CustomizeMathJax{\def\ddag{\unicode{x2021}}}
7350 \CustomizeMathJax{\def\P{\unicode{x00B6}}}
7351 \CustomizeMathJax{\def\copyright{\unicode{x00A9}}}
7352 \CustomizeMathJax{\def\pounds{\unicode{x00A3}}}
7353 \end{warpMathJax}
7354
7355
7356 \begin{warpHTML}% due to warpMathJax
```

\LWR@customizeMathJax Prints MATHJAX commands to the HTML output.

```
7357 \newcommand{\LWR@customizeMathJax}{%
7358 \ifbool{mathjax}{
7359 \LWR@stoppars
7360 \LWR@htmlcomment{MathJax customizations:}
7361
7362 \begin{BlockClass}{hidden}
7363 \LWR@stoppars
```

Avoid ligatures while printing MATHJAX customizations:

```
7364 {
7365     \LWR@print@ttfamily
7366     \LWR@customizedMathJax
```

```
7367 }
7368 \LWR@startpars
7369 \end{BlockClass}
7370
7371 \LWR@startpars
7372 }{}
7373 }
```

```
7374 \end{warpHTML}
```

for PRINT output: 7375 `\begin{warpprint}`

\CustomizeMathJax The print-mode version:

```
7376 \newcommand*{\CustomizeMathJax}[1]{}
```

\FilenameSimplify * {⟨*expression*⟩}

```
7377 \NewDocumentCommand{\FilenameSimplify}{s m}{}
```

```
7378 \end{warpprint}
```

for HTML output: 7379 `\begin{warpHTML}`

\LWR@createfooter If specified, create the first or later web page footer.

```
7380 \newcommand*{\LWR@createfooter}{%
7381     \ifnumless{\value{LWR@htmlseqfilenumber}}{1}{%
7382         \ifdefempty{\LWR@firstpagebottom}{}{%
7383             \LWR@htmlelement{footer}
7384
7385             \LWR@firstpagebottom
7386
7387             \LWR@htmlelementend{footer}
7388         }%
7389     }{%
7390         \ifdefempty{\LWR@pagebottom}{}{%
7391             \LWR@htmlelement{footer}
7392
7393             \LWR@pagebottom
7394
7395             \LWR@htmlelementend{footer}
7396         }%
7397     }%
7398 }
```

\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.

```
7399 \newcommand*{\LWR@newhtmlfile}[1]{
7400 \LWR@traceinfo{LWR@newhtmlfile}
```

At the bottom of the ending file:

```
7401 \LWR@htmlelementclassend{section}{textbody}
7402 \LWR@htmlelementclassend{main}{bodycontainer}
7403 \LWR@htmlelementclassend{div}{bodyandsidetoc}
7404
7405 \LWR@printpendingfootnotes
7406
```

No footer between files if EPUB:

```
7407 \ifbool{FormatEPUB}{}{\LWR@createfooter}
```

No bottom navigation if are finishing the home page or formatting for EPUB or a word-processor.

```
7408 \ifthenelse{\boolean{FormatEPUB}\OR\boolean{FormatWP}}
7409     {}
7410     {\ifnumcomp{\value{LWR@htmlfilenumber}}{>}{0}{\LWR@botnavigation}{}}
```

End of this HTML file:

```
7411 \LWR@stoppars
7412 \LWR@htmltag{/body}\LWR@orignewline
7413 \LWR@htmltag{/html}\LWR@orignewline
7414 \LWR@traceinfo{LWR@newhtmlfile: about to LWR@orignewpage}
7415 \LWR@maybe@orignewpage
```

```
7416 \addtocounter{LWR@htmlfilenumber}{1}%
7417 \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.

```
7418 \ifbool{FileSectionNames}%
7419 {%
```

Convert the section name to a filename with blanks and common words removed. The resulting filename is in \LWR@thisfilename.

```
7420     \LWR@filenamenoblanks{#1}%
```

Create a macro name from the MD5 hash of the file name, to detect duplicates:

```
7421     \edef\LWR@hashedname{\LWR@mdfive{\LWR@thisfilename}}%
```

If the macro name is not yet defined, this filename is unique.

```
7422     \ifcsundef{LWR@filename\LWR@hashedname}{%
```

If the filename is unique, create a macro using the hashed name, to be used to test for additional duplicates in the future.

```
7423        \csdef{LWR@filename\LWR@hashedname}{}%
7424    }{%
```

If the filename is not unique, create an error.

```
7425        \PackageError{lwarp}%
7426          {%
7427            The section name:\MessageBreak
7428            ''#1'',\MessageBreak
7429            at the line number listed below,\MessageBreak
7430            generates the filename\MessageBreak
7431            ''\LWR@thisfilename'',\MessageBreak
7432            which appears to be a duplicate.  There is a\MessageBreak
7433          previous section with an identical or similar name.\MessageBreak
7434           While generating file names, Lwarp sanitizes math,\MessageBreak
7435            most symbols, and a few common short words,\MessageBreak
7436            and this may cause a conflict.\MessageBreak
7437            Enter 'H' for possible solutions%
7438          }%
7439          {%
7440            \LWR@avoiddupfilenames%
7441          }%
7442    }%
7443 }%
```

If using file numbers instead of names, the name is set to the next file number.

```
7444 {\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.

```
7445 \LWR@traceinfo{LWR@newhtmlfile: about to print start file}%
```

\LWR@nullfonts to allow math in a section name.

```
7446 \begingroup%
7447 \LWR@nullfonts%
7448 \LWR@htmlblockcomment{%
7449 |Start file|%
7450 \LWR@htmlsectionfilename{\LWR@thisfilename}|%
7451 }
7452 \endgroup%
```

At the top of the starting file:

```
7453 \LWR@stoppars
7454
```

Start a new file with the given section name:

```
7455 \LWR@filestart[#1]
7456
```

Track the PDF page numbers of the HTML output. This is updated more frequently than LWR@currentautosecpage.

```
7457 \setcounter{LWR@currentautosecfloatpage}{\value{page}}%
7458 \LWR@newautopagelabel{LWR@currentautosecfloatpage}%
```

No navigation between files if formatting for an EPUB or word processor:

```
7459 \ifthenelse{\boolean{FormatEPUB}\OR\boolean{FormatWP}}
7460     {}
7461     {\LWR@topnavigation}
7462
```

No header if between files if formatting for an EPUB or word processor:

```
7463 \ifthenelse{\boolean{FormatEPUB}\OR\boolean{FormatWP}}
7464     {}
7465     {
7466         \ifdefempty{\LWR@pagetop}{}{
7467             \LWR@htmlelement{header}
7468
7469             \LWR@pagetop
7470
7471             \LWR@htmlelementend{header}
7472         }
7473     }
7474
```

The container for the sidetoc and text body:

```
7475 \LWR@htmlelementclass{div}{bodyandsidetoc}
```

No sideTOC if formatting for an EPUB or word processor:

```
7476 \ifthenelse{\boolean{FormatEPUB}\OR\boolean{FormatWP}}
7477     {}
7478     {\LWR@sidetoc}
7479
```

Start of the `<textbody>`:

```
7480 \LWR@htmlelementclass{main}{bodycontainer}
7481 \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.

```
7482 \boolfalse{LWR@setseqfilelabel}
```

Print title only if there is one. Skip if formatting for an EPUB or word processor:

```
7483 \ifthenelse{\boolean{FormatEPUB}\OR\boolean{FormatWP}}%
7484     {}%
7485     {%
7486         \ifcsvoid{thetitle}{}{%
7487             \LWR@printthetitle%
7488         }%
7489     }%
```

Keep paragraph tags disabled for now:

```
7490 \LWR@stoppars
7491
```

If using MATHJAX, print the customizations here.

```
7492 \LWR@customizeMathJax
```

```
7493 \LWR@traceinfo{LWR@newhtmlfile: done}
7494 }
```

```
7495 \end{warpHTML}
```

64   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:
```
7496 \begin{warpHTML}
```

64.1   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.

```
7497 \newbool{LWR@forcinghtmlpage}
7498 \boolfalse{LWR@forcinghtmlpage}
7499
7500 \newcommand*{\ForceHTMLPage}{%
7501 \global\booltrue{LWR@forcinghtmlpage}%
7502 }
```

\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.

```
7503 \newbool{LWR@forcinghtmltoc}
7504 \boolfalse{LWR@forcinghtmltoc}
7505
7506 \newcommand*{\ForceHTMLTOC}{%
7507 \global\booltrue{LWR@forcinghtmltoc}%
7508 }
```

```
7509 \end{warpHTML}
```

for PRINT output:
```
7510 \begin{warpprint}
7511 \newcommand*{\ForceHTMLPage}{}
7512 \newcommand*{\ForceHTMLTOC}{}
7513 \end{warpprint}
```

for HTML output:
```
7514 \begin{warpHTML}
```

64.2   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.

```
7515 \newbool{LWR@mainmatter}
7516 \DeclareDocumentCommand{\mainmatter}{}{%
7517 \booltrue{LWR@mainmatter}%
7518 }
```

\frontmatter Declare the front matter section of the document, using arabic numbering for the
             internal numbering. Does not reset the page number.

```
7519 \DeclareDocumentCommand{\frontmatter}{}{%
7520 \boolfalse{LWR@mainmatter}%
7521 }
```

\backmatter Declare the back matter section of the document. Does not reset the page number.

```
7522 \DeclareDocumentCommand{\backmatter}{}{%
7523 \boolfalse{LWR@mainmatter}
7524 }
```

64.3   Sectioning support macros

\LWR@sectionnumber {⟨*section type*⟩}

Typeset a section number and its trailing space with css formatting:

```
7525 \newcommand*{\LWR@sectionnumber}[1]{%
7526 \InlineClass{sectionnumber}{#1}%
7527 }
```

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.

```
7528 \newcommand*{\LWR@createautosec}[1]{%
7529 \LWR@htmltag{%
7530     #1 % space
7531     id=\textquotedbl\LWR@print@mbox{autosec-\arabic{page}}\textquotedbl%
7532 }%
7533 }
```

\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.

```
7534 \NewDocumentCommand{\LWR@pushoneclose}{m}{%
7535 \LWR@traceinfo{LWR@pushoneclose #1}%
7536     \LWR@pushclose{#1}%
7537 }
```

\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.

```
7538 \NewDocumentCommand{\LWR@startnewdepth}{m}{%
```

> Close any stacked sections up to this new one.

```
7539 \LWR@closeprevious{#1}%
```

> Push a new section depth:

```
7540 \LWR@pushoneclose{#1}%
7541 }
```

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.

```
7542 \newcounter{LWR@prevFileDepth}
7543 \setcounter{LWR@prevFileDepth}{\LWR@depthsubparagraph}
```

\@seccntformat {⟨*sectiontype*⟩}

```
7544 \def\@seccntformat#1{\csname the#1\endcsname\quad}
```

\simplechapterdelim Used by tocbibind and anonchap.

```
7545 \newcommand*{\simplechapterdelim}{}
```

\@chapcntformat {⟨*sectiontype*⟩}

> \let to \@seccntformat by default, but may be redefined by \simplechapter and \restorechapter from **tocbibind** or **anonchap**.

7546 `\let\@chapcntformat\@seccntformat`

\@partcntformat {⟨*sectiontype*⟩}

> \let to \@seccntformat by default, but may be redefined by **ctex**.

7547 `\let\@partcntformat\@seccntformat`

\@partnameformat Prints "Part" for part sections.

> Nullified by **ctex**.

7548 `\newcommand*{\@partnameformat}{\LWR@isolate{\partname}~}%`

\LWR@printchaptername Print \chaptername in most cases, but this is nullified for **ctexbook**, **komascript**, **ujt*** classes.

7549 `\newcommand*{\LWR@printchaptername}{%`
7550 `    \ifdefvoid{\chaptername}{}{\chaptername~}%`
7551 `}`

\LWR@section * [⟨*TOC name*⟩] {⟨*name*⟩} {⟨*sectiontype*⟩}

> The common actions for the high-level sectioning commands.

7552 `\DeclareDocumentCommand{\LWR@section}{m m m m}{%`
7553 `\IfValueTF{#2}%`
7554 `    {\LWR@traceinfo{LWR@section: starting #4 #2}}%`
7555 `    {\LWR@traceinfo{LWR@section: starting #4 #3}}%`

> Warn if starting a section inside a <span>:

7556 `    \LWR@spanwarninvalid{section}%`

7557 `\LWR@maybeprintpendingfootnotes{\csuse{LWR@depth#4}}%`
7558 `\LWR@stoppars%`
7559 `\LWR@startnewdepth{#4}%`

> Cancel special `minipage` horizontal space interaction:

7560 `\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:

7561 `\LWR@traceinfo{LWR@section: testing whether to start a new HTML file}%`

```
7562 \IfBooleanT{#1}{\LWR@traceinfo{LWR@section: starred}}%
7563 \ifbool{LWR@forcinghtmlpage}{\LWR@traceinfo{LWR@section: forcinghtmlpage}}{}%
7564 \ifthenelse{%
7565     \(%
7566         \(\NOT\equal{#1}{\BooleanTrue}\)\OR%
7567         \(\cnttest{\@nameuse{LWR@depth#4}}{=}{\LWR@depthpart}\)\OR%
7568         \(\boolean{LWR@forcinghtmlpage}\)%
7569     \)%
7570     \AND%
7571     \cnttest{\@nameuse{LWR@depth#4}}{<=}{\value{FileDepth}}%
7572     \AND%
7573     \(%
7574         \NOT\boolean{CombineHigherDepths}\OR%
7575         \cnttest{\@nameuse{LWR@depth#4}}{<=}{\value{LWR@prevFileDepth}}%
7576     \)%
7577     \AND%


7578     \(% phantomsection
7579         \NOT\isempty{#3}%
7580         \OR%
7581         \(\NOT\equal{#1}{\BooleanTrue}\)%
7582     \)%
7583 }%
```

If so: start a new HTML file:

```
7584 {% new file
7585     \LWR@traceinfo{LWR@section: new HTML file}%
```

See if there was an optional TOC name entry:

```
7586     \IfNoValueTF{#2}%
```

If no optional entry

```
7587         {\LWR@newhtmlfile{#3}}%
```

If yes an optional entry

```
7588         {\LWR@newhtmlfile{#2}}%
7589 }% new file
```

Else: No new HTML file:

```
7590 {% not new file
```

Generate a new LATEX page so that TOC and index page number points to the section:

```
7591     \LWR@traceinfo{LWR@section: not a new HTML file, about to LWR@orignewpage}%
7592     \LWR@maybe@orignewpage%
7593 }% not new file
7594
```

Remember this section's name for \nameref:

```
7595 \IfValueT{#3}{%
```

```
7596        \LWR@traceinfo{LWR@section: about to LWR@setlatestname}%
7597        \IfValueTF{#2}{\LWR@setlatestname{#2}}{\LWR@setlatestname{#3}}%
7598 }%
```

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.

```
7599 \begingroup%
7600 \LWR@nullfonts%
7601 \LWR@nullifyfootnotes%
7602 \LWR@htmlcomment{%
7603        \LWR@orignewline%
7604        \IfValueTF{#2}%
7605            {...... #4 #2 ......}%
7606            {...... #4 #3 ......}%
7607        \LWR@orignewline%
7608 }%
7609 \LWR@orignewline%
7610 \ifbool{HTMLDebugComments}%
7611        {%
7612            \IfBooleanTF{#1}% starred
7613                {%
7614                    \IfNoValueTF{#2}% short TOC
7615                        {\LWR@htmlcomment{Opening #4* ''#3''}}%
7616                        {\LWR@htmlcomment{Opening #4* ''#2''}}%
7617                }%
7618                {%
7619                    \IfNoValueTF{#2}% short TOC
7620                        {\LWR@htmlcomment{Opening #4 ''#3''}}%
7621                        {\LWR@htmlcomment{Opening #4 ''#2''}}%
7622                }%
7623            \LWR@orignewline%
7624        }%
7625        {}%
7626 \endgroup%
```

For inline sections paragraph and subparagraph, start a new paragraph now:

```
7627 \ifthenelse{%
7628        \cnttest{\@nameuse{LWR@depth#4}}{>=}{\LWR@depthparagraph}%
7629 }%
7630        {\LWR@startpars}%
7631        {}%
```

Create the opening tag with an autosec:

```
7632 \LWR@traceinfo{LWR@section: about to LWR@createautosec}%
7633 \LWR@createautosec{\@nameuse{LWR@tag#4}}%
```

```
7634 \setcounter{LWR@currentautosecpage}{\value{page}}%
```

Check if starred:

```
7635 \IfBooleanTF{#1}%
7636 {%
7637        \LWR@traceinfo{LWR@section: starred}%
```

Starred, but also forcing a TOC entry, so add unnumbered TOC name or regular name:

```
7638    \ifbool{LWR@forcinghtmltoc}%
7639    {%
7640        \addcontentsline{toc}{#4}{%
7641            \IfValueTF{#2}{\LWR@isolate{#2}}{\LWR@isolate{#3}}%
7642        }%
7643    }%
7644    {}%
7645 }% starred
```

Not starred, so step counter and add to TOC:

```
7646 {% not starred
```

Only add a numbered TOC entry if section number is not too deep:

```
7647    \ifthenelse{%
7648        \cnttest{\@nameuse{LWR@depth#4}}{<=}{\value{secnumdepth}}%
7649    }%
7650    {% if secnumdepth
```

If in the main matter, step the counter and add the TOC entry. For `article` class, lwarp assumes that all is mainmatter.

```
7651        \LWR@traceinfo{LWR@section: about to test main matter}%
7652        \ifbool{LWR@mainmatter}%
7653        {%
7654            \LWR@traceinfo{LWR@section: yes mainmatter}%
7655            \refstepcounter{#4}%
```

Add main matter numbered TOC entry with the TOC name or the regular name:

```
7656        \LWR@traceinfo{LWR@section: about to addcontentsline}%
7657        \addcontentsline{toc}{#4}%
7658        {%
7659            \protect\numberline{%
7660                \@nameuse{pre#4name}%
7661                \@nameuse{the#4}%
7662                \@nameuse{post#4name}%
7663            }%
7664            {%
7665                \ignorespaces%
7666            \IfValueTF{#2}{\LWR@isolate{#2}}{\LWR@isolate{#3}}\protect\relax%
7667            }%
7668        }%
7669        \LWR@traceinfo{LWR@section: finished addcontentsline}%
7670    }% end of if main matter
```

If not main matter, add unnumbered TOC name or regular name:

```
7671        {% not main matter
7672            \LWR@traceinfo{LWR@section: no main matter}%
7673            \addcontentsline{toc}{#4}{%
7674                \IfValueTF{#2}{\LWR@isolate{#2}}{\LWR@isolate{#3}}%
7675            }%
7676        }% end of not main matter
7677    }% end of secnumdepth
```

Deeper than secnumdepth, so add an unnumbered TOC entry:

```
7678    {%
7679        \addcontentsline{toc}{#4}{%
7680            \IfValueTF{#2}{\LWR@isolate{#2}}{\LWR@isolate{#3}}%
7681        }%
7682    }%
```

For part, print "Part":

```
7683    \ifbool{LWR@mainmatter}%
7684    {%
7685        \ifthenelse{%
7686            \(\cnttest{\@nameuse{LWR@depth#4}}{<=}%
7687                {\value{secnumdepth}}\) \AND%
7688            \(\cnttest{\@nameuse{LWR@depth#4}}{=}{\LWR@depthpart}\)%
7689        }%
7690        {\@partnameformat}%
7691        {}%
```

Print the section number:

```
7692        \LWR@traceinfo{LWR@section: about to print section number}%
7693        \ifthenelse{%
7694            \cnttest{\@nameuse{LWR@depth#4}}{<=}{\value{secnumdepth}}%
7695        }%
7696        {%
7697            \ifstrequal{#4}{part}%
7698            {\protect\LWR@sectionnumber{\@partcntformat{#4}}}%
7699            {%
7700                \ifstrequal{#4}{chapter}%
7701                    {%

7702                        \LWR@printchaptername%
7703                        \protect\LWR@sectionnumber{\@chapcntformat{#4}}%
7704                    }%
7705                    {\protect\LWR@sectionnumber{\@seccntformat{#4}}}%
7706            }%
7707        }%
7708        {}%
7709        \LWR@traceinfo{LWR@section: finished print section number}%
7710    }{}%
7711 }% not starred
```

Print the section name:

```
7712 \LWR@traceinfo{LWR@section: about to print the section name}%
7713 \LWR@isolate{#3}%
```

Close the heading tag, such as /H2:

```
7714 \LWR@traceinfo{LWR@section: about to close the heading tag}%
7715 \LWR@htmltag{\@nameuse{LWR@tag#4end}}%
7716 \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.

```
7717 \LWR@traceinfo{LWR@section: about to create the LaTeX label}%
7718 \setcounter{LWR@currentautosecfloatpage}{\value{page}}%
7719 \LWR@newautopagelabel{LWR@currentautosecpage}\LWR@orignewline%
```

If this is the first section found in this file, create a label for prevous/next links:

```
7720 \ifbool{LWR@setseqfilelabel}{}{%
7721     \label{\BaseJobname-autofile-\arabic{LWR@htmlseqfilenumber}}%
7722     \booltrue{LWR@setseqfilelabel}%
7723 }%
```

Start paragraph handing unless is an inline paragraph or subparagraph:

```
7724 \ifthenelse{%
7725     \cnttest{\@nameuse{LWR@depth#4}}{<}{\LWR@depthparagraph}%
7726 }%
7727     {\LWR@startpars}%
7728     {}%
```

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.

```
7729 \ifthenelse{%
7730     \NOT\equal{#1}{\BooleanTrue}\OR%
7731     \cnttest{\@nameuse{LWR@depth#4}}{=}{\LWR@depthpart}%
7732 }%
7733     {% not starred
7734         \setcounter{LWR@prevFileDepth}{\@nameuse{LWR@depth#4}}%
7735     }% not starred
7736     {}%
```

Reset to defaults if not a phantomsection:

```
7737 \ifstrempty{#3}%
7738     {}%
7739     {%
7740         \global\boolfalse{LWR@forcinghtmlpage}%
7741         \global\boolfalse{LWR@forcinghtmltoc}%
7742     }%
7743 %
7744 \LWR@traceinfo{LWR@section: done}%
7745 }
```

64.4 Pre- and post- sectioning names

\prebookname Usually null, but is used by uj* and ut* Japanese classes.
\postbookname

```
7746 \providecommand*{\prebookname}{}
7747 \providecommand*{\postbookname}{}
```

\prepartname Usually null, but is used by uj* and ut* Japanese classes.
\postpartname

```
7748 \providecommand*{\prepartname}{}
7749 \providecommand*{\postpartname}{}
```

\prechaptername Usually null, but is used by uj* and ut* Japanese classes.
\postchaptername

```
7750 \providecommand*{\prechaptername}{}
7751 \providecommand*{\postchaptername}{}
```

\presectionname Always null, but provided here for algorithmic simplicity in \LWR@section.
\postsectionname

```
7752 \providecommand*{\presectionname}{}
7753 \let\postsectionname\presectionname
7754
7755 \let\presubsectionname\presectionname
7756 \let\postsubsectionname\postsectionname
7757
7758 \let\presubsubsectionname\presectionname
7759 \let\postsubsubsectionname\postsectionname
7760
7761 \let\preparagraphname\presectionname
7762 \let\postparagraphname\postsectionname
7763
7764 \let\presubparagraphname\presectionname
7765 \let\postsubparagraphname\postsectionname
```

64.5 \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⟩}

```
7766 \newcommand{\part@preamble}{}% for koma-script
7767
7768 \DeclareDocumentCommand{\part}{s d() o o d() m}{%
7769     \LWR@section{#1}{#3}{#6}{part}%
7770
7771     \part@preamble% for koma-script
7772     \renewcommand{\part@preamble}{}%
7773 }
```

\chapter * ((⟨*2:PDF name*⟩)) [⟨*3:TOC name*⟩] [⟨*4:PDF name*⟩] ((⟨*5:PDF name*⟩)) {⟨*6:name*⟩}

```
7774 \let\@printcites\relax% for quotchap package
7775
7776 \newcommand{\chapter@preamble}{}% for koma-script
7777
7778 \@ifundefined{chapter}
7779 {}
7780 {%
7781     \DeclareDocumentCommand{\chapter}{s d() o o d() m}{%
7782         \LWR@section{#1}{#3}{#6}{chapter}%
7783
7784         \@printcites% for quotchap package
7785
7786         \chapter@preamble% for koma-script
7787         \renewcommand{\chapter@preamble}{}%
7788     }
7789 }
```

\section * ((⟨*2:PDF name*⟩)) [⟨*3:TOC name*⟩] [⟨*4:PDF name*⟩] ((⟨*5:PDF name*⟩)) {⟨*6:name*⟩}

```
7790 \DeclareDocumentCommand{\section}{s d() o o d() m}{%
7791     \LWR@section{#1}{#3}{#6}{section}%
7792 }
```

\subsection * ((⟨*2:PDF name*⟩)) [⟨*3:TOC name*⟩] [⟨*4:PDF name*⟩] ((⟨*5:PDF name*⟩)) {⟨*6:name*⟩}

```
7793 \DeclareDocumentCommand{\subsection}{s d() o o d() m}{%
7794     \LWR@section{#1}{#3}{#6}{subsection}%
7795 }
```

\subsubsection * ((⟨*2:PDF name*⟩)) [⟨*3:TOC name*⟩] [⟨*4:PDF name*⟩] ((⟨*5:PDF name*⟩)) {⟨*6:name*⟩}

```
7796 \DeclareDocumentCommand{\subsubsection}{s d() o o d() m}{%
7797     \LWR@section{#1}{#3}{#6}{subsubsection}%
7798 }
```

\paragraph * ((⟨*2:PDF name*⟩)) [⟨*3:TOC name*⟩] [⟨*4:PDF name*⟩] ((⟨*5:PDF name*⟩)) {⟨*6:name*⟩}

```
7799 \DeclareDocumentCommand{\paragraph}{s d() o o d() m}{%
7800     \LWR@section{#1}{#3}{#6}{paragraph}%
7801 }
```

\subparagraph * ((⟨*2:PDF name*⟩)) [⟨*3:TOC name*⟩] [⟨*4:PDF name*⟩] ((⟨*5:PDF name*⟩)) {⟨*6:name*⟩}

```
7802 \DeclareDocumentCommand{\subparagraph}{s d() o o d() m}{%
7803     \LWR@section{#1}{#3}{#6}{subparagraph}%
7804 }
```

```
7805 \end{warpHTML}
```

65   Starting a new file

7806 `\begin{warpall}`

\HTMLLanguage   Default language for the HTML lang tag.

```
7807 \newcommand*{\LWR@currentHTMLLanguage}{en-US}
7808
7809 \newcommand*{\HTMLLanguage}[1]{%
7810     \renewcommand*{\LWR@currentHTMLLanguage}{#1}%
7811 }
```

\theHTMLTitleSeparator   May be used inside \theHTMLTitleSection to separate the website's overall HTML title and the particular page's section name.

```
7812 \ifPDFTeX% pdflatex or dvi latex
7813     \ifdefstring{\inputencodingname}{utf8}{%
7814         \newcommand*{\theHTMLTitleSeparator}{ — }% EMdash
7815     }{%
7816         \newcommand*{\theHTMLTitleSeparator}{ -\ }% hyphen
7817     }%
7818 \else%
7819     \ifpTeX
7820         \newcommand*{\theHTMLTitleSeparator}{ -\ }% hyphen
7821     \else
7822         \newcommand*{\theHTMLTitleSeparator}{ —\ }% EMdash
7823     \fi%
7824 \fi%
```

\HTMLTitleBeforeSection   Sets the HTML page's meta title tag to show the website title before the section name.

```
7825 \newcommand*{\HTMLTitleBeforeSection}{%
7826     \def\theHTMLTitleSection{%
7827         \theHTMLTitle\theHTMLTitleSeparator\theHTMLSection%
7828     }%
7829 }
```

\HTMLTitleAfterSection   Sets the HTML page's meta title tag to show the section name before the website title.

```
7830 \newcommand*{\HTMLTitleAfterSection}{%
7831     \def\theHTMLTitleSection{%
7832         \theHTMLSection\theHTMLTitleSeparator\theHTMLTitle%
7833     }%
7834 }
```

\theHTMLTitleSection   Forms the HTML page's meta title tag. The default is to show the website title before the section name.

```
7835 \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.

7836 \newcommand*{\theHTMLSection}{}

7837 \end{warpall}

7838 \begin{warpHTML}

\LWR@filestart [⟨*section name*⟩]   Creates the opening HTML tags.

7839 \newcommand*{\LWR@filestart}[1][]{%
7840 \LWR@traceinfo{LWR@filestart !#1!}%

Locally temporarily disable direct-formatting commands:

7841 \begingroup%
7842 \LWR@nullfonts%

Save the section name for use while creating the HTML meta title tag:

7843 \edef\theHTMLSection{#1}%

Remove extra material:

7844 \StrSubstitute{\theHTMLSection}{\protect}{\detokenize{-}}[\theHTMLSection]%
7845 \StrSubstitute{\theHTMLSection}%
7846   {\detokenize{-----}}{\detokenize{-}}[\theHTMLSection]%
7847 \StrSubstitute{\theHTMLSection}%
7848   {\detokenize{----}}{\detokenize{-}}[\theHTMLSection]%
7849 \StrSubstitute{\theHTMLSection}%
7850   {\detokenize{---}}{\detokenize{-}}[\theHTMLSection]%
7851 \StrSubstitute{\theHTMLSection}%
7852   {\detokenize{--}}{\detokenize{-}}[\theHTMLSection]%

If starts with a dash, remove the leading dash:

7853 \IfBeginWith{\theHTMLSection}{\detokenize{-}}{%
7854   \StrGobbleLeft{\theHTMLSection}{1}[\theHTMLSection]%
7855 }{}%

Create the page's HTML header:

7856 \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.

7857 \LWR@htmltag{%
7858   html lang=\LWR@orig@textquotedbl\LWR@currentHTMLLanguage\LWR@orig@textquotedbl%
7859 }\LWR@orignewline

Start of the meta data:

7860 \LWR@htmltag{head}\LWR@orignewline

Charset is fixed at UTF-8:

```
7861 \LWR@htmltag{%
7862     meta charset=\LWR@orig@textquotedbl{}UTF-8\LWR@orig@textquotedbl\ /%
7863 }\LWR@orignewline
```

Author:

```
7864 \ifthenelse{\equal{\theHTMLAuthor}{}}%
7865     {}%
7866     {%
7867         \LWR@htmltag{%
7868         meta name=\LWR@orig@textquotedbl{}author\LWR@orig@textquotedbl\ % space
7869         content=\LWR@orig@textquotedbl\theHTMLAuthor\LWR@orig@textquotedbl\ /%
7870         }\LWR@orignewline%
7871     }%
```

lwarp is the generator:

```
7872 \LWR@htmltag{%
7873     meta % space
7874     name=\LWR@orig@textquotedbl{}generator\LWR@orig@textquotedbl\ % space
7875   content=\LWR@orig@textquotedbl{}LaTeX Lwarp package\LWR@orig@textquotedbl\ /%
7876 }\LWR@orignewline%
```

If there is a description, add it now:

```
7877 \ifdefempty{\LWR@currentHTMLDescription}{}{%
7878     \LWR@htmltag{%
7879      meta name=\LWR@orig@textquotedbl{}description\LWR@orig@textquotedbl\ % space
7880      content=\LWR@orig@textquotedbl\LWR@currentHTMLDescription\LWR@orig@textquotedbl\ /%
7881     }\LWR@orignewline
7882 }%
```

Mobile-friendly viewport:

```
7883 \LWR@htmltag{%
7884     meta % space
7885     name=\LWR@orig@textquotedbl{}viewport\LWR@orig@textquotedbl\ % space
7886   content=\LWR@orig@textquotedbl{}width=device-width, initial-scale=1.0\LWR@orig@textquotedbl\ /%
7887 }\LWR@orignewline
```

IE patch:

```
7888 \LWR@htmltag{!-\/-[if lt IE 9]}\LWR@orignewline
7889 \LWR@htmltag{%
7890     script % space
7891     src=\LWR@orig@textquotedbl{}%
7892         http://html5shiv.googlecode.com/svn/trunk/html5.js%
7893     \LWR@orig@textquotedbl%
7894 }%
7895 \LWR@htmltag{/script}\LWR@orignewline
7896 \LWR@htmltag{![endif]-\/-}\LWR@orignewline
```

The page's title, if there is one. A section name is also added if given.

```
7897 \ifthenelse{\equal{\theHTMLTitle}{}}%
7898     {}%
```

```
7899    {%
7900        \LWR@htmltag{title}%
7901        \ifdefempty{\theHTMLSection}%
7902            {\theHTMLTitle}%
7903            {\theHTMLTitleSection}%
7904        \LWR@htmltag{/title}\LWR@orignewline%
7905    }%
```

The page's stylesheet:

```
7906 \LWR@htmltag{%
7907    link % space
7908    rel=\LWR@orig@textquotedbl{}stylesheet\LWR@orig@textquotedbl\ % space
7909    type=\LWR@orig@textquotedbl{}text/css\LWR@orig@textquotedbl\ % space
7910    href=\LWR@orig@textquotedbl\LWR@currentcss\LWR@orig@textquotedbl\ /%
7911 }%
7912 \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.

```
7913 \ifbool{mathjax}%
7914 {%
7915    \begingroup%
7916    \LWR@restoreoriglists%
7917    \boolfalse{LWR@verbtags}%

7918        \IfFileExists{\LWR@mathjaxfilename}%
7919            {\verbatiminput{\LWR@mathjaxfilename}}%
7920            {%
7921                \PackageError{lwarp}%
7922                    {%
7923                    \protect\MathJaxFilename\space specified the file\MessageBreak
7924                        \space\space\LWR@mathjaxfilename\MessageBreak
7925                        which does not exist%
7926                    }%
7927                {Specify an existing file, or remove \protect\MathJaxFilename.}%
7928            }%

7929    \booltrue{LWR@verbtags}%
7930    \endgroup%
7931    \LWR@stoppars%
7932 }% end of mathjax
7933 {}%
```

End of the header:

```
7934 \LWR@htmltag{/head}\LWR@orignewline
```

Start of the body:

```
7935 \LWR@htmltag{body}\LWR@orignewline

7936 \endgroup%
7937 \LWR@traceinfo{LWR@filestart: done}%
7938 }
```

7939 `\end{warpHTML}`

66   Starting HTML output

7940 `\begin{warpHTML}`

`\LWR@LwarpStart` Executed at the beginning of the entire document.

The use of `\textquotedbl` instead of " improves compatibility with xeCJK.

```
7941 \catcode`\$=\active
7942 \newcommand*{\LWR@LwarpStart}
7943 {%
7944 \LWR@traceinfo{LWR@lwarpStart}
```

If formatting for a word processor, force filedepth to single-file only, force HTML debug comments off.

```
7945 \ifbool{FormatWP}{%
7946     \setcounter{FileDepth}{-5}%
7947     \boolfalse{HTMLDebugComments}%
7948 }{}
```

Expand and detokenize `\HomeHTMLFilename` and `\HTMLFilename`:

```
7949 \edef\LWR@strresult{\HomeHTMLFilename}
7950 \edef\HomeHTMLFilename{\detokenize\expandafter{\LWR@strresult}}
7951 \edef\LWR@strresult{\HTMLFilename}
7952 \edef\HTMLFilename{\detokenize\expandafter{\LWR@strresult}}
```

Force onecolumn and empty page style:

```
7953 \LWR@origonecolumn%
7954 \LWR@origpagestyle{empty}%
```

No black box for overfull lines:

```
7955 \overfullrule=0pt
```

Reduce chance of line overflow when HTML tags are added:

```
7956 \LWR@print@footnotesize%
```

In PDF output, don't allow line breaks to interfere with HTML tags:

```
7957 \LWR@print@raggedright%
7958 \LetLtxMacro{\\}{\LWR@endofline}%
```

Spread the lines for *pdftotext* to read them well:

```
7959 \linespread{1.3}%
```

For *pdftotext* to reliably identify paragraph splits:

```
7960 \setlength{\parindent}{0pt}
7961 \setlength{\parskip}{2ex}
```

For the `lateximage` record file:

```
7962 \immediate\openout\LWR@lateximagesfile=\BaseJobname-images.txt
```

Removes space around the caption in the HTML:

```
7963 \setlength{\belowcaptionskip}{0ex}
7964 \setlength{\abovecaptionskip}{0ex}
```

Redefine the plain page style to be empty when used by index pages:

```
7965 \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:

```
7966 \let\LWR@origcaption\caption
```

Not yet started any paragraph handling:

```
7967 \global\booltrue{LWR@doingparhooks}
7968 \global\boolfalse{LWR@doingapar}
7969 \global\boolfalse{LWR@doingstartpars}
```

`\color@endgroup`'s `\endgraf` was conflicting with lwarp's paragraph handling.

```
7970 \let\color@endgroup\endgroup
```

Document and page settings:

```
7971 \mainmatter
7972 \LWR@origpagenumbering{arabic}
```

Start a new HTML file and a header:

```
7973 \LWR@traceinfo{LWR@lwarpStart: Starting new file.}
7974 \LWR@filestart%
```

Tell *lwarpmk* that the lwarp package is in use. This allows *lwarpmk* to warn if usepackage{lwarp} was somehow disabled.

```
7975 \begingroup%
7976 \LWR@nullfonts%
7977 \LWR@htmlblockcomment{%
7978 |Using lwarp|%
7979 \LWR@htmlsectionfilename{\LWR@thisfilename}|%
7980 }
7981 \endgroup%
```

```
7982 \LWR@traceinfo{LWR@lwarpStart: Generating first header.}
7983 \ifdefempty{\LWR@firstpagetop}{}{%
7984     \LWR@htmltag{header}\LWR@orignewline
7985     \LWR@startpars
7986     \LWR@firstpagetop
7987     \LWR@stoppars
```

```
7988    \LWR@htmltag{/header}\LWR@orignewline
7989 }%
```

```
7990 \LWR@htmlelementclass{div}{bodywithoutsidetoc}
7991 \LWR@htmlelementclass{main}{bodycontainer}
7992 \LWR@traceinfo{LWR@lwarpStart: Generating textbody.}
7993 \LWR@htmlelementclass{section}{textbody}
```

Create a label for previous/next links, and remember it has been done:

```
7994 \booltrue{LWR@setseqfilelabel}%
7995 \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.

```
7996 \LWR@patchlists
```

Ensure that math mode is active to call lwarp's patches:

```
7997 \catcode`\$=\active
```

Required for \nameref to work with svg math:

```
7998 \immediate\write\@mainaux{\catcode`\string$\active}%
7999 \LetLtxMacro\LWR@syntaxhighlightone$% balance for editor syntax highlighting
```

Allow html paragraphs to begin:

```
8000 \LWR@startpars
```

If using MathJax, disable \ensuremath by printing a nullified definition at the start of each file, and add further customizations:

```
8001 \ifbool{mathjax}{
8002     \typeout{---}
8003     \typeout{Package lwarp:}
8004     \typeout{Processing MathJax customizations for the first HTML page.}
8005     \typeout{Later HTML pages will take the same amount of time.}
8006   \typeout{If this takes too long, see the Lwarp manual regarding customizing MathJax.}
8007 }{}
8008
8009 \LWR@customizeMathJax
8010
8011 \ifbool{mathjax}{
8012     \typeout{Done.}
8013     \typeout{---}
8014 }{}
```

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 svg image in the section name. Also, the final page after the section has been created is updated in LWR@currentautosecfloatpage.

```
8015 \setcounter{LWR@currentautosecfloatpage}{\value{page}}%
8016 \LWR@newautopagelabel{LWR@currentautosecpage}%
```

```
8017 \LWR@traceinfo{LWR@lwarpStart: done}
8018 }
8019 \catcode'\$=3% math shift until lwarp starts
```

```
8020 \end{warpHTML}
```

67 Ending HTML output

for HTML output: 8021 \begin{warpHTML}

\LWR@requesttoc {⟨*boolean*⟩} {⟨*suffix*⟩} Requests that a TOC, LOF, or LOTbe generated.

```
8022 \newcommand*{\LWR@requesttoc}[2]{%
8023 \ifbool{#1}
8024 {
8025     \expandafter\newwrite\@nameuse{tf@#2}
8026     \immediate\openout \@nameuse{tf@#2} \jobname.#2\relax
8027 }{}
8028 }
```

\LWR@LwarpEnd Final stop of all HTML output:

```
8029 \newcommand*{\LWR@LwarpEnd}
8030 {
8031 \LWR@stoppars
8032 \LWR@closeprevious{finished}
```

At the bottom of the ending file:

Print any pending footnotes:

```
8033 \LWR@printpendingfootnotes
```

Close the textbody.

(The \LWR@origtilde is in case no autopage is required for the label, which would not print anything, and something must be printed before the newline.)

```
8034 \label{\BaseJobname-autofile-last}\LWR@origtilde\LWR@orignewline
```

```
8035 \LWR@htmlelementclassend{section}{textbody}
8036 \LWR@htmlelementclassend{main}{bodycontainer}
8037 \LWR@htmlelementclassend{div}{bodyandsidetoc}
```

Create the footer if not EPUB

```
8038 \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.

```
8039 \ifthenelse{\boolean{FormatEPUB}\OR\boolean{FormatWP}}
8040     {}
8041     {
8042         \ifnumcomp{\value{LWR@htmlfilenumber}}{>}{0}{\LWR@botnavigation}{}
8043     }
```

```
8044 \LWR@stoppars% final stop of all paragraphs
```

Finish the HTML file:

```
8045 \LWR@htmltag{/body}\LWR@orignewline
8046 \LWR@htmltag{/html}\LWR@orignewline
```

Seems to be required sometimes:

```
8047 \LWR@maybe@orignewpage
8048 }
```

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**.

```
8049 \ifdef{\AddToHook}{% newer kernel
8050     \AddToHook{enddocument/info}{%
8051         \if@filesw
8052         \ifx \@multiplelabels \relax
8053             \if@tempswa
```

This is where warnings of duplicate labels would appear.

```
8054             \else
```

No duplicate labels, so safe to create images.

```
8055                 \immediate\write\LWR@lateximagesfile{%
8056                     |end|end|end|%
8057                 }%
8058             \fi
8059         \fi\fi
8060     }
8061 }% newer kernel
8062 {% older kernel
8063     \xpatchcmd{\enddocument}
8064         {%
8065             \if@tempswa
8066             \@latex@warning@no@line{Label(s) may have changed.
8067             Rerun to get cross-references right}%
8068             \fi
8069         }
8070         {%
8071             \if@tempswa
8072                 \@latex@warning@no@line{Label(s) may have changed.
8073                 Rerun to get cross-references right}%
8074             \else
```

No duplicate labels, so safe to create images.

```
8075                    \immediate\write\LWR@lateximagesfile{%
8076                        |end|end|end|%
8077                    }%
8078                \fi
8079            }
8080        {}
8081        {
8082            \AtEndDocument{
8083                \PackageWarningNoLine{lwarp}
8084                {%
8085                    Could not patch \protect\enddocument.\MessageBreak
8086                If labels have changed, be sure to recompile before\MessageBreak
8087                    creating lateximages with\MessageBreak
8088                    \space\space lwarpmk limages,\MessageBreak
8089                    or the images may be corrupt%
8090                }
8091            }
8092        }
8093 }% older kernel
```

68   Nullifying foreground/background hooks

See **texdoc lthooks-doc** and **textdoc ltshipout-doc**.

shipoout/background (*Hook*)   Nullified.
[LaTeX]
shipoout/foreground (*Hook*)   Nullified.
[LaTeX]

```
8094 \ifdef{\RemoveFromHook}{
8095    \AfterEndPreamble{
8096        \IfHookEmptyTF{shipout/background}{}{
8097            \PackageInfo{lwarp}{Removing background hook}
8098            \RemoveFromHook{shipout/background}[*]
8099        }
8100        \IfHookEmptyTF{shipout/foreground}{}{
8101            \PackageInfo{lwarp}{Removing foreground hook}
8102            \RemoveFromHook{shipout/foreground}[*]
8103        }
8104    }
8105 }{}
```

```
8106 \end{warpHTML}
```

69   Title page

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 69.8.

affiliation   lwarp provides for the \author macro an additional \affiliation macro to pro-

vide 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.

69.1 Setting the title, etc.

The following provide setting commands for both HTML and print outputs.

\author {⟨*author*⟩} While using \maketitle and print mode, the author is treated as
\and 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 \@title, \@author, and \@date store the values as originally assigned, includ-
\@author ing any \thanks, \and, or \affiliation. These are low-level macros intended to
\@date 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 \printtitle, etc. are user-level macros intended to be used in custom
\printauthor titlepage or titlingpage environments in cases where \maketitle is not de-
\printdate sired. These commands preserve the \thanks, etc., and should not be used in the main text.

\thetitle \thetitle, \theauthor, and \thedate are available if titling has been loaded,
\theauthor and are sanitized user-level versions from which have been removed the \thanks
\thedate and \affiliation, and \and is changed for inline text usage. The author is printed
\HTMLPageBottom 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.

69.2 \if@titlepage

for HTML & PRINT: 8107 \begin{warpall}

\if@titlepage Some classes do not provide \if@titlepage. In this case, provide it and force it false.

```
8108 \ifcsvoid{@titlepagefalse}{
8109     \newif\if@titlepage
8110     \@titlepagefalse
8111 }{}
```

```
8112 \end{warpall}
```

69.3 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: 8113 \begin{warpall}
8114 \providerobustcmd\affiliation[1]{}
8115 \end{warpall}

for PRINT output: 8116 \begin{warpprint}

```
8117 \AtBeginEnvironment{titlepage}{
8118 \renewrobustcmd\affiliation[1]{\\ \textsc{\small#1}}
8119 }
8120
8121 \AtBeginDocument{
8122 \IfPackageLoadedTF{titling}{
8123 \AtBeginEnvironment{titlingpage}{
8124 \renewrobustcmd\affiliation[1]{\\ \textsc{\small#1}}
```

8125 }
8126 }{}% titling loaded
8127 }% AtBeginDocument

8128 \end{warpprint}

8129 \begin{warpHTML}

titlepage (*env.*) Sets up a <div> of class titlepage. Provided even for memoir class, since it is used by \maketitle.

8130 \DeclareDocumentEnvironment{titlepage}{}
8131 {%
8132   \renewrobustcmd{\affiliation}[1]{\\ \InlineClass{affiliation}{##1}}%
8133   \LWR@printpendingfootnotes
8134   \LWR@forcenewpage
8135   \BlockClass{titlepage}
8136 }
8137 {
8138   \endBlockClass
8139   \LWR@printpendingfootnotes
8140 }

8141 \end{warpHTML}

69.4   Printing the thanks

\printthanks Forces the \thanks to be printed. This is necessary in a titlingpage environment when \maketitle was not used.

8142 \begin{warpprint}
8143 \newcommand*{\printthanks}{\@thanks}
8144 \end{warpprint}

8145 \begin{warpHTML}
8146 \newcommand*{\printthanks}{\LWR@stoppars\@thanks\LWR@startpars}
8147 \end{warpHTML}

69.5   Printing the title, etc. in HTML

The following are for printing the title, etc. in a titlepage or a titlingpage in HTML:

8148 \begin{warpHTML}

\printtitle

8149 \newcommand*{\printtitle}
8150 {%
8151   \LWR@stoppars%
8152   \LWR@htmltag{\LWR@tagtitle}%
8153   \@title%
8154   \LWR@htmltag{\LWR@tagtitleend}%
8155   \LWR@startpars%
8156 }

\LWR@printthetitle A private version which prints the title without footnotes, used to title each HTML page.

```
8157 \newcommand*{\LWR@printthetitle}
8158 {%
8159     \LWR@stoppars%
8160     \LWR@htmltag{\LWR@tagtitle}%
8161     \thetitle%
8162     \LWR@htmltag{\LWR@tagtitleend}%
8163     \LWR@startpars%
8164 }
```

\printauthor HTML version.

```
8165 \newcommand*{\printauthor}{
```

The entire author block is contained in a <div> named author:

```
8166 \begin{BlockClass}{author}
```

\and finishes one author and starts the next:

```
8167 \renewcommand{\and}{%
8168 \end{BlockClass}
8169 \begin{BlockClass}{oneauthor}
8170 }
```

Individual authors are contained in a <div> named oneauthor:

```
8171 \begin{BlockClass}{oneauthor}
8172 \@author
8173 \end{BlockClass}
8174 \end{BlockClass}
8175 }
```

\printdate

```
8176 \newcommand*{\printdate}{%
8177 \begin{BlockClass}{titledate}
8178 \@date
8179 \end{BlockClass}
8180 }
```

```
8181 \end{warpHTML}
```

69.6   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: ` 8182 \begin{warpprint}`

\printtitle

```
8183 \newcommand*{\printtitle}{{\Huge\@title}}
```

\printauthor   Print mode.

```
8184 \newcommand*{\printauthor}
8185     {{\large\begin{tabular}[t]{c}\@author\end{tabular}}}
```

\printdate

```
8186 \newcommand*{\printdate}{{\small\textit{\@date}}}
```

```
8187 \end{warpprint}
```

69.7   \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 60 for other kinds of footnotes.

See \thanksmarkseries{series}, below, to set the style of the footnote marks.

for HTML output: 8188 \begin{warpHTML}

```
8189 \IfClassLoadedTF{memoir}
8190 {
8191 \newcommand{\LWR@setfootnoteseries}{%
8192     \renewcommand\thefootnote{\@arabic\c@footnote}%
8193 }
8194 }{% not memoir
8195 \if@titlepage
8196 \newcommand{\LWR@setfootnoteseries}{%
8197     \renewcommand\thefootnote{\@arabic\c@footnote}%
8198 }
8199 \else
8200 \newcommand{\LWR@setfootnoteseries}{%
8201     \renewcommand\thefootnote{\@fnsymbol\c@footnote}%
8202 }
8203 \fi
8204 }% not memoir
```

\LWR@maketitlesetup   Patches \thanks macros.

```
8205 \newcommand*{\LWR@maketitlesetup}{%
```

Redefine the footnote mark:

```
8206 \LWR@setfootnoteseries%
8207 \def\@makefnmark{%
8208     \textsuperscript{\thefootnote}%
8209 }
```

$$\text{\thefootnote} \Rightarrow \text{\nameuse{arabic}{footnote}, or}$$
$$\text{\thefootnote} \Rightarrow \text{\nameuse{fnsymbol}{footnote}}$$

Redefine the footnote text:

```
8210 \long\def\@makefntext##1{%
```

Make the footnote mark and some extra horizontal space for the tags:

8211 `\textsuperscript{\@thefnmark}~%`

$$\backslash\text{makethanksmark} \Rightarrow \backslash\text{thanksfootmark} \Rightarrow \backslash\text{tamark} \Rightarrow$$
$$\backslash\text{@thefnmark} \Rightarrow \backslash\text{itshape a (or similar)}$$

Print the text:

8212 `{##1}%`
8213 `}%`
8214 `}`

\@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*.

8215 `\def\LWR@HTML@@fnsymbol#1{%`
8216 `    \ifcase#1\or *\or`
8217 `    \HTMLentity{dagger}\or`
8218 `    \HTMLentity{Dagger}\or`
8219 `    \HTMLentity{sect}\or`
8220 `    \HTMLentity{para}\or`
8221 `    \HTMLunicode{2016}\or`
8222 `    **\or`
8223 `    \HTMLentity{dagger}\HTMLentity{dagger} \or`
8224 `    \HTMLentity{Dagger}\HTMLentity{Dagger} \else`
8225 `    \@ctrerr\fi%`
8226 `}`
8227 `\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.

8228 `\newcommand*{\LWR@maketitle}{%`

An HTML titlepage <div> is used for all classes.

8229 `\begin{titlepage}`

Set up special patches:

8230 `\LWR@maketitlesetup`

Typeset the title, etc:

8231 `\@maketitle`

Immediately generate any \thanks footnotes:

8232 `\LWR@stoppars\@thanks\LWR@startpars`

Close the HTML titlepage div and cleanup:

```
8233 \end{titlepage}
8234 \setcounter{footnote}{0}%
8235 \global\let\thanks\relax
8236 \global\let\maketitle\relax
8237 \global\let\@maketitle\relax
8238 \global\let\@thanks\@empty
8239 \global\let\@author\@empty
8240 \global\let\@date\@empty
8241 \global\let\@title\@empty
8242 \global\let\title\relax
8243 \global\let\author\relax
8244 \global\let\date\relax
8245 \global\let\and\relax
8246 }
8247
8248 \LetLtxMacro\maketitle\LWR@maketitle
```

\@maketitle HTML mode. Typesets the title, etc.:

```
8249 \providecommand*{\@maketitle}{}
8250 \renewrobustcmd{\@maketitle}{%
8251     \LWR@stoppars%
8252     \LWR@htmltag{\LWR@tagtitle}%
8253     \@title%
8254     \LWR@htmltag{\LWR@tagtitleend}%
8255     \LWR@startpars%
8256     \begin{BlockClass}{author}%
```

For IEEEtran class:

```
8257     \renewcommand*{\cr}{}%
8258     \renewcommand*{\crcr}{}%
8259     \renewcommand*{\noalign}{}%

8260         \renewcommand{\and}{%
8261             \end{BlockClass}%
8262             \begin{BlockClass}{oneauthor}%
8263         }%
8264         \begin{BlockClass}{oneauthor}%
8265             \@author%
8266         \end{BlockClass}%
8267     \end{BlockClass}%
8268     \begin{BlockClass}{titledate}%
8269     \@date%
8270     \end{BlockClass}%
8271 }
```

\LWR@titlingmaketitle \maketitle for use inside an HTML titlingpage environment.

```
8272 \newcommand*{\LWR@titlingmaketitle}{%
```

Keep pending footnotes out of the title block:

```
8273 \LWR@stoppars\@thanks\LWR@startpars
```

Set up special patches:

8274 \LWR@maketitlesetup

Typeset the title, etc:

8275 \@maketitle

Immediately generate any \thanks footnotes:

8276 \LWR@stoppars\@thanks\LWR@startpars
8277 }

8278 \end{warpHTML}

69.8 \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: 8279 \begin{warpall}

\AddSubtitlePublished Adds \published and \subtitle, and related.

8280 \newcommand*{\AddSubtitlePublished}{%
8281 \IfPackageLoadedTF{titling}{% yes titling package
8282   \newcommand{\@published}{}%
8283   \newcommand{\published}[1]{\gdef\@published{##1}}%
8284   \renewcommand*{\maketitlehooka}{\printpublished}%
8285   \newcommand*{\printpublished}{%
8286   \warpprintonly{\begin{center}\unskip}%
8287   \begin{BlockClass}{published}%
8288   \warpprintonly{\large\itshape}%
8289   \@published%
8290   \end{BlockClass}%
8291   \warpprintonly{\end{center}}%
8292   }%
8293   \newcommand{\@subtitle}{}%
8294   \newcommand{\subtitle}[1]{\gdef\@subtitle{##1}}%
8295   \renewcommand*{\maketitlehookb}{\printsubtitle}%
8296   \newcommand*{\printsubtitle}{%
8297   \warpprintonly{\begin{center}\unskip}%
8298   \begin{BlockClass}{subtitle}%
8299   \warpprintonly{\Large\itshape}%
8300   \@subtitle%

```
8301          \end{BlockClass}%
8302          \warpprintonly{\end{center}}%
8303     }%
8304 }% yes titling package
8305 {% no titling package


8306     \def\@published{}%
8307     \DeclareDocumentCommand{\published}{m}{\gdef\@published{##1}}%
8308     \DeclareDocumentCommand{\printpublished}{}{}%
8309     \def\@subtitle{}%
8310     \DeclareDocumentCommand{\subtitle}{m}{\gdef\@subtitle{##1}}%
8311     \DeclareDocumentCommand{\printsubtitle}{}{}%
8312 }% no titling package
8313 }% \AddSubtitlePublished


8314 \end{warpall}
```

70 Abstract

The following code replaces the LATEX default, and will itself be replaced later if the abstract package is loaded.

`8315 \begin{warpHTML}`

`\abstractname` User-redefinable title for the abstract.

Also over-written by the babel package.

`8316 \providecommand*{\abstractname}{Abstract}`

Some classes allow an optional name, so it is allowed here.

abstract (*env.*)

```
8317 \DeclareDocumentEnvironment{abstract}{O{\abstractname}}
8318 {
8319     \LWR@forcenewpage
8320     \BlockClass{abstract}
8321     \BlockClassSingle{abstracttitle}{#1}
8322 }
8323 {
8324     \endBlockClass
8325 }
```

`8326 \end{warpHTML}`

71 Quote and verse

71.1 Attributions

`\attribution` {⟨*name*⟩}

For use with quote, quotation, verse:

Ex: "A quotation." \attribution{\textsc{Author Name}\\\textsl{Book Title}}

for HTML & PRINT:
```
8327 \begin{warpall}
8328 \newcommand{\attribution}[1]{
8329     \begin{flushright}
8330     \unskip
8331     #1
8332     \end{flushright}%
8333 }
8334 \end{warpall}
```

for HTML output:
```
8335 \begin{warpHTML}
8336 \newcommand{\LWR@HTML@attribution}[1]{%
8337     \LWR@stoppars%
8338     \begin{BlockClass}{attribution}
8339     #1
8340     \end{BlockClass}
8341     \LWR@startpars%
8342 }
8343 \LWR@formatted{attribution}
8344 \end{warpHTML}
```

71.2   Quotes, quotations

for HTML output:
```
8345 \begin{warpHTML}
```

quote (*env.*)

```
8346 \newenvironment*{LWR@HTML@quote}
8347 {
8348     \LWR@forcenewpage
8349     \LWR@htmlblocktag{blockquote}
8350 }
8351 {\LWR@htmlblocktag{/blockquote}}
8352
8353 \LWR@formattedenv{quote}
```

quotation (*env.*)

```
8354 \newenvironment*{LWR@HTML@quotation}
8355 {
8356     \LWR@forcenewpage
8357     \LWR@htmlblocktag{blockquote}
8358 }
8359 {\LWR@htmlblocktag{/blockquote}}
8360
8361 \LWR@formattedenv{quotation}
8362 \end{warpHTML}
```

71.3   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
\HTMLvleftskip (*Len*) and \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 op-
tion) 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.

71.3.1 LᴬTEX core verse environment

for HTML output: 8363 \begin{warpHTML}

verse (*env.*)

```
8364 \newenvironment{LWR@HTML@verse}
8365            {\let\\\newline% lwarp
8366             \list{}{\itemsep       \z@
8367                     \itemindent   -1.5em%
8368                     \listparindent\itemindent
8369                     \rightmargin  \leftmargin
8370                     \advance\leftmargin 1.5em}%
8371             \item\relax}
8372            {\endlist}
8373
8374 \LWR@formattedenv{verse}
```

8375 \end{warpHTML}

for HTML & PRINT: 8376 \begin{warpall}

71.3.2 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.

```
8377 \newlength{\HTMLvleftskip}
8378 \setlength{\HTMLvleftskip}{1em}
```

\HTMLleftmargini (*Len*)   Sets \leftmargini inside a verse environment in HTML.

```
8379 \newlength{\HTMLleftmargini}
8380 \setlength{\HTMLleftmargini}{4.5em}
```

```
8381 \end{warpall}
```

72   Verbatim and tabbing

for HTML & PRINT: 8382 \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.

```
8383 \newlength{\VerbatimHTMLWidth}
8384 \setlength{\VerbatimHTMLWidth}{4in}
8385 \end{warpall}
```

for HTML output: 8386 \begin{warpHTML}

LWR@verbtags (*bool*)   Used to temporarily turn off verbatim tags while doing \verbatiminput in the HTML head.

```
8387 \newbool{LWR@verbtags}
8388 \booltrue{LWR@verbtags}
```

\verb Patched to encapsulate the verbatim text inside span with a class of texttt.

```
8389 \LetLtxMacro\LWR@orig@verb@egroup\verb@egroup
8390
8391 \def\LWR@verb@egroup@endspan{%
8392     \LWR@orig@verb@egroup%
8393     \LWR@htmltag{/span}%
8394     \endgroup%
8395 }
```

```
8396 \xpretocmd{\verb}
8397     {%
8398         \begingroup%
8399         \LWR@htmltag{span class=\textquotedbl{}texttt\textquotedbl}%
8400         \let\verb@egroup\LWR@verb@egroup@endspan%
8401     }
8402     {}
8403     {\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.

```
8404 \newcommand*{\LWR@atbeginverbatim}[2][]
8405 {%
```

Stop generating html paragraph tags:

```
8406 \LWR@stoppars%
```

Avoid excessive space between lines:

```
8407 \setlength{\parskip}{0ex}%
8408 \setlength{\topsep}{0pt}%
8409 \setlength{\partopsep}{0pt}%
```

Inside the verbatim, temporarily prevent underfull \hbox warnings.

```
8410          \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.

```
8411 \ifbool{LWR@verbtags}{%
8412     \LWR@htmltag{pre class=\textquotedbl#2\textquotedbl%
8413         \ifthenelse{\equal{#1}{}}{}{ style=\textquotedbl#1\textquotedbl}%
8414     }%
8415     \par%
8416 }{}%
```

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.

```
8417 \begingroup%
```

```
8418 \LWR@print@normalfont%
8419 \LWR@origttfamily%
8420 \LWR@print@scriptsize%
```

Since inside a <pre>, restore the original list processing:

```
8421 \LWR@restoreoriglists%
```

Turn off babel-french extra space before punctuation:

```
8422 \LWR@hook@processingtags%
```

Do not produce html tags for \hspace inside a verse par. Restore plain LaTeX \hspace functionality:

```
8423 \let\hspace\LWR@print@hspace%
8424 }
```

\LWR@afterendverbatim Finishes enclosing a verbatim environment.

```
8425 \newcommand*{\LWR@afterendverbatim}{%
8426 \endgroup%
8427 \par%
```

At the end of the environment, close the pre:

```
8428 \ifbool{LWR@verbtags}{%
8429     \noindent\LWR@htmltag{/pre}\par% pre
8430 }{}%
```

Resume regular paragraph handling:

```
8431 \LWR@startpars%
8432 }
```

\verbatiminput {⟨*filename*⟩}

Patch \verbatiminput to add HTML tags:

```
8433 \newcommand{\LWR@HTML@verbatim@input}[2]{%
8434     \ifbool{LWR@verbtags}{\LWR@forcenewpage}{}%
8435     \LWR@atbeginverbatim{Verbatim}%
8436     \LWR@print@verbatim@input{#1}{#2}%
8437     \LWR@afterendverbatim%
8438 }
8439
8440 \LWR@formatted{verbatim@input}
```

verbatim (*env.*)

```
8441 \AfterEndPreamble{
8442 \LWR@traceinfo{Patching verbatim.}
8443 \AtBeginEnvironment{verbatim}{%
8444     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
8445         {}%
8446         {%
8447             \LWR@forcenewpage%
8448             \LWR@atbeginverbatim{verbatim}%
8449         }%
8450 }
8451 \AfterEndEnvironment{verbatim}{%
8452     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
8453         {}%
8454         {%
8455             \LWR@afterendverbatim%
8456         }%
8457 }
8458 %
8459 \AtBeginEnvironment{verbatim*}{%
8460     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
8461         {}%
8462         {%
8463             \LWR@forcenewpage%
8464             \LWR@atbeginverbatim{verbatim}%
8465         }%
8466 }
8467 \AfterEndEnvironment{verbatim*}{%
8468     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
```

```
8469        {}%
8470        {%
8471            \LWR@afterendverbatim%
8472        }%
8473 }
8474 }
```

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.

```
8475 \newenvironment*{LWR@HTML@tabbing}
8476 {%
8477        \LWR@forcenewpage%
8478        \LWR@atbeginverbatim{tabbing}%
8479        \let\enskip\LWR@print@enskip%
8480        \let\quad\LWR@print@quad%
8481        \let\qquad\LWR@print@qquad%
8482        \let~\LWR@origtilde%
8483        \let\,\LWR@origcomma%
8484        \let\thinspace\LWR@print@thinspace%
8485        \let\negthinspace\LWR@print@negthinspace%
8486        \LWR@print@tabbing%
8487 }
8488 {%
8489        \endLWR@print@tabbing%
8490        \LWR@afterendverbatim%
8491 }
8492
8493 \LWR@formattedenv{tabbing}
8494 \end{warpHTML}
```

73 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: 8495 \begin{warpHTML}

\@begintheorem {⟨*name*⟩} {⟨*number*⟩}

```
8496 \renewcommand{\@begintheorem}[2]{%
8497 \LWR@forcenewpage
```

```
8498        \LWR@printpendingfootnotes%                          lwarp
```

```
8499 \BlockClass{theoremcontents}
8500 \trivlist
8501 \item[\InlineClass{theoremlabel}{#1\ #2\ }]\itshape
8502 }
```

\@opargbegintheorem {⟨*name*⟩} {⟨*number*⟩} {⟨*oparg*⟩}

> LaTeX defines this, but amsthm \relaxes it, so it will not be defined if amsthm is loaded before lwarp.

```
8503 \ifundef{\@opargbegintheorem}{}{
8504     \renewcommand{\@opargbegintheorem}[3]{%
8505         \LWR@forcenewpage
8506         \BlockClass{theoremcontents}
8507         \trivlist
8508         \item[\InlineClass{theoremlabel}{#1\ #2\ (#3)\ }]\itshape
8509     }
8510 }
```

\@endtheorem

```
8511 \renewcommand*{\@endtheorem}{%
8512 \endtrivlist
```

```
8513        \LWR@printpendingfootnotes%                          lwarp
```

```
8514 \endBlockClass% theoremcontents
8515 }
```

```
8516 \end{warpHTML}
```

74 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}
```

\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.

74.1   List environment

8517 \begin{warpHTML}

\LWR@printcloselist   May be locally redefined by enumerate or description.

          8518 \newcommand*{\LWR@printcloselist}{\LWR@printcloseitemize}

\LWR@printopenlist   May be locally redefined by enumerate or description.

          8519 \newcommand*{\LWR@printopenlist}{%
          8520   ul style=\textquotedbl\LWR@print@mbox{list-style-type:none}\textquotedbl{}%
          8521 }

       \@mklab   Removes PDF spacing.

          8522 \AtBeginDocument{
          8523 \def\@mklab#1{%
          8524 %   \hfil %
          8525   #1}
          8526 \let\makelabel\@mklab
          8527 }

  \@donoparitem   Modified for HTML output by replacing TEX boxes with plain text. Also removes
               PDF spacing.

          8528 \def\@donoparitem{%
          8529 \@noparitemfalse
          8530 %   \global\setbox\@labels\hbox{\hskip -\leftmargin
          8531 %   \unhbox\@labels
          8532 %   \hskip \leftmargin}%
          8533 % \if@minipage\else
          8534 %   \@tempskipa\lastskip
          8535 %   \vskip -\lastskip
          8536 %   \advance\@tempskipa\@outerparskip
          8537 %   \advance\@tempskipa -\parskip
          8538 %   \vskip\@tempskipa
          8539 % \fi
          8540 }

        \@item   Modified for HTML output by replacing TEX boxes with plain text. Also removes
               PDF spacing.
```

```
8541 \def\LWR@HTML@item[#1]{%
8542 \LWR@traceinfo{@item}%
8543   \if@noparitem
8544     \@donoparitem
8545   \else
8546 %     \if@inlabel
8547 %       \indent
8548 %     \fi
8549     \ifhmode
8550 %       \unskip\unskip
8551     \fi
8552     \if@newlist
8553       \if@nobreak
8554         \@nbitem
8555       \else
8556 %         \addpenalty\@beginparpenalty
8557 %         \addvspace\@topsep
8558 %         \addvspace{-\parskip}%
8559       \fi
8560     \else
8561 %       \addpenalty\@itempenalty
8562 %       \addvspace\itemsep
8563     \fi
8564     \global\@inlabeltrue
8565   \fi
8566 %   \everypar{%
8567     \@minipagefalse
8568     \global\@newlistfalse

8569 %     \if@inlabel
8570 %       \global\@inlabelfalse

8571 %       {\setbox\z@\lastbox
8572 %        \ifvoid\z@
8573 %          \kern-\itemindent
8574 %        \fi}%

8575 %       \box\@labels
8576 %       \penalty\z@
8577 %     \fi

8578 %     \if@nobreak
8579 %       \@nobreakfalse
8580 %       \clubpenalty \@M
8581 %     \else
8582 %       \clubpenalty \@clubpenalty
8583 %       \everypar{}%
8584 %     \fi}%

8585   \if@noitemarg
8586     \@noitemargfalse
8587     \if@nmbrlist

8588       \refstepcounter\@listctr
8589     \fi
8590   \fi

8591     \makelabel{#1} % extra space
```

```
8592 %     \sbox\@tempboxa{\makelabel{#1}}%
8593 %     \global\setbox\@labels\hbox{%
8594 %       \unhbox\@labels
8595 %       \hskip \itemindent
8596 %       \hskip -\labelwidth
8597 %       \hskip -\labelsep
8598 %       \ifdim \wd\@tempboxa >\labelwidth
8599 %         \box\@tempboxa

8600 %       \else
8601 %         \hbox to\labelwidth {\unhbox\@tempboxa}%
8602 %       \fi
8603 %       \hskip \labelsep}%
8604   \ignorespaces%
8605 }
```

\@nbitem

```
8606 \def\@nbitem{%
8607 %   \@tempskipa\@outerparskip
8608 %   \advance\@tempskipa -\parskip
8609 %   \addvspace\@tempskipa
8610 }
```

\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.

```
8611 \newcommand*{\LWR@listitem}{%
8612   \LWR@stoppars%
8613   \LWR@startnewdepth{listitem}%
8614   \LWR@htmltag{li}%
8615   \LWR@orignewline%
8616   \LWR@startpars%
8617   \LWR@ensuredoingapar%
8618   \LWR@origitem%
8619 }
```

\LWR@nulllistfills Nullifies various TEX fill commands, in case they are used inside \makelabel. Problems are caused when these are nullified all the time.

```
8620 \newcommand*{\LWR@nulllistfills}{%
8621   \renewcommand*{\hss}{}%
8622   \renewcommand*{\llap}[1]{##1}%
8623   \renewcommand*{\rlap}[1]{##1}%
8624   \renewcommand*{\hfil}{}%
8625   \renewcommand*{\hfilneg}{}%
8626   \renewcommand*{\hfill}{}%
8627 }
```

list (*env.*) {⟨*label*⟩} {⟨*commands*⟩}

```
8628 \newcommand*{\LWR@liststart}{%
8629   \LWR@traceinfo{LWR@liststart}%
```

```
8630    \LWR@stoppars%
8631    \LWR@pushoneclose{list}%
8632    \LWR@htmltag{\LWR@printopenlist}\LWR@orignewline%
8633    \LWR@startpars%
8634    \setlength{\topsep}{0pt}%
8635    \setlength{\partopsep}{0pt}%
8636    \setlength{\itemsep}{0pt}%
8637    \setlength{\parsep}{0pt}%
8638    \setlength{\leftmargin}{0pt}%
8639    \setlength{\rightmargin}{0pt}%
8640    \setlength{\listparindent}{0pt}%
8641    \setlength{\itemindent}{0pt}%
8642    \setlength{\labelsep}{1em}%
8643    \LWR@nulllistfills%
8644 }


8645 \newcommand*{\LWR@listend}{%
8646    \LWR@traceinfo{LWR@listend}%
8647    \LWR@stoppars%
8648    \LWR@closeprevious{list}%
8649    \LWR@startpars%
8650 }
```

74.2   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.

```
8651 \newcommand*{\LWR@itemizeitem}{%
8652    \LWR@stoppars%
8653    \LWR@startnewdepth{listitem}%
8654    \LWR@htmltag{li}%
8655    \LWR@orignewline%
8656    \LWR@startpars%
8657    \LWR@ensuredoingapar%
8658    \LWR@origitem%
8659 }
```

itemize (*env.*) [⟨*options*⟩]

```
8660 \newcommand*{\LWR@itemizestart}{%
8661    \renewcommand*{\LWR@printcloselist}{\LWR@printcloseitemize}%
8662    \renewcommand*{\LWR@printopenlist}{%
8663     ul style=\textquotedbl\LWR@print@mbox{list-style-type:none}\textquotedbl{}%
8664    }%
8665    \LetLtxMacro\item\LWR@itemizeitem%
8666    \LWR@nulllistfills%
8667 }
```

74.3   Enumerate

An HTML unordered list is used with customized LATEX-generated labels.

enumerate (*env.*) [⟨*options*⟩]

```
8668 \newcommand*{\LWR@enumeratestart}{%
8669    \renewcommand*{\LWR@printcloselist}{\LWR@printcloseitemize}%
8670    \renewcommand*{\LWR@printopenlist}{%
8671     ul style=\textquotedbl\LWR@print@mbox{list-style-type:none}\textquotedbl{}%
8672    }%
8673    \LetLtxMacro\item\LWR@itemizeitem%
8674    \LWR@nulllistfills%
8675 }
```

74.4   Description

\LWR@descitem [⟨*label*⟩] Handles an \item inside a description.

```
8676 \newcommand*{\LWR@descitem}[1][]{%
8677    \LWR@stoppars%
8678    \LWR@setlatestname{#1}%
8679    \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.

```
8680    \begingroup%
8681    \let\LWR@orig@desc@makelabel\makelabel
8682    \renewcommand*{\makelabel}[1]{%
8683       \LWR@htmltag{dt}%
8684       \LWR@orig@desc@makelabel{#1}%
8685       \LWR@htmltag{/dt}%
8686    }
8687    \RenewDocumentCommand{\hspace}{s m}{}%
8688    \LWR@origitem[#1]%
8689    \endgroup%
8690    \LWR@orignewline%
8691    \LWR@htmltag{dd}%
8692    \LWR@startpars%
8693 }
```

description (*env.*) [⟨*options*⟩]

Footnotes are modified to correctly parse optional arguments.

```
8694 \newcommand*{\LWR@descriptionstart}{%
8695    \renewcommand*{\LWR@printcloselist}{\LWR@printclosedescription}
8696    \renewcommand*{\LWR@printopenlist}{dl}
8697    \LetLtxMacro\item\LWR@descitem%
8698    \LWR@nulllistfills%
8699 }
```

74.5 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.

```
8700 \newcommand*{\LWR@patchlists}{%
8701     \LetLtxMacro\item\LWR@listitem%
8702     \LetLtxMacro\@item\LWR@HTML@item%
8703     \renewcommand*{\@trivlist}{%
8704         \LWR@traceinfo{@trivlist start}%
8705         \LWR@liststart%
8706         \LWR@orig@trivlist%
8707         \LWR@traceinfo{@trivlist done}%
8708     }%
8709     \renewcommand*{\trivlist}{%
8710         \LWR@traceinfo{trivlist}%
8711         \LWR@origtrivlist%
8712     }%
8713     \renewcommand*{\endtrivlist}{%
8714         \LWR@traceinfo{endtrivlist start}%
8715         \LWR@origendtrivlist\LWR@listend%
8716         \LWR@traceinfo{endtrivlist done}%
8717     }%
8718     \renewcommand*{\itemize}{%
8719         \LWR@itemizestart\LWR@origitemize%
8720     }%
8721     \renewcommand*{\enumerate}{%
8722         \LWR@enumeratestart\LWR@origenumerate%
8723     }%
8724     \renewcommand*{\description}{%
8725         \LWR@descriptionstart\LWR@origdescription%
8726     }%
8727 }
```

\LWR@restoreoriglists Restores the original trivlist environment.

```
8728 \newcommand*{\LWR@restoreoriglists}{%
8729     \LWR@traceinfo{LWR@restoreoriglists}%
8730     \LetLtxMacro\item\LWR@origitem%
8731     \LetLtxMacro\@item\LWR@orig@item%
8732     \let\@trivlist\LWR@orig@trivlist%
8733     \let\trivlist\LWR@origtrivlist%
8734     \let\endtrivlist\LWR@origendtrivlist%
8735     \LetLtxMacro\itemize\LWR@origitemize%
8736     \LetLtxMacro\enditemize\LWR@endorigitemize%
8737     \LetLtxMacro\enumerate\LWR@origenumerate%
8738     \LetLtxMacro\endenumerate\LWR@endorigenumerate%
8739     \LetLtxMacro\description\LWR@origdescription%
8740     \LetLtxMacro\enddescription\LWR@endorigdescription%
8741     \let\@mklab\LWR@orig@mklab%
8742     \let\makelabel\LWR@origmakelabel%
8743     \let\@donoparitem\LWR@orig@donoparitem%
8744     \let\@nbitem\LWR@orig@nbitem%
8745 }
```

8746 `\end{warpHTML}`

75 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.

75.1 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: 8747 \begin{warpHTML}

75.2   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.

75.2.1   arydshln

Emualated by the original LaTeX non-dashed versions.

```
8748 \LetLtxMacro\hdashline\hline
8749 \LetLtxMacro\cdashline\cline
8750 \LetLtxMacro\firsthdashline\hline
8751 \LetLtxMacro\lasthdashline\hline
```

75.3   Token lookahead

Used by \LWR@futurenonspacelet to look at the next token.

\LWR@mynexttoken

```
8752 \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 paragraphs.

Based on the booktabs style:

```
8753 \def\LWR@futurenonspacelet#1{\def\LWR@cs{#1}%
8754 \afterassignment\LWR@fnslone\let\nexttoken= }
8755
8756 \def\LWR@fnslone{\expandafter\futurelet\LWR@cs\LWR@fnsltwo}
8757
8758 \def\LWR@fnsltwo{%
8759     \expandafter\ifx\LWR@cs\@sptoken%
8760         \let\next=\LWR@fnslthree%
8761     \else%
8762         \expandafter\ifx\LWR@cs\par%
8763             \let\next=\LWR@fnslthree%
8764         \else%
8765             \let\next=\nexttoken%
8766         \fi%
8767     \fi\next}
8768
8769 \def\LWR@fnslthree{\afterassignment\LWR@fnslone\let\next= }
```

\LWR@getmynexttoken   Looks ahead and copies the next token into \LWR@mynexttoken.

```
8770 \newcommand*{\LWR@getmynexttoken}{%
8771     \LWR@traceinfo{LWR@getmynexttoken}%
```

⚠   Nothing must follow this next line:

```
8772     \LWR@futurenonspacelet\LWR@mynexttoken\LWR@tabledatacolumntag
8773 }
```

75.4   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.

8774 \newbool{LWR@startedrow}
8775 \boolfalse{LWR@startedrow}

LWR@tabularcelladded (*bool*)   True if have added a data cell for this position.

8776 \newbool{LWR@tabularcelladded}
8777 \boolfalse{LWR@tabularcelladded}

LWR@hlines (*Ctr*)   Number of \hlines or \midrules above the next row.

8778 \newcounter{LWR@hlines}

LWR@hdashedlines (*Ctr*)   Number of arydshln dashed lines above the next row.

8779 \newcounter{LWR@hdashedlines}

LWR@doingtbrule (*bool*)   True if the next row will have a top/bottom rule above it.

8780 \newbool{LWR@doingtbrule}
8781 \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.

8782 \newbool{LWR@doingcmidrule}
8783 \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.

8784 \newbool{LWR@tableparcell}

LWR@skippingmrowcell (*bool*)   True if are doing an empty \multirow cell, and thus there is no data tag to close.

8785 \newbool{LWR@skippingmrowcell}

LWR@skippingmcolrowcell   True if are doing an empty \multicolumnrow cell, and thus there is no data tag
(*bool*)   to close, and do not print @ and ! columns.

8786 \newbool{LWR@skippingmcolrowcell}

LWR@usedmultirow (*bool*)   Used to error if used \multirow or \multicolumnrow without using \mrowcell or \mcolrowcell.

8787 \newbool{LWR@usedmultirow}

LWR@foundmrowcell (*bool*)   Used to error if used \multirow or \multicolumnrow without using \mrowcell or \mcolrowcell.

8788 \newbool{LWR@foundmrowcell}

LWR@skipatbang (*bool*)   True if just finished a \multicolumn so should not create the trailing @ or ! columns table data cells.

8789 \newbool{LWR@skipatbang}

LWR@emptyatbang (*bool*)   True if finishing a row and should print empty @ or ! column table data cells.

```
8790 \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.

```
8791 \newbool{LWR@intabularmetadata}
8792 \boolfalse{LWR@intabularmetadata}
```

LWR@exitingtabular (*bool*)   When \end is found, turns off the next opening data tag.

```
8793 \newbool{LWR@exitingtabular}
```

LWR@tabularmutemods (*bool*)   Mutes HTML output for @, !, < and >.

This is used while printing the final row to generate \bottomrules.

```
8794 \newbool{LWR@tabularmutemods}
```

LWR@tabularfinalrow (*bool*)   Used to set aria-hidden if adding a final row for the purpose of adding the bottom border.

```
8795 \newbool{LWR@tabularfinalrow}
```

LWR@validtablecol (*bool*)   True if found a valid table column type.

```
8796 \newbool{LWR@validtablecol}
```

LWR@opttablecol (*bool*)   True if found a table column optional argument.

```
8797 \newbool{LWR@opttablecol}
```

Used to add a style to a table data cell:

```
8798 \newbool{LWR@tdhavecellstyle}
```

LWR@tabularDepth (*Ctr*)   Tracks whether & is being used inside a tabular.

```
8799 \newcounter{LWR@tabulardepth}
8800 \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.

```
8801 \newcounter{LWR@tabularpardepth}
8802 \setcounter{LWR@tabularpardepth}{0}
```

```
8803 \newcommand*{\LWR@colsresult}{}%temp storage for column format results
8804 \newcommand*{\LWR@pposition}{}
8805 \newcommand*{\LWR@pleft}{}
8806 \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.

```
8807 \providecommand*{\LWR@strresult}{}
8808 \providecommand*{\LWR@strresulttwo}{}
```

\LWR@origcolspec   Holds the original column specs given to tabular.

```
8809 \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.)

```
8810 \newcounter{LWR@tablecolspecwidth}
```

LWR@tablecolspecindex (*Ctr*)   While parsing the LaTeX table column specification, starts at 1 and is incremented per token of the specification.

```
8811 \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.

```
8812 \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.)

```
8813 \newcounter{LWR@tabletotalLaTeXcols}
```

LWR@tabletotalLaTeXcolsnext (*Ctr*)   Holds the next LaTeX table column index while parsing, equal to one more than LWR@tabletotalLaTeXcols.

```
8814 \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 42.

LWR@colbangspec   A data array of specifications for ! columns. The leftmost's index is leftedge, the others are counter values. See section 42.

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.

```
8815 \newcounter{LWR@cellcolordepth}
```

75.4.1 Multicolumn variables

```
8816 \newcounter{LWR@tablemulticolswidth}
```

Indexes into the multicolumn specification:

```
8817 \newcounter{LWR@tablemulticolspos}
```

Remembers multicolumn vertical rules if found in the column spec.

```
8818 \newcounter{LWR@mcolvertbarsl}
8819 \newcounter{LWR@mcolvertbarsr}
8820 \newcounter{LWR@mcolvertbarsldash}
8821 \newcounter{LWR@mcolvertbarsrdash}
8822 \newbool{LWR@mcolvertbaronleft}
```

75.4.2 Longtable variables

LWR@starredlongtable (*bool*)   Per the caption package, step the counter if longtable*.

```
8823 \newbool{LWR@starredlongtable}
8824 \boolfalse{LWR@starredlongtable}
```

75.4.3 Midrule variables

LWR@midrulecounter (*Ctr*)   Indexes across the LWR@midrules and LWR@trim<l/r>rules data arrays.

```
8825 \newcounter{LWR@midrulecounter}
```

75.5 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/)

\LWR@insertatbangcols

```
8826 \newcommand*{\LWR@insertatbangcols}{%
8827     \ifbool{LWR@skipatbang}%
8828     {}%
8829     {%
8830         \LWR@printatbang{at}{\arabic{LWR@tableLaTeXcolindex}}%
8831         \LWR@printatbang{bang}{\arabic{LWR@tableLaTeXcolindex}}%
8832     }%
8833 }
```

\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.

```
8834 \newcommand*{\LWR@closetabledatacell}{%
8835    \booltrue{LWR@intabularmetadata}%
8836    \ifbool{LWR@exitingtabular}%
8837    {%

8838       \LWR@stoppars%
8839    }%
8840    {% not exiting tabular
8841     \ifboolexpr{bool{LWR@skippingmrowcell} or bool{LWR@skippingmcolrowcell}}%
8842       {%

8843          \LWR@stoppars%
```

If not skipping a \multicolumnrow cell, insert the @ and ! columns after this non-existant column.

```
8844          \ifbool{LWR@skippingmcolrowcell}%
8845             {}%
8846             {\LWR@insertatbangcols}%
8847       }%
8848       {% not skippingmrowcell
```

Insert any < then any @ and ! column contents, unless muted for the \bottomrule or a \multicolumn:

```
8849          \unskip%
8850          \ifboolexpr{%
8851             bool{LWR@tabularmutemods} or
8852             bool{LWR@skipatbang} or
8853             bool{LWR@emptyatbang}
8854          }%
8855             {}%
8856             {%
8857                \LWR@getexparray{LWR@colafterspec}%
8858                   {\arabic{LWR@tableLaTeXcolindex}}%
8859             }%
```

Close paragraphs:

```
8860             \LWR@stoppars%
8861             \boolfalse{LWR@tableparcell}%
```

Close the table data cell.

Close any color <div>s.

```
8862             \whileboolexpr{test {\ifnumcomp{\value{LWR@cellcolordepth}}{>}{0}}}{%
8863                \LWR@htmltag{/div}\LWR@orignewline%
8864                \defaddtocounter{LWR@cellcolordepth}{-1}%
8865             }%
```

Skip the @ and ! cells if are closing a multicolumn cell.

```
8866             \leavevmode\unskip\LWR@htmltag{/td}\LWR@orignewline%
```

```
8867              \global\booltrue{LWR@tabularcelladded}%
8868              \LWR@insertatbangcols%
8869          }% not skipping mrowcell
8870      }% not exiting tabular
8871      \boolfalse{LWR@skippingmrowcell}%
8872      \boolfalse{LWR@skippingmcolrowcell}%
8873      \boolfalse{LWR@skipatbang}%
```

Color control. Column is set by >{} for each cell, so it must be cleared here.

```
8874      \def\LWR@cellHTMLcolor{}%
8875      \def\LWR@columnHTMLcolor{}%
8876      \defcounter{LWR@cellcolordepth}{0}%
8877 }
```

When not used inside a tabular, & performs its original function as recorded here (with catcode 4).

```
8878 \let\LWR@origampmacro&
```

```
8879 \end{warpHTML}
```

75.5.1 Handling &

`8880 \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.

```
8881 \newcommand*{\LWR@tabularampersand}{%
8882     \LWR@traceinfo{LWR@tabularampersand}%
8883     \ifnumcomp{\value{LWR@tabulardepth}}{>}{0}%
8884     {%
```

If not skipping a multirow cell, close the current data cell.

```
8885          \unskip%
8886          \LWR@closetabledatacell%
```

Move to the next column.

```
8887          \defaddtocounter{LWR@tableLaTeXcolindex}{1}%
```

Have not yet added data in this column:

```
8888          \global\boolfalse{LWR@tabularcelladded}%
```

Look at the next token to decide multi or single column data tag.

```
8889          \LWR@getmynexttoken%
8890      }%
```

If not inside a tabular, performs the original action:

```
8891    {%
8892        \LWR@origampmacro%
8893    }%
8894 }
```

& 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.

75.6   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.

```
8895 \newcommand*{\LWR@tabularfinishrow}{%
```

If not exiting the tabular, or doing a rule, or have already started a row, finish this row:

```
8896    \ifboolexpr{%
8897        not bool {LWR@exitingtabular} or%
8898        bool{LWR@doingtbrule} or%
8899        bool{LWR@doingcmidrule} or%
8900        test{\ifnumcomp{\value{LWR@hlines}}{>}{0}} or%
8901        test{\ifnumcomp{\value{LWR@hdashedlines}}{>}{0}} or%
8902        bool{LWR@startedrow}%
8903    }{%
```

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:

```
8904    \ifbool{LWR@exitingtabular}{%
8905        \booltrue{LWR@tabularmutemods}%
8906    }{%
8907        \boolfalse{LWR@tabularmutemods}%
8908    }%
```

Locally reenable the table data tags until finished with the final row:

```
8909    \boolfalse{LWR@exitingtabular}%
```

Generate table data tags and ampersands until the right edge:

```
8910    \whileboolexpr{%
```

```
8911        test {
8912            \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{<}
8913                {\value{LWR@tabletotalLaTeXcols}}
8914        } or %
8915        (%
8916            bool{LWR@intabularmetadata} and%
8917            not bool{LWR@tabularcelladded} and%
8918            test {
8919                \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{=}
8920                    {\value{LWR@tabletotalLaTeXcols}}
8921            }%
8922        )%
8923    }%
8924    {%
8925        \LWR@tabledatasinglecolumntag%
```

The following is essentially \LWR@tabularampersand with LWR@emptyatbang added
to empty the following cells:

```
8926        \LWR@closetabledatacell%
8927        \defaddtocounter{LWR@tableLaTeXcolindex}{1}%
8928        \global\boolfalse{LWR@tabularcelladded}%
8929        \booltrue{LWR@emptyatbang}%
```

Starts the next cell:

```
8930        \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{<}
8931                {\value{LWR@tabletotalLaTeXcols}}%
8932            {\LWR@getmynexttoken}%
8933            {}%
8934    }%
```

Reenable the original LWR@exitingtabular to close the entire table:

```
8935    \ifbool{LWR@tabularmutemods}{%
8936        \booltrue{LWR@exitingtabular}%
8937    }{%
8938        \boolfalse{LWR@exitingtabular}%
8939    }%
8940    \boolfalse{LWR@tabularmutemods}%

8941    \boolfalse{LWR@emptyatbang}%
8942    }{}% ifboolexpr
8943 }
```

75.7 Handling \\

Inside tabular, \\ is redefined to \LWR@tabularendofline

Throws away options \\[dim] or *

\LWR@tabularendofline

```
8944 \NewDocumentCommand{\LWR@tabularendofline}{s o}{%
```

Finish the row:

```
8945    \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{<}
8946            {\value{LWR@tabletotalLaTeXcols}}%
8947        {\LWR@tabularfinishrow}%
8948        {\LWR@closetabledatacell}%
8949    \LWR@htmltag{/tr}\LWR@orignewline%
```

xcolor row color support:

```
8950    \@rowc@lors%
```

No longer inside a data cell:

```
8951    \booltrue{LWR@intabularmetadata}%
```

Not yet started a table row:

```
8952    \boolfalse{LWR@startedrow}%
```

Additional setup:

```
8953    \defcounter{LWR@hlines}{0}%
8954    \defcounter{LWR@hdashedlines}{0}%
8955    \boolfalse{LWR@doingtbrule}%
8956    \boolfalse{LWR@doingcmidrule}%
8957    \LWR@clearmidrules%

8958    \def\LWR@rowHTMLcolor{}%
```

Start at first column:

```
8959    \defcounter{LWR@tableLaTeXcolindex}{1}%
```

Have not yet added data in this column:

```
8960    \global\boolfalse{LWR@tabularcelladded}%
```

Allow TeX to flush the pending paragraph. Not doing so causes a slowdown for very large tables.

```
8961    \LWR@stoppars%
8962    \LWR@forceemptyline%
```

Look at the next token to decide between single column data tag or a special case:

```
8963    \LWR@getmynexttoken%
8964 }
```

75.8 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}.

```
8965 \newcommand*{\LWR@columnspeclookahead}[1]{%
8966     \setcounter{LWR@tempcountone}{\value{LWR@tablecolspecindex}}%
8967     \addtocounter{LWR@tempcountone}{#1}%
8968     \fullexpandarg%
8969   \StrChar{\LWR@origcolspec}{\arabic{LWR@tempcountone}}[\LWR@strresulttwo]%
```

Get the contents of the first group in \LWR@strresulttwo:

```
8970        \exploregroups%
8971        \StrChar{\LWR@strresulttwo}{1}[\LWR@strresulttwo]%
8972        \noexploregroups%
8973 }
```

75.9   Parsing @, >, <, !, bar columns

Holds the parsed argument for @, >, <, or ! columns:

```
8974 \newcommand*{\LWR@colparameter}{}
```

\LWR@parseatcolumn {⟨*this column type*⟩}

Handles @{text} columns.

The argument is ignored, but provided for compatibility with \LWR@parsenormalcolumn.

```
8975 \newcommand*{\LWR@parseatcolumn}[1]{%
```

Move to the next token after the '@':

```
8976     \LWR@traceinfo{at column}%
8977     \defaddtocounter{LWR@tablecolspecindex}{1}%
```

Read the next token into \LWR@colparameter, expanding once:

```
8978     \LWR@traceinfo{about to read the next token:}%
8979     \expandarg%
8980     \StrChar{\LWR@origcolspec}%
8981        {\arabic{LWR@tablecolspecindex}}[\LWR@colparameter]%
8982     \fullexpandarg%
```

Store the result into a data array, expanding once out of \LWR@colparameter:

```
8983     \LWR@traceinfo{have now read the next token}%
8984     \ifnumcomp{\value{LWR@tabletotalLaTeXcols}}{=}{0}%
8985     {% left edge of the table:
8986        \LWR@traceinfo{at the left edge}%
8987        \LWR@setexparray{LWR@colatspec}%
8988            {leftedge}%
8989            {\expandafter\@firstofone\LWR@colparameter}%
8990        \LWR@traceinfo{at the left edge: %
8991            \LWR@getexparray{LWR@colatspec}{leftedge}}%
8992     }%
8993     {% not at the left edge:
8994        \LWR@traceinfo{not at the left edge}%
8995        \LWR@setexparray{LWR@colatspec}%
8996            {\arabic{LWR@tabletotalLaTeXcols}}%
```

```
8997                {\expandafter\@firstofone\LWR@colparameter}%
8998            \LWR@traceinfo{at \arabic{LWR@tabletotalLaTeXcols}%
8999            : % space
9000            \LWR@getexparray{LWR@colatspec}{\arabic{LWR@tabletotalLaTeXcols}}}%
9001        }%
9002        \let\LWR@colparameter\relax%
9003        \booltrue{LWR@validtablecol}%
9004 }
```

\LWR@parsebangcolumn {⟨*this column type*⟩} Handles !{text} columns.

The argument is ignored, but provided for compatibility with \LWR@parsenormalcolumn.

```
9005 \newcommand*{\LWR@parsebangcolumn}[1]{%
```

Move to the next token after the '!':

```
9006        \LWR@traceinfo{bang column}%
9007        \defaddtocounter{LWR@tablecolspecindex}{1}%
```

Read the next token into \LWR@colparameter, expanding once:

```
9008        \LWR@traceinfo{about to read the next token:}%
9009        \expandarg%
9010        \StrChar{\LWR@origcolspec}%
9011            {\arabic{LWR@tablecolspecindex}}[\LWR@colparameter]%
9012        \fullexpandarg%
```

Store the result into a data array, expanding once out of \LWR@colparameter:

```
9013        \LWR@traceinfo{have now read the next token}%
9014        \ifnumcomp{\value{LWR@tabletotalLaTeXcols}}{=}{0}%
9015        {% left edge of the table:
9016            \LWR@traceinfo{at the left edge}%
9017            \LWR@setexparray{LWR@colbangspec}%
9018                {leftedge}%
9019                {\expandafter\@firstofone\LWR@colparameter}%
9020        }%
9021        {% not at the left edge:
9022            \LWR@traceinfo{not at the left edge}%
9023            \LWR@setexparray{LWR@colbangspec}%
9024                {\arabic{LWR@tabletotalLaTeXcols}}%
9025                {\expandafter\@firstofone\LWR@colparameter}%
9026          \LWR@traceinfo{bang \arabic{LWR@tabletotalLaTeXcols}: \LWR@colparameter!}%
9027        }%
9028        \let\LWR@colparameter\relax%
9029        \booltrue{LWR@validtablecol}%
9030 }
```

\LWR@checkbeforeaddclass {⟨*compared csname*⟩} {⟨*css class to add*⟩}

```
9031 \newcommand*{\LWR@checkbeforeaddclass}[2]{%
9032    \ifcsstrequal{LWR@tempone}{#1}%
9033        {%
9034            \LWR@setexparray{LWR@coladdclass}%
9035                {\arabic{LWR@tabletotalLaTeXcolsnext}}%
9036                { #2}% space is intentional
9037        }{}%
```

```
9038 }
```

\LWR@checkmathcolpar Error if using math in column parameters.

```
9039 \newcommand*{\LWR@checkmathcolpar}{%
9040     \IfSubStr{\detokenize\expandafter{\LWR@colparameter}}{\LWRdollar}%
9041        {%
9042            \PackageError{lwarp}%
9043               {%
9044               Lwarp does not support '$' in column specifiers.\MessageBreak
9045                 Specify '$' math for each cell in the column.\MessageBreak
9046                   Enter 'h' for more info%
9047               }%
9048               {%
9049              For example, replace '>{$}c<{$}' with 'c', and then\MessageBreak
9050                   use '$cell contents$' for each cell in the column.%
9051               }%
9052        }{}%
9053 }
```

\LWR@parsebeforecolumn {⟨*this column type*⟩}

Handles >{text} columns.

The argument is ignored, but provided for compatibility with \LWR@parsenormalcolumn.

```
9054 \newcommand*{\LWR@parsebeforecolumn}[1]{%
```

Move to the next token after the '>':

```
9055     \defaddtocounter{LWR@tablecolspecindex}{1}%
```

Read the next token, expanding once into \LWR@colparameter:

```
9056     \expandarg%
9057     \StrChar{\LWR@origcolspec}%
9058         {\arabic{LWR@tablecolspecindex}}[\LWR@colparameter]%
9059     \fullexpandarg%
```

Error if using >{$}, which is not supported by lwarp.

```
9060     \LWR@checkmathcolpar%
```

Store the result into a data array, expanding once out of \LWR@colparameter:

```
9061     \LWR@setexparray{LWR@colbeforespec}%
9062         {\arabic{LWR@tabletotalLaTeXcolsnext}}%
9063         {\expandafter\@firstofone\LWR@colparameter}%
9064 %
9065     \edef\LWR@tempone{\expandafter\@firstofone\LWR@colparameter}%
```

If detect >{\centering\arraybackslash} or related, add a css class.

```
9066     \LWR@checkbeforeaddclass{LWR@detect@centeringarraybackslash}{tdcenter}
9067     \LWR@checkbeforeaddclass{LWR@detect@raggedrightarraybackslash}{tdleft}
9068     \LWR@checkbeforeaddclass{LWR@detect@raggedleftarraybackslash}{tdright}
9069     \LWR@checkbeforeaddclass{LWR@detect@itshape}{tditshape}
```

```
9070        \LWR@checkbeforeaddclass{LWR@detect@bfseries}{tdbfseries}
9071        \LWR@checkbeforeaddclass{LWR@detect@bfit}{tdbfit}

9072        \let\LWR@colparameter\relax%
9073        \booltrue{LWR@validtablecol}%
9074 }
```

\LWR@parseaftercolumn {⟨*this column type*⟩}

        Handles <{text} columns.

        The argument is ignored, but provided for compatibility with \LWR@parsenormalcolumn.

```
9075 \newcommand*{\LWR@parseaftercolumn}[1]{%
```

        Move to the next token after the '<':

```
9076        \defaddtocounter{LWR@tablecolspecindex}{1}%
```

        Read the next token, expanding once into \LWR@colparameter:

```
9077        \expandarg%
9078        \StrChar{\LWR@origcolspec}%
9079            {\arabic{LWR@tablecolspecindex}}[\LWR@colparameter]%
9080        \fullexpandarg%
```

        Error if using >{$}, which is not supported by lwarp.

```
9081        \LWR@checkmathcolpar%
```

        Store the result into a data array, expanding once out of \LWR@colparameter:

```
9082        \LWR@setexparray{LWR@colafterspec}%
9083            {\arabic{LWR@tabletotalLaTeXcols}}%
9084            {\expandafter\@firstofone\LWR@colparameter}%
9085        \let\LWR@colparameter\relax%
9086        \booltrue{LWR@validtablecol}%
9087 }
```

\LWR@parsebarcolumn {⟨*this column type*⟩}

        Handles vertical rules.

        The argument is ignored, but provided for compatibility with \LWR@parsenormalcolumn.

```
9088 \newcommand*{\LWR@parsebarcolumn}[1]{%
9089        \LWR@traceinfo{\LWR@parsebarcolumn}%
```

        Remember the bar at this position:

```
9090        \ifnumcomp{\value{LWR@tabletotalLaTeXcols}}{=}{0}%
9091        {% left edge of the table:
9092            \edef\LWR@tempone{\LWR@getexparray{LWR@colbarspec}{leftedge}}%
9093            \ifdefstring{\LWR@tempone}{tvertbarl}%
9094            {\LWR@setexparray{LWR@colbarspec}{leftedge}{tvertbarldouble}}%
9095            {\LWR@setexparray{LWR@colbarspec}{leftedge}{tvertbarl}}%
9096        }%
```

```
9097        {% not at the left edge:
9098            \edef\LWR@tempone{%
9099             \LWR@getexparray{LWR@colbarspec}{\arabic{LWR@tabletotalLaTeXcols}}%
9100            }%
9101            \ifdefstring{\LWR@tempone}{tvertbarr}%
9102            {%
9103                \LWR@setexparray{LWR@colbarspec}%
9104                    {\arabic{LWR@tabletotalLaTeXcols}}{tvertbarrdouble}%
9105            }%
9106            {%
9107                \LWR@setexparray{LWR@colbarspec}%
9108                    {\arabic{LWR@tabletotalLaTeXcols}}{tvertbarr}%
9109            }%
9110        }%
9111        \booltrue{LWR@validtablecol}%
9112 }
```

\LWR@parsecoloncolumn {⟨*this column type*⟩}

> Handles vertical rules.

> The argument is ignored, but provided for compatibility with \LWR@parsenormalcolumn.

```
9113 \newcommand*{\LWR@parsecoloncolumn}[1]{%
9114     \LWR@traceinfo{LWR@parsecoloncolumn}%
```

> Remember the bar at this position:

```
9115     \ifnumcomp{\value{LWR@tabletotalLaTeXcols}}{=}{0}%
9116     {% left edge of the table:
9117        \edef\LWR@tempone{\LWR@getexparray{LWR@colbarspec}{leftedge}}%
9118        \ifdefstring{\LWR@tempone}{tvertbarldash}%
9119        {\LWR@setexparray{LWR@colbarspec}{leftedge}{tvertbarldoubledash}}%
9120        {\LWR@setexparray{LWR@colbarspec}{leftedge}{tvertbarldash}}%
9121     }%
9122     {% not at the left edge:
9123        \edef\LWR@tempone{%
9124         \LWR@getexparray{LWR@colbarspec}{\arabic{LWR@tabletotalLaTeXcols}}%
9125        }%
9126        \ifdefstring{\LWR@tempone}{tvertbarrdash}%
9127        {\LWR@setexparray{LWR@colbarspec}%
9128            {\arabic{LWR@tabletotalLaTeXcols}}{tvertbarrdoubledash}}%
9129        {\LWR@setexparray{LWR@colbarspec}%
9130            {\arabic{LWR@tabletotalLaTeXcols}}{tvertbarrdash}}%
9131     }%
9132     \booltrue{LWR@validtablecol}%
9133 }
```

\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.

```
9134 \newcommand*{\LWR@parsesemicoloncolumn}[1]{%
```

Treat ; as a : column:

```
9135    \LWR@parsecoloncolumn{}%
9136 }
```

75.10 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.

```
9137 \newcommand*{\LWR@HTML@LWR@parsenormalcolumn}[1]{%
9138    \defaddtocounter{LWR@tabletotalLaTeXcols}{1}%
9139    \defaddtocounter{LWR@tabletotalLaTeXcolsnext}{1}%

9140    \LWR@setexparray{LWR@tablecolspec}{\arabic{LWR@tabletotalLaTeXcols}}{#1}%

9141    \LWR@traceinfo{normal column \arabic{LWR@tabletotalLaTeXcols}: #1}%
9142    \LWR@setexparray{LWR@colatspec}{\arabic{LWR@tabletotalLaTeXcolsnext}}{}%
9143    \LWR@setexparray{LWR@colbangspec}{\arabic{LWR@tabletotalLaTeXcolsnext}}{}%
9144    \LWR@setexparray{LWR@colbeforespec}{\arabic{LWR@tabletotalLaTeXcolsnext}}{}%
9145    \LWR@setexparray{LWR@colafterspec}{\arabic{LWR@tabletotalLaTeXcolsnext}}{}%
9146    \LWR@setexparray{LWR@colbarspec}{\arabic{LWR@tabletotalLaTeXcolsnext}}{}%
9147    \LWR@setexparray{LWR@coladdclass}{\arabic{LWR@tabletotalLaTeXcolsnext}}{}%
9148    \booltrue{LWR@validtablecol}%
9149 }
9150
9151 \newcommand*{\LWR@print@LWR@parsenormalcolumn}[1]{}
9152
9153 \LWR@formatted{LWR@parsenormalcolumn}
```

75.11 Parsing 'w' columns

W and w columns are handled via array with \HTMLnewcolumntype.

75.12 Parsing '*' columns

\LWR@parsestarcolumn {⟨*this column type*⟩} Star columns should already have been expanded, so this should never be used.

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.

```
9154 \newcommand*{\LWR@parsestarcolumn}[1]{}
```

Table 13: Tabular baseline

l	p	m	b	r
			bot	
		mid	bot	
l	par	mid	bot	r
	par	mid		
	par			

75.13 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}

```
9155 \newcommand*{\LWR@expandpreamble}[1]{%
9156     \edef\@tempa{\@temptokena={#1}}%
9157     \@tempa%
9158     \@tempswatrue%
9159     \@whilesw\if@tempswa\fi{%
9160         \@tempswafalse\the\NC@list%
9161     }%
9162 }
```

75.14 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*⟩}

Add HTML functionality to an existing print version column type.

```
9163 \newcommand*{\LWR@modifycolumntype}[5]{%
9164     \LWR@traceinfo{LWR@modifycolumntype !#1!#2!#3!#4!#5!}%
9165         \LWR@traceinfo{LWR@modifycolumntype #1}%
9166         \edef\@tempa{%
9167             \noexpand\csdef{LWR@columntype@#1}{%
9168                 \noexpand\@nameuse{#3}{#1}%
9169                 \noexpand\defaddtocounter{LWR@tablecolspecindex}{#2}%
```

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.

```
9170                }%
9171                \noexpand\csdef{LWR@columntype@mctype@#1}{%
9172                    \noexpand\@nameuse{#4}{#1}%
9173                }%
9174                \noexpand\csdef{LWR@columntype@mcdata@#1}{%
9175                    \noexpand\@nameuse{#5}{#2}%
9176                }%
9177            }%
9178            \@tempa%
9179        \LWR@traceinfo{LWR@modifycolumntype done}%
9180 }

9181 \LWR@modifycolumntype{l}{0}{LWR@parsenormalcolumn}
9182     {LWR@printmccoltype@normal}{LWR@printmccoldata@normal}
9183
9184 \LWR@modifycolumntype{c}{0}{LWR@parsenormalcolumn}
9185     {LWR@printmccoltype@normal}{LWR@printmccoldata@normal}
9186
9187 \LWR@modifycolumntype{r}{0}{LWR@parsenormalcolumn}
9188     {LWR@printmccoltype@normal}{LWR@printmccoldata@normal}


9189 \LWR@modifycolumntype{@}{0}{LWR@parseatcolumn}
9190     {LWR@printmccoltype@ignore}{LWR@printmccoldata@other}
9191
9192 \LWR@modifycolumntype{!}{0}{LWR@parsebangcolumn}
9193     {LWR@printmccoltype@ignore}{LWR@printmccoldata@other}
9194
9195 \LWR@modifycolumntype{>}{0}{LWR@parsebeforecolumn}
```

```
9196        {LWR@printmccoltype@ignore}{LWR@printmccoldata@other}
9197
9198 \LWR@modifycolumntype{<}{0}{LWR@parseaftercolumn}
9199        {LWR@printmccoltype@ignore}{LWR@printmccoldata@other}
9200
9201 \LWR@modifycolumntype{|}{0}{LWR@parsebarcolumn}
9202        {LWR@printmccoltype@vertbar}{LWR@printmccoldata@skip}
9203
9204 \LWR@modifycolumntype{:}{0}{LWR@parsecoloncolumn}
9205        {LWR@printmccoltype@colon}{LWR@printmccoldata@skip}
9206
9207 \LWR@modifycolumntype{;}{1}{LWR@parsesemicoloncolumn}
9208        {LWR@printmccoltype@semicolon}{LWR@printmccoldata@skip}

9209 \LWR@modifycolumntype{p}{1}{LWR@parsenormalcolumn}
9210        {LWR@printmccoltype@normal}{LWR@printmccoldata@paragraph}
9211
9212 \LWR@modifycolumntype{m}{1}{LWR@parsenormalcolumn}
9213        {LWR@printmccoltype@normal}{LWR@printmccoldata@paragraph}
9214
9215 \LWR@modifycolumntype{b}{1}{LWR@parsenormalcolumn}
9216        {LWR@printmccoltype@normal}{LWR@printmccoldata@paragraph}
```

A star column:

```
9217 \LWR@modifycolumntype{*}{2}{LWR@parsestarcolumn}
9218        {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>}
```

```
9219 \NewDocumentCommand{\HTMLnewcolumntype}{m O{0} o m}{%
9220     \IfValueTF{#3}
9221     {
9222         \expandafter\newcommand\expandafter*%
9223             \csname LWR@HTML@NC@rewrite@#1\endcsname[#2][#3]{\NC@find #4}%
9224         \LWR@formatted{NC@rewrite@#1}%
9225     }
9226     {
9227         \expandafter\newcommand\expandafter*%
9228             \csname LWR@HTML@NC@rewrite@#1\endcsname[#2]{\NC@find #4}%
9229         \LWR@formatted{NC@rewrite@#1}%
9230     }
9231 }

9232 \end{warpHTML}
```

for PRINT output: `9233 \begin{warpprint}`

9234 \NewDocumentCommand{\HTMLnewcolumntype}{m O{0} o m}{}

9235 \end{warpprint}

9236 \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.

9237 \newcommand*{\LWR@parsetablecols}[1]{%
9238   \LWR@traceinfo{LWR@parsetablecols}%

Remember the original supplied column spec:

9239   \renewcommand*{\LWR@origcolspec}{#1}%

Remove spaces:

9240   \expandarg%
9241   \StrSubstitute{\LWR@origcolspec}{ }{}[\LWR@origcolspec]%

Expand any star columns:

9242   \LWR@expandpreamble{\LWR@origcolspec}%
9243   \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:

9244   \defcounter{LWR@tabletotalLaTeXcols}{0}%
9245   \defcounter{LWR@tabletotalLaTeXcolsnext}{1}%
9246   \LWR@setexparray{LWR@colatspec}{leftedge}{}%
9247   \LWR@setexparray{LWR@colatspec}{1}{}%
9248   \LWR@setexparray{LWR@colatspec}{2}{}%
9249   \LWR@setexparray{LWR@colatspec}{3}{}%
9250   \LWR@setexparray{LWR@colbangspec}{leftedge}{}%
9251   \LWR@setexparray{LWR@colbangspec}{1}{}%
9252   \LWR@setexparray{LWR@colbangspec}{2}{}%
9253   \LWR@setexparray{LWR@colbangspec}{3}{}%
9254   \LWR@setexparray{LWR@colbeforespec}{1}{}%
9255   \LWR@setexparray{LWR@colbeforespec}{2}{}%
9256   \LWR@setexparray{LWR@colbeforespec}{3}{}%
9257   \LWR@setexparray{LWR@colafterspec}{1}{}%
9258   \LWR@setexparray{LWR@colafterspec}{2}{}%
9259   \LWR@setexparray{LWR@colafterspec}{3}{}%
9260   \LWR@setexparray{LWR@colbarspec}{leftedge}{}%
9261   \LWR@setexparray{LWR@colbarspec}{1}{}%
9262   \LWR@setexparray{LWR@colbarspec}{2}{}%

```
9263    \LWR@setexparray{LWR@colbarspec}{3}{}%
9264    \LWR@setexparray{LWR@coladdclass}{1}{}%
9265    \LWR@setexparray{LWR@coladdclass}{2}{}%
9266    \LWR@setexparray{LWR@coladdclass}{3}{}%
```

Starting at the first column specification:

```
9267    \defcounter{LWR@tablecolspecindex}{1}%
```

Place the colspecs string length into \LWR@strresult, and remember the number of characters in the column specification:

```
9268    \expandarg%
9269    \StrLen{\LWR@origcolspec}[\LWR@strresult]%
9270    \fullexpandarg%
9271    \LWR@traceinfo{original column spec length: \LWR@strresult}%
9272    \defcounter{LWR@tablecolspecwidth}{\LWR@strresult}%
```

Haven't seen any optional arguments so far

```
9273    \boolfalse{LWR@opttablecol}%
```

Scan through the column specifications:

```
9274    \whileboolexpr{%
9275        not test{%
9276            \ifnumcomp{\value{LWR@tablecolspecindex}}{>}%
9277                {\value{LWR@tablecolspecwidth}}%
9278        }%
9279    }%
9280    {%
```

Place the next single-character column type into \LWR@strresult:

```
9281    \expandarg%
9282    \StrChar{\LWR@origcolspec}{\arabic{LWR@tablecolspecindex}}[\LWR@strresult]%
9283    \LWR@traceinfo{position \arabic{LWR@tablecolspecindex}: \LWR@strresult}%
9284    \fullexpandarg%
```

Not yet found a valid column type:

```
9285    \boolfalse{LWR@validtablecol}%
```

Skip over any optional arguments, such as siunitx S column:

```
9286    \IfStrEq{\LWR@strresult}{[}{\booltrue{LWR@opttablecol}}{}%
```

Throw away anything found inside the optional argument:

```
9287    \ifbool{LWR@opttablecol}%
9288    {}% inside an optional argument
9289    {% not an optional tabular argument
```

Not inside an optional argument, so consider the column type:

```
9290        \ifcsdef{LWR@columntype@\LWR@strresult}%
9291            {\csuse{LWR@columntype@\LWR@strresult}}%
9292            {}%
```

If an unknown column type, use l:

```
9293        \ifbool{LWR@validtablecol}{}{%
9294            \LWR@traceinfo{invalid column type: \LWR@strresult}%
9295            \LWR@parsenormalcolumn{l}%
9296        }%
9297    }% not an optional column argument
```

If read the closing bracket, no longer inside the optional argument:

```
9298    \IfStrEq{\LWR@strresult}{]}{\boolfalse{LWR@opttablecol}}{}%
```

Move to the next character:

```
9299    \defaddtocounter{LWR@tablecolspecindex}{1}%
9300    }% whiledo
9301 }%
```

75.15   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.

```
9302 \@ifundefined{rownum}{\newcount\rownum}{}
```

\@rowcolors Emulated in case xcolor is not used.

```
9303 \newcommand*{\@rowcolors}{}
```

\@rowc@lors Emulated in case xcolor is not used.

```
9304 \newcommand*{\@rowc@lors}{}
```

\LWR@xcolorrowHTMLcolor Emulated xcolor row color.

```
9305 \newcommand*{\LWR@xcolorrowHTMLcolor}{}
```

\LWR@columnHTMLcolor HTMLstyle code for the column color.

```
9306 \def\LWR@columnHTMLcolor{}
```

\LWR@rowHTMLcolor HTMLstyle code for the row color.

```
9307 \def\LWR@rowHTMLcolor{}
```

\LWR@cellHTMLcolor HTMLstyle code for the cell color.

9308 \def\LWR@cellHTMLcolor{}

\LWR@ruleHTMLcolor HTMLstyle code for the rule color.

9309 \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.

9310 \newcommand*{\rowcolor}{\LWR@getmynexttoken}%

\arrayrulecolor [⟨*model*⟩] {⟨*color*⟩}

\arrayrulecolornexttoken [⟨*model*⟩] {⟨*color*⟩}

Print versions for use outside and inside a tabular:

9311 \newcommand{\arrayrulecolor}[2][named]{}
9312 \newcommand{\arrayrulecolornexttoken}[2][named]{\LWR@getmynexttoken}

\doublerulesepcolor [⟨*model*⟩] {⟨*color*⟩}

\doublerulesepcolornexttoken [⟨*model*⟩] {⟨*color*⟩}

Print versions for use inside and outside a tabular:

9313 \newcommand{\doublerulesepcolor}[2][named]{}
9314 \newcommand{\doublerulesepcolornexttoken}[2][named]{\LWR@getmynexttoken}

75.16   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.

9315 \newcommand*{\LWR@maybenewtablerow}
9316 {%
9317   \ifbool{LWR@startedrow}%
9318   {}% started the row
9319   {% 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:

9320   \ifbool{LWR@tabularfinalrow}%
9321   {%
9322   \renewcommand*{\LWR@tempone}%
9323   { aria-hidden=\textquotedbl{}true\textquotedbl}%
9324   }%
9325   {%
9326   \renewcommand*{\LWR@tempone}{}%
9327   }%

Start a new row if doing \hline:

```
9328        \ifboolexpr{%
9329            test{\ifnumcomp{\value{LWR@hlines}}{>}{0}} or%
9330            test{\ifnumcomp{\value{LWR@hdashedlines}}{>}{0}}%
9331        }%
9332        {%
9333            \LWR@htmltag{%
9334                tr %
9335                class=\textquotedbl{}hline\textquotedbl%
9336                \LWR@tempone% aria-hidden
9337            }%
9338            \LWR@orignewline%
```

Remember that now have started the row, and create the row tag, with a class if necessary.

```
9339                \booltrue{LWR@startedrow}%
9340                \booltrue{LWR@intabularmetadata}%
9341        }%
```

If not doing \hline, start a row if doing a top or bottom rule:

```
9342        {% not doing hline
9343            \ifbool{LWR@doingtbrule}%
9344            {%
9345                \ifdefvoid{\LWR@ruleHTMLcolor}{%
9346                    \LWR@htmltag{%
9347                        tr %
9348                        class=\textquotedbl{}tbrule\textquotedbl%
9349                        \LWR@tempone% aria-hidden
9350                    }%
9351                }{%
9352                    \LWR@htmltag{%
9353                        tr class=\textquotedbl{}tbrule\textquotedbl\ % space
9354                        style=\textquotedbl{}border-top: 1px solid % space
9355                            \LWR@origpound\LWR@ruleHTMLcolor \textquotedbl{}%
9356                        \LWR@tempone% aria-hidden
9357                    }%
9358                }%
9359                \LWR@orignewline%
```

Remember that now have started the row, and create the row tag, with a class if necessary.

```
9360                \booltrue{LWR@startedrow}%
9361                \booltrue{LWR@intabularmetadata}%
9362            }%
9363            {%
```

If not the final row, start a new row:

```
9364                \ifbool{LWR@tabularfinalrow}%
9365                    {}%
9366                    {%
9367                        \LWR@htmltag{tr}\LWR@orignewline%
```

Remember that now have started the row, and create the row tag, with a class if necessary.

```
9368                              \booltrue{LWR@startedrow}%
9369                              \booltrue{LWR@intabularmetadata}%
9370                          }%
9371              }%
9372         }% end of not doing hline
9373     }% end of not started the row
9374 }
```

75.17   Printing vertical bar tags

\LWR@printbartag {⟨*index*⟩}

Adds to a tabular data cell an HTML class name for a left/right vertical bar.

```
9375 \newcommand*{\LWR@printbartag}[1]{%
9376     \LWR@traceinfo{LWR@printbartag !#1!}%
9377     \ifboolexpr{bool{LWR@tabularmutemods} or bool{LWR@emptyatbang}}%
9378     {}% muting or empty
9379     {% not muting
9380         \edef\LWR@tempone{\LWR@getexparray{LWR@colbarspec}{#1}}%
9381         \ifdefempty{\LWR@tempone}{}{ \LWR@tempone}%
9382     }% not muting
9383     \LWR@traceinfo{LWR@printbartag done}%
9384 }
```

75.18   Printing @ or ! tags

\LWR@printatbang {⟨*at — or — bang*⟩} {⟨*index*⟩}

```
9385 \newcommand*{\LWR@printatbang}[2]{%
```

Fetch the column at or bang spec:

```
9386     \xdef\LWR@atbangspec{\LWR@getexparray{LWR@col#1spec}{#2}}%
9387     \LWR@traceinfo{atbang: #2 !\LWR@atbangspec!}%
```

Only generate if is not empty;

```
9388     \ifdefempty{\LWR@atbangspec}%
9389     {}%
9390     {% not empty
9391         \LWR@htmltag{%
9392             td class=\textquotedbl{}td#1%
9393             \LWR@subaddcmidruletrim{}{}%
9394             \LWR@printbartag{#2}%
9395             \textquotedbl{}%
9396             \LWR@tdstartstyles%
9397             \LWR@addcmidrulewidth%
9398             \LWR@addcdashline%
9399             \LWR@addtabularrulecolors%
9400             \LWR@tdendstyles%
9401         }%
```

Create an empty cell if muting for the \bottomrule:

```
9402        \ifboolexpr{bool{LWR@tabularmutemods} or bool{LWR@emptyatbang}}%
9403            {}%
9404            {\LWR@atbangspec}%
9405 %
9406        \LWR@htmltag{/td}\LWR@orignewline%
9407        \global\booltrue{LWR@tabularcelladded}%
9408    }% not empty
9409 }%
```

\LWR@addleftmostbartag

```
9410 \newcommand*{\LWR@addleftmostbartag}{%
9411    \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{=}{1}{%
9412        \LWR@printbartag{leftedge}%
9413    }{}%
9414 }
```

\LWR@tabularleftedge

```
9415 \newcommand*{\LWR@tabularleftedge}{%
9416    \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{=}{1}%
9417    {%
9418        \LWR@printatbang{at}{leftedge}%
9419        \LWR@printatbang{bang}{leftedge}%
9420    }% left edge
9421    {}% not left edge
9422 }
```

75.19   Cell opening tag

\LWR@thiscolspec   Temporary storage.

```
9423 \newcommand*{\LWR@thiscolspec}{}
```

\LWR@tabledatasinglecolumntag   Print a table data opening tag with style for alignment and color.

```
9424 \newcommand*{\LWR@tabledatasinglecolumntag}%
9425 {%
9426    \LWR@traceinfo{LWR@tabledatasinglecolumntag}%
9427    \LWR@maybenewtablerow%
```

Don't start a new paragraph tag if have already started one:

```
9428    \ifbool{LWR@intabularmetadata}%
9429    {%
```

If have found the end of tabular command, do not create the next data cell:

```
9430        \ifbool{LWR@exitingtabular}{}%
9431        {% not exiting tabular
```

Print the @ and ! contents before first column:

```
9432            \LWR@tabularleftedge%
```

Fetch the current column's alignment character into \LWR@strresult:

```
9433                \xdef\LWR@strresult{%
9434              \LWR@getexparray{LWR@tablecolspec}{\arabic{LWR@tableLaTeXcolindex}}}%
9435                }%
```

Print the start of a new table data cell:

```
9436              \LWR@traceinfo{LWR@tabledatasinglecolumntag: about to print td tag}%
9437                \LWR@htmltag{%
9438                    td class=\textquotedbl{}td%
```

Append this column's spec:

```
9439                  \LWR@strresult%
```

If this column has a cmidrule, add "rule" to the end of the HTML class tag. Also add vertical bar tags.

```
9440                  \LWR@addcmidruletrim%
9441                  \LWR@addleftmostbartag%
9442                  \LWR@printbartag{\arabic{LWR@tableLaTeXcolindex}}%
```

Add any tabular > column text alignment or font control CSS:

```
9443                  \LWR@getexparray{LWR@coladdclass}%
9444                      {\arabic{LWR@tableLaTeXcolindex}}%
```

Close the class description:

```
9445                  \textquotedbl{}%
```

Add styles for rules, alignment:

```
9446                  \LWR@tdstartstyles%
9447                  \LWR@addcmidrulewidth%
9448                  \LWR@addcdashline%

9449                  \xdef\LWR@thiscolspec{%
9450                      \LWR@getexparray{LWR@tablecolspec}%
9451                          {\arabic{LWR@tableLaTeXcolindex}}%
9452                  }%
9453                  \LWR@addformatwpalignment{\LWR@thiscolspec}%
```

Add styles for cell and rule colors:

```
9454                  \LWR@addtabularrowcolor%
9455                  \LWR@addtabularrulecolors%

9456                  \LWR@tdendstyles%
9457              }% HTML td
9458              \LWR@traceinfo{LWR@tabledatasinglecolumntag: done printing td tag}%
```

If this is a p, m, b, or X column, allow paragraphs:

```
9459                  \ifboolexpr{%
9460                      test{ \ifdefstring{\LWR@strresult}{p} } or
```

```
9461                    test{ \ifdefstring{\LWR@strresult}{m} } or
9462                    test{ \ifdefstring{\LWR@strresult}{b} }
9463                  }%
9464                  {% allow pars
9465              \LWR@traceinfo{LWR@tabledatasinglecolumntag: about to LWR@startpars}%
9466                    \booltrue{LWR@tableparcell}%
9467                    \LWR@startpars%
9468              \LWR@traceinfo{LWR@tabledatasinglecolumntag: done with LWR@startpars}%
9469                  }% allow pars
9470                  {}% no pars
```

Print the > contents unless muted for the \bottomrule:

```
9471                  \ifboolexpr{bool{LWR@tabularmutemods} or bool{LWR@emptyatbang}}%
9472                  {}%
9473                  {%
9474                  \LWR@getexparray{\LWR@colbeforespec}{\arabic{LWR@tableLaTeXcolindex}}%
9475                  }%
9476                  \boolfalse{LWR@intabularmetadata}%
9477            }% not exiting tabular
9478      }{}% in tabular metadata
9479      \LWR@traceinfo{LWR@tabledatasinglecolumntag: done}%
9480 }%
```

75.20   Midrules

LWR@midrules   LWR@midrules is a data array (section 42) 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 42) of columns containing l if a midrule should be left trimmed for each column.

LWR@trimrrules   LWR@trimrrules is a data array (section 42) of columns containing r if a midrule should be right trimmed for each column.

LWR@cdashlines   LWR@cdashlines is a data array (section 42) 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.

```
9481 \newlength{\LWR@heavyrulewidth}
9482 \setlength{\LWR@heavyrulewidth}{.08em}
```

\LWR@lightrulewidth (*Len*)   The default width of the rule.

```
9483 \newlength{\LWR@lightrulewidth}
9484 \setlength{\LWR@lightrulewidth}{.05em}
```

\LWR@cmidrulewidth (*Len*)   The default width of the rule.

```
9485 \newlength{\LWR@cmidrulewidth}
9486 \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.

```
9487 \newlength{\LWR@thiscmidrulewidth}
9488 \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.

```
9489 \newcommand*{\LWR@clearmidrules}
9490 {%
9491     \defcounter{LWR@midrulecounter}{1}%
9492     \whileboolexpr{%
9493         not test{%
9494             \ifnumcomp{\value{LWR@midrulecounter}}{>}%
9495                 {\value{LWR@tabletotalLaTeXcols}}%
9496         }%
9497     }%
9498     {%
9499         \LWR@setexparray{LWR@midrules}{\arabic{LWR@midrulecounter}}{0pt}%
9500         \setlength{\LWR@thiscmidrulewidth}{\LWR@cmidrulewidth}%
9501         \LWR@setexparray{LWR@trimlrules}{\arabic{LWR@midrulecounter}}{}%
9502         \LWR@setexparray{LWR@trimrrules}{\arabic{LWR@midrulecounter}}{}%
9503         \LWR@setexparray{LWR@cdashlines}{\arabic{LWR@midrulecounter}}{N}%
9504         \defaddtocounter{LWR@midrulecounter}{1}%
9505     }%
9506 }
```

\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.

```
9507 \newcommand*{\LWR@subcmidrule}[4]{%
9508     \defcounter{LWR@midrulecounter}{#3}%
9509     \whileboolexpr{%
9510         not test {%
9511             \ifnumcomp{\value{LWR@midrulecounter}}{>}{#4}%
9512         }%
9513     }%
9514     {%
9515         \LWR@setexparray{LWR@midrules}{\arabic{LWR@midrulecounter}}{#1}%
9516         \defaddtocounter{LWR@midrulecounter}{1}%
9517     }% whiledo
9518     \IfSubStr{#2}{l}{\LWR@setexparray{LWR@trimlrules}{#3}{l}}{}%
9519     \IfSubStr{#2}{r}{\LWR@setexparray{LWR@trimrrules}{#4}{r}}{}%
9520     \booltrue{LWR@doingcmidrule}%
9521 }
```

\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.

```
9522 \NewDocumentCommand{\LWR@docmidrule}
9523     {O{\LWR@cmidrulewidth} D(){} >{\SplitArgument{1}{-}}m}
9524     {\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.

```
9525 \newcommand*{\LWR@subcdashline}[2]{%
9526     \defcounter{LWR@midrulecounter}{#1}%
9527     \whileboolexpr{%
9528         not test {%
9529             \ifnumcomp{\value{LWR@midrulecounter}}{>}{#2}%
9530         }%
9531     }%
9532     {%
9533         \LWR@setexparray{LWR@cdashlines}{\arabic{LWR@midrulecounter}}{Y}%
9534         \defaddtocounter{LWR@midrulecounter}{1}%
9535     }% whiledo
9536     \booltrue{LWR@doingcmidrule}%
9537 }
```

\LWR@docdashline {⟨*leftcolumn-rightcolumn*⟩}

    Marks LWR@cdashlines data array elements to be Y from left to right columns.

```
9538 \NewDocumentCommand{\LWR@docdashline}{>{\SplitArgument{1}{-}}m}%
9539 {%
9540     \LWR@subcdashline#1%
9541 }
```

\LWR@tdstartstyles Begins possibly adding a table data cell style.

```
9542 \newcommand*{\LWR@tdstartstyles}{\boolfalse{LWR@tdhavecellstyle}}
```

\LWR@tdaddstyle Starts adding a table data cell style.

```
9543 \newcommand*{\LWR@tdaddstyle}{%
9544     \ifbool{LWR@tdhavecellstyle}%
9545         {; }%
9546         { style=\textquotedbl}%
9547     \booltrue{LWR@tdhavecellstyle}%
9548 }
```

\LWR@tdendstyles Finishes possibly adding a table data cell style. Prints the closing quote.

```
9549 \newcommand*{\LWR@tdendstyles}{%
9550     \ifbool{LWR@tdhavecellstyle}%
9551         {%
9552             \textquotedbl%
9553             \boolfalse{LWR@tdhavecellstyle}%
9554         }{}%
9555 }
```

\LWR@subaddcmidruletrim {⟨*lefttrim*⟩} {⟨*righttrim*⟩} Adds a \cmidrule with optional trim.

```
9556 \newcommand*{\LWR@subaddcmidruletrim}[2]{%
9557     \setlength{\LWR@templengthone}{%
9558         \LWR@getexparray{LWR@midrules}{\arabic{LWR@tableLaTeXcolindex}}%
9559     }%
9560     \ifdimcomp{\LWR@templengthone}{>}{0pt}%
9561         {%
```

Print the class with left and right trim letters appended:

```
9562             \LWR@origtilde tdrule#1#2%
```

Remember the width of the rule:

```
9563             \setlength{\LWR@thiscmidrulewidth}{\LWR@templengthone}%
9564         }%
9565         {%
9566             \setlength{\LWR@thiscmidrulewidth}{0pt}%
9567         }%
9568 }
```

\LWR@addcmidruletrim Adds left or right trim to a \cmidrule.

```
9569 \newcommand*{\LWR@addcmidruletrim}{%
9570     \LWR@subaddcmidruletrim%
9571         {\LWR@getexparray{LWR@trimlrules}{\arabic{LWR@tableLaTeXcolindex}}}%
9572         {\LWR@getexparray{LWR@trimrrules}{\arabic{LWR@tableLaTeXcolindex}}}%
9573 }
```

\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.

```
9574 \newcommand{\LWR@addrulewidth}[2]{%
```

Only add a custom width if thiswidth is different than the defaultwidth, or if a color is being used:

```
9575     \ifboolexpr{%
9576         test{\ifdimcomp{#1}{=}{0pt}} or
9577         (
9578             ( test{\ifdimcomp{#1}{=}{#2}} and not bool{FormatWP} )
9579             and ( test {\ifdefvoid{\LWR@ruleHTMLcolor}} )
9580         )
9581     }%
9582     {}% default width and color
9583     {% custom width and/or color
```

Ensure that the width is wide enough to display in the browser:

```
9584         \LWR@forceminwidth{#1}%
```

Begin adding another style:

```
9585        \LWR@tdaddstyle%
```

The style itself:

```
9586        border-top:\LWR@printlength{\LWR@atleastonept} solid % space
```

If default gray, the darkness of the color depends on the thickness of the rule:

```
9587        \ifdefvoid{\LWR@ruleHTMLcolor}{%
9588            \ifdimcomp{#1}{<}{\LWR@lightrulewidth}%
9589            {\LWR@origpound{}A0A0A0}%
9590            {% lightrule or heaver
9591                \ifdimcomp{#1}{<}{\LWR@heavyrulewidth}%
9592                {\LWR@origpound{}808080}%
9593                {black}%
9594            }% lightrule or heavier
9595        }{%
9596            \LWR@origpound\LWR@ruleHTMLcolor%
9597        }%
9598    }% custom width and/or color
9599 }
```

\LWR@addcmidrulewidth Adds a style for the rule width.

Must be placed between \LWR@tdstartstyles and \LWR@tdendstyles.

```
9600 \newcommand{\LWR@addcmidrulewidth}{%
9601     \LWR@addrulewidth{\LWR@thiscmidrulewidth}{\LWR@cmidrulewidth}%
9602 }
```

\LWR@addcdashline Must be placed between \LWR@tdstartstyles and \LWR@tdendstyles.

```
9603 \newcommand{\LWR@addcdashline}{%
9604     \edef\LWR@tempone{%
9605         \LWR@getexparray{LWR@cdashlines}{\arabic{LWR@tableLaTeXcolindex}}%
9606     }%
9607     \ifdefstring{\LWR@tempone}{Y}{%
9608         \LWR@tdaddstyle%
9609         border-top: 1pt dashed %
9610         \ifdefvoid{\LWR@ruleHTMLcolor}%
9611            {black}%
9612            {\LWR@origpound\LWR@ruleHTMLcolor}%
9613    }{}%
9614 }
```

\LWR@WPcell {⟨*text-align*⟩} {⟨*vertical-align*⟩}

```
9615 \newcommand*{\LWR@WPcell}[2]{%
9616     \LWR@tdaddstyle%
9617     \LWR@print@mbox{text-align:#1}; \LWR@print@mbox{vertical-align:#2}%
9618 }
```

\LWR@addformatwpalignment {⟨*colspec*⟩}

If FormatWP, adds a style for the alignment.

Must be placed between \LWR@tdstartstyles and \LWR@tdendstyles.

```
9619 \newcommand*{\LWR@addformatwpalignment}[1]{%
9620     \ifbool{FormatWP}{%
9621         \IfSubStr{#1}{l}{\LWR@WPcell{left}{middle}}{}%
9622         \IfSubStr{#1}{c}{\LWR@WPcell{center}{middle}}{}%
9623         \IfSubStr{#1}{r}{\LWR@WPcell{right}{middle}}{}%
9624         \IfSubStr{#1}{p}{\LWR@WPcell{left}{bottom}}{}%
9625         \IfSubStr{#1}{m}{\LWR@WPcell{left}{middle}}{}%
9626         \IfSubStr{#1}{b}{\LWR@WPcell{left}{top}}{}%
9627     }{}%
9628 }
```

75.21 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.

```
9629 \newcommand*{\LWR@addtabularrowcolor}{%
9630     \ifbool{LWR@tabularmutemods}{}{%
9631         \ifdefvoid{\LWR@rowHTMLcolor}{%
9632             \ifdefvoid{\LWR@xcolorrowHTMLcolor}{}%
9633             {% xcolor row color
9634                 \LWR@tdaddstyle%
9635                 background:\LWR@origpound\LWR@xcolorrowHTMLcolor%
9636             }%
9637         }%
9638         {% explicit row color
9639             \LWR@tdaddstyle%
9640             background:\LWR@origpound\LWR@rowHTMLcolor%
9641         }%
9642     }%
9643 }
```

\LWR@addtabularhrulecolor Adds a cell's horizontal rule color style, if needed.

```
9644 \newcommand*{\LWR@addtabularhrulecolor}{%
```

If either form of horizontal rule is requested:

```
9645     \ifboolexpr{%
9646         test{\ifnumcomp{\value{LWR@hlines}}{>}{0}} or%
9647         test{\ifnumcomp{\value{LWR@hdashedlines}}{>}{0}} or%
9648         bool{LWR@doingtbrule}%
9649     }{%
```

If there is a no custom color:

```
9650         \ifdefvoid{\LWR@ruleHTMLcolor}%
9651         {%
9652             \ifnumcomp{\value{LWR@hlines}}{>}{1}%
9653             {%
9654                 \LWR@tdaddstyle%
9655                 border-top: 4px double%
9656             }{% else
```

```
9657                \ifnumcomp{\value{LWR@hdashedlines}}{>}{1}%
9658                {%
9659                    \LWR@tdaddstyle%
9660                    border-top: 2px dashed%
9661                }{% else
9662                \ifnumcomp{\value{LWR@hdashedlines}}{=}{1}%
9663                {%
9664                    \LWR@tdaddstyle%
9665                    border-top: 1px dashed%
9666                }{}}}%
```

If no color and not doubled or dashed, then add nothing, since a simpler rule is the default.

```
9667            }%
```

If there is a custom color:

```
9668            {%
9669                \ifnumcomp{\value{LWR@hlines}}{>}{1}%
9670                {%
9671                    \LWR@tdaddstyle%
9672                    border-top: 4px double \LWR@origpound\LWR@ruleHTMLcolor%
9673                }{% else
9674                \ifnumcomp{\value{LWR@hdashedlines}}{>}{1}%
9675                {%
9676                    \LWR@tdaddstyle%
9677                    border-top: 2px dashed \LWR@origpound\LWR@ruleHTMLcolor%
9678                }{% else
9679                \ifnumcomp{\value{LWR@hdashedlines}}{=}{1}%
9680                {%
9681                    \LWR@tdaddstyle%
9682                    border-top: 1px dashed \LWR@origpound\LWR@ruleHTMLcolor%
9683                }{% else
9684                    \LWR@tdaddstyle%
9685                    border-top: 1px solid \LWR@origpound\LWR@ruleHTMLcolor%
9686                }}}%
9687            }%
9688        }{}%
9689 }
```

\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.

```
9690 \newcommand*{\LWR@addtabularrulecolors}{%
```

Custom horizonal rule color:

```
9691        \LWR@addtabularhrulecolor%
```

No vertical rules if finishing the tabular with a row of empty cells:

```
9692        \ifbool{LWR@tabularmutemods}{}{%
```

If at the leftmost cell, possibly add a leftmost vertical rule:

```
9693            \ifnumequal{\value{LWR@tableLaTeXcolindex}}{1}{%
```

Fetch the left edge's vertical bar specification:

```
9694            \edef\LWR@tempone{\LWR@getexparray{LWR@colbarspec}{leftedge}}%
```

Add a custom style if a vertical bar was requested:

```
9695            \ifdefstring{\LWR@tempone}{tvertbarl}{%
9696                \LWR@tdaddstyle%
9697                border-left: 1px solid % space
9698                    \LWR@vertruleHTMLcolor%
9699            }{}%
9700            \ifdefstring{\LWR@tempone}{tvertbarldouble}{%
9701                \LWR@tdaddstyle%
9702                border-left: 4px double % space
9703                    \LWR@vertruleHTMLcolor%
9704            }{}%
9705            \ifdefstring{\LWR@tempone}{tvertbarldash}{%
9706                \LWR@tdaddstyle%
9707                border-left: 1px dashed % space
9708                    \LWR@vertruleHTMLcolor%
9709            }{}%
9710            \ifdefstring{\LWR@tempone}{tvertbarldoubledash}{%
9711                \LWR@tdaddstyle%
9712                border-left: 2px dashed % space
9713                    \LWR@vertruleHTMLcolor%
9714            }{}%
9715        }{}%
```

Possibly add a right vertical rule for this cell:

```
9716        \edef\LWR@tempone{%
9717         \LWR@getexparray{LWR@colbarspec}{\arabic{LWR@tableLaTeXcolindex}}%
9718        }%
9719        \ifdefstring{\LWR@tempone}{tvertbarr}{%
```

Add a custom style if a vertical bar was requested:

```
9720                \LWR@tdaddstyle%
9721                border-right: 1px solid \LWR@vertruleHTMLcolor%
9722            }{}%
9723            \ifdefstring{\LWR@tempone}{tvertbarrdouble}{%
9724                \LWR@tdaddstyle%
9725                border-right: 4px double \LWR@vertruleHTMLcolor%
9726            }{}%
9727            \ifdefstring{\LWR@tempone}{tvertbarrdash}{%
9728                \LWR@tdaddstyle%
9729                border-right: 1px dashed \LWR@vertruleHTMLcolor%
9730            }{}%
9731            \ifdefstring{\LWR@tempone}{tvertbarrdoubledash}{%
9732                \LWR@tdaddstyle%
9733                border-right: 2px dashed \LWR@vertruleHTMLcolor%
9734            }{}%
9735        }%
9736 }
```

\LWR@subaddtabularcellcolor {⟨*html color*⟩}

```
9737 \newcommand*{\LWR@subaddtabularcellcolor}[1]{%
9738    \LWR@htmltag{div class=\textquotedbl{}cellcolor\textquotedbl\ % space
```

```
9739        style=\textquotedbl{}%
9740        background:\LWR@origpound{}{}#1 %
9741        \textquotedbl\ %
9742    }% space
9743    \defaddtocounter{LWR@cellcolordepth}{1}%
9744 }
```

\LWR@addtabularcellcolor Adds a cell color style, if needed.

```
9745 \newcommand*{\LWR@addtabularcellcolor}{%
9746    \ifdefvoid{\LWR@cellHTMLcolor}%
9747    {%
9748        \ifdefvoid{\LWR@rowHTMLcolor}%
9749        {%
9750            \ifdefvoid{\LWR@xcolorrowHTMLcolor}%
9751            {%
9752                \ifdefvoid{\LWR@columnHTMLcolor}%
9753                {}%
9754                {\LWR@subaddtabularcellcolor{\LWR@columnHTMLcolor}}%
9755            }%
9756            {\LWR@subaddtabularcellcolor{\LWR@xcolorrowHTMLcolor}}%
9757        }%
9758        {\LWR@subaddtabularcellcolor{\LWR@rowHTMLcolor}}%
9759    }%
9760    {\LWR@subaddtabularcellcolor{\LWR@cellHTMLcolor}}%
9761 }
```

75.22   Multicolumns

75.22.1   Parsing multicolumns

\LWR@printmccoltype@normal {⟨col type⟩}

Prints the column type, and remembers that any vertical bars are no longer on the left edge.

```
9762 \newcommand*{\LWR@printmccoltype@normal}[1]{%
9763    #1%
9764    \boolfalse{LWR@mcolvertbaronleft}%
9765 }
```

\LWR@printmccoltype@ignore {⟨col type⟩}

This type does not print a multi-column data cell.

```
9766 \newcommand*{\LWR@printmccoltype@ignore}[1]{}%
```

\LWR@printmccoltype@vertbar {⟨col type⟩}

Adds a left or right vertical bar.

```
9767 \newcommand*{\LWR@printmccoltype@vertbar}[1]{%
9768    \ifbool{LWR@mcolvertbaronleft}%
9769        {\defaddtocounter{LWR@mcolvertbarsl}{1}}% left edge
9770        {\defaddtocounter{LWR@mcolvertbarsr}{1}}% not left edge
9771 }
```

`\LWR@printmccoltype@colon` {⟨*col type*⟩}

Adds a left or right vertical bar.

```
9772 \newcommand*{\LWR@printmccoltype@colon}[1]{%
9773     \ifbool{LWR@mcolvertbaronleft}%
9774         {\defaddtocounter{LWR@mcolvertbarsldash}{1}}% left edge
9775         {\defaddtocounter{LWR@mcolvertbarsrdash}{1}}% not left edge
9776 }
```

`\LWR@printmccoltype@semicolon` {⟨*col type*⟩}

Adds a left or right vertical bar.

```
9777 \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`.

```
9778 \newcommand*{\LWR@printmccoltype}[1]{%
9779     \LWR@traceinfo{lwr@printmccoltype -#1-}%
```

Get one token of the column spec:

```
9780     \StrChar{#1}{\arabic{LWR@tablemulticolspos}}[\LWR@strresult]%
```

Detokenize to avoid problems with special characters:

```
9781     \edef\LWR@strresult{\detokenize\expandafter{\LWR@strresult}}%
```

Add to the HTML tag depending on which column type is found:

```
9782     \ifcsdef{LWR@columntype@mctype@\LWR@strresult}%
9783         {\csuse{LWR@columntype@mctype@\LWR@strresult}}%
9784         {\boolfalse{LWR@mcolvertbaronleft}}%
9785     \LWR@traceinfo{lwr@printmccoltype done}%
9786 }
```

`\LWR@printmccoldata@other` {⟨*num args to skip*⟩} {⟨*entire colspec*⟩}

For @, !, >, <, print the next token without paragraph tags:

```
9787 \newcommand*{\LWR@printmccoldata@other}[2]{%
9788     \defaddtocounter{LWR@tablemulticolspos}{1}%
9789     \StrChar{#2}{\arabic{LWR@tablemulticolspos}}[\LWR@strresult]%
9790     \LWR@strresult%
```

A valid column data type was found:

```
9791     \booltrue{LWR@validtablecol}%
9792 }
```

\LWR@printmccoldata@skip {⟨*num args to skip*⟩} {⟨*entire colspec*⟩}

Nothing to print for this column type.

```
9793 \newcommand*{\LWR@printmccoldata@skip}[2]{%
9794     \defaddtocounter{LWR@tablemulticolspos}{#1}%
```

A valid column data type was found:

```
9795     \booltrue{LWR@validtablecol}%
9796 }
```

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*⟩}

```
9797 \newcommand*{\LWR@printmccoldata@normal}[2]{%
9798     \LWR@multicoltext%
9799     \defaddtocounter{LWR@tablemulticolspos}{#1}%
9800 }
```

\LWR@printmccoldata@paragraph {⟨*num args to skip*⟩} {⟨*entire colspec*⟩}

```
9801 \newcommand*{\LWR@printmccoldata@paragraph}[2]{%
9802     \LWR@startpars%
9803     \LWR@multicoltext%
9804     \defaddtocounter{LWR@tablemulticolspos}{#1}%
9805     \LWR@stoppars%
9806 }
```

\LWR@printmccoldata {⟨*entire colspec*⟩}

Print the data for any valid column type found.

```
9807 \newcommand*{\LWR@printmccoldata}[1]{%
9808     \LWR@traceinfo{lwr@printmccoldata -#1}%
```

Not yet found a valid column type:

```
9809     \boolfalse{LWR@validtablecol}%
```

Get one token of the column spec, into a local copy in case nested.

```
9810     \StrChar{#1}{\arabic{LWR@tablemulticolspos}}[\LWR@strresult]%
9811     \edef\LWR@printmccoldatatoken{\LWR@strresult}%
```

Print the text depending on which column type is found. Also handles @, >, < as it comes to them.

```
9812     \ifcsdef{LWR@columntype@mcdata@\LWR@printmccoldatatoken}%
9813         {\csuse{LWR@columntype@mcdata@\LWR@printmccoldatatoken}{#1}}%
9814         {}%
```

If an unknown column type, print the text:

```
9815      \ifbool{LWR@validtablecol}{}{\LWR@multicoltext{}}%
```

Tracing:

```
9816      \LWR@traceinfo{lwr@printmccoldata done}%
9817 }
```

\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.

```
9818 \newcommand*{\LWR@parsemulticolumnalignment}[2]{%
9819      \defcounter{LWR@tablemulticolspos}{1}%
9820      \StrLen{#1}[\LWR@strresult]%
9821      \defcounter{LWR@tablemulticolswidth}{\LWR@strresult}%
```

Scan across the tokens in the column spec:

```
9822      \whileboolexpr{%
9823          not test {%
9824              \ifnumcomp{\value{LWR@tablemulticolspos}}{>}%
9825                  {\value{LWR@tablemulticolswidth}}%
9826          }%
9827      }%
9828      {%
```

Execute the assigned print function for each token in the column spec:

```
9829          \csuse{#2}{#1}%
```

Move to the next token in the column spec:

```
9830          \defaddtocounter{LWR@tablemulticolspos}{1}%
9831      }%
9832 }
```

75.22.2 Multicolumn factored code

\LWR@addmulticolvertrulecolor

```
9833 \newcommand*{\LWR@addmulticolvertrulecolor}{%
```

No vertical rules if finishing the tabular with a row of empty cells:

```
9834      \ifbool{LWR@tabularmutemods}{}{%
```

Left side:

```
9835          \ifnumcomp{\value{LWR@mcolvertbarsl}}{=}{1}{%
9836              \LWR@tdaddstyle%
9837              border-left: 1px solid \LWR@vertruleHTMLcolor%
```

```
9838        }{}%
9839        \ifnumcomp{\value{LWR@mcolvertbarsl}}{>}{1}{%
9840            \LWR@tdaddstyle%
9841            border-left: 4px double \LWR@vertruleHTMLcolor%
9842        }{}%
9843        \ifnumcomp{\value{LWR@mcolvertbarsldash}}{=}{1}{%
9844            \LWR@tdaddstyle%
9845            border-left: 1px dashed \LWR@vertruleHTMLcolor%
9846        }{}%
9847        \ifnumcomp{\value{LWR@mcolvertbarsldash}}{>}{1}{%
9848            \LWR@tdaddstyle%
9849            border-left: 2px dashed \LWR@vertruleHTMLcolor%
9850        }{}%
```

Right side:

```
9851        \ifnumcomp{\value{LWR@mcolvertbarsr}}{=}{1}{%
9852            \LWR@tdaddstyle%
9853            border-right: 1px solid \LWR@vertruleHTMLcolor%
9854        }{}%
9855        \ifnumcomp{\value{LWR@mcolvertbarsr}}{>}{1}{%
9856            \LWR@tdaddstyle%
9857            border-right: 4px double \LWR@vertruleHTMLcolor%
9858        }{}%
9859        \ifnumcomp{\value{LWR@mcolvertbarsrdash}}{=}{1}{%
9860            \LWR@tdaddstyle%
9861            border-right: 1px dashed \LWR@vertruleHTMLcolor%
9862        }{}%
9863        \ifnumcomp{\value{LWR@mcolvertbarsrdash}}{>}{1}{%
9864            \LWR@tdaddstyle%
9865            border-right: 2px dashed \LWR@vertruleHTMLcolor%
9866        }{}%
9867    }%
9868 }
```

```
9869 \newcommand{\LWR@multicoltext}{}
```

To find multicolumn right trim:

```
9870 \newcounter{LWR@lastmulticolumn}
```

\LWR@domulticolumn [⟨*1: vpos*⟩] [⟨*2: #rows*⟩] {⟨*3: numLaTeXcols*⟩} {⟨*4: numHTMLcols*⟩} {⟨*5: colspec*⟩}
{⟨*6: text*⟩}

```
9871 \NewDocumentCommand{\LWR@domulticolumn}{o o m m m +m}{%
9872    \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:

```
9873    \renewcommand{\LWR@multicoltext}{%
9874        #6%
9875        \booltrue{LWR@validtablecol}%
9876    }%
```

Expand the preamble and save it.

```
9877        \LWR@expandpreamble{#5}%
9878        \edef\LWR@origmccolspec{\the\@temptokena}%
```

Compute the rightmost column to be included. This is used to create the right trim.

```
9879        \defcounter{LWR@lastmulticolumn}{\value{LWR@tableLaTeXcolindex}}%
9880        \defaddtocounter{LWR@lastmulticolumn}{#3}%
9881        \defaddtocounter{LWR@lastmulticolumn}{-1}%
```

Row processing:

```
9882        \LWR@maybenewtablerow%
```

Begin the opening table data tag:

```
9883    \LWR@htmltag{%
9884        td colspan=\textquotedbl#4\textquotedbl\ %

9885        \IfValueT{#2}{ % rows?
9886            rowspan=\textquotedbl#2\textquotedbl\ %
9887        }%

9888        class=\textquotedbl{}td%
```

Print the column type and vertical bars:

```
9889        \defcounter{LWR@mcolvertbarsl}{0}%
9890        \defcounter{LWR@mcolvertbarsr}{0}%
9891        \defcounter{LWR@mcolvertbarsldash}{0}%
9892        \defcounter{LWR@mcolvertbarsrdash}{0}%
9893        \booltrue{LWR@mcolvertbaronleft}%
9894      \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:

```
9895            \LWR@subaddcmidruletrim%
9896          {%
9897                \LWR@getexparray{LWR@trimlrules}%
9898                    {\arabic{LWR@tableLaTeXcolindex}}%
9899          }%
9900          {%
9901                \LWR@getexparray{LWR@trimrrules}%
9902                    {\arabic{LWR@lastmulticolumn}}%
9903          }%
```

Also add vertical bar class.

```
9904        \ifnumcomp{\value{LWR@mcolvertbarsl}}{=}{1}{ tvertbarl}{}%
9905        \ifnumcomp{\value{LWR@mcolvertbarsl}}{>}{1}{ tvertbarldouble}{}%
9906        \ifnumcomp{\value{LWR@mcolvertbarsr}}{=}{1}{ tvertbarr}{}%
9907        \ifnumcomp{\value{LWR@mcolvertbarsr}}{>}{1}{ tvertbarrdouble}{}%
9908        \ifnumcomp{\value{LWR@mcolvertbarsldash}}{=}{1}{ tvertbarldash}{}%
9909        \ifnumcomp{\value{LWR@mcolvertbarsldash}}{>}{1}%
9910            { tvertbarldoubledash}{}%
```

```
9911        \ifnumcomp{\value{LWR@mcolvertbarsrdash}}{=}{1}{ tvertbarrdash}{}%
9912        \ifnumcomp{\value{LWR@mcolvertbarsrdash}}{>}{1}%
9913            { tvertbarrdoubledash}{}%
```

Close the class tag's opening quote:

```
9914        \textquotedbl{}%
```

```
9915        \LWR@tdstartstyles%
```

Style for vertical position:

```
9916        \IfValueT{#1}{% vpos?
9917            \ifstrequal{#1}{b}%
9918                {%
9919                    \LWR@tdaddstyle%
9920                    \LWR@print@mbox{vertical-align:bottom}%
9921                }{}%
9922            \ifstrequal{#1}{t}%
9923                {%
9924                    \LWR@tdaddstyle%
9925                    \LWR@print@mbox{vertical-align:top}%
9926                }{}%
9927        }% vpos?
```

Style for row colors:

```
9928        \LWR@addtabularrowcolor%
```

Other styles:

```
9929        \LWR@addcmidrulewidth%
9930        \LWR@addcdashline%
9931        \LWR@addtabularhrulecolor%
9932        \LWR@addmulticolvertrulecolor%
9933        \LWR@addformatwpalignment{\LWR@origmccolspec}%
9934        \LWR@tdendstyles%
9935    }% end of the opening table data tag
9936    \boolfalse{LWR@intabularmetadata}%
9937    \LWR@parsemulticolumnalignment{\LWR@origmccolspec}{LWR@printmccoldata}%
9938 }
```

75.22.3   Multicolumn

\LWR@htmlmulticolumn {⟨*numcols*⟩} {⟨*alignment*⟩} {⟨*text*⟩}

```
9939 \NewDocumentCommand{\LWR@htmlmulticolumn}{m m +m}%
9940 {%
```

Figure out how many extra HTML columns to add for @ and ! columns:

```
9941        \LWR@tabularhtmlcolumns{\arabic{LWR@tableLaTeXcolindex}}{#1}%
```

Create the multicolumn tag:

```
9942        \LWR@domulticolumn{#1}{\arabic{LWR@tabhtmlcoltotal}}{#2}{#3}%
```

Move to the next LATEX column:

```
9943      \defaddtocounter{LWR@tableLaTeXcolindex}{#1}%
9944      \defaddtocounter{LWR@tableLaTeXcolindex}{-1}%
```

Skip any trailing @ or ! columns for this cell:

```
9945      \booltrue{LWR@skipatbang}%
9946 }
```

75.22.4   Longtable captions

longtable captions use \multicolumn.

Per the caption package. User-redefinable float type.

```
9947 \providecommand*{\LTcaptype}{table}
```

\LWR@longtabledatacaptiontag * [⟨*toc entry*⟩] {⟨*caption*⟩}

```
9948 \NewDocumentCommand{\LWR@longtabledatacaptiontag}{s o +m}
9949 {%
```

Remember the latest name for \nameref:

```
9950      \IfValueTF{#2}{% optional given?
9951          \ifblank{#2}% optional empty?
9952          {\LWR@setlatestname{#3}}% empty
9953          {\LWR@setlatestname{#2}}% given and non-empty
9954      }% optional given
9955      {\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:

```
9956      \LWR@tabularhtmlcolumns{1}{\arabic{LWR@tabletotalLaTeXcols}}%
```

Create the multicolumn tag. The caption will be centered by the CSS caption class.

```
9957      \LWR@domulticolumn{\arabic{LWR@tabletotalLaTeXcols}}%
9958          {\arabic{LWR@tabhtmlcoltotal}}%
9959          {p}%
9960      {% \LWR@domulticolumn
9961      \IfBooleanTF{#1}% star?
```

Star version, show a caption but do not make a LOT entry:

```
9962      {% yes star
9963          \LWR@figcaption%
9964          \LWR@isolate{#3}%
9965          \endLWR@figcaption%
9966      }%
9967      {% No star:
```

Not the star version:

Don't step the counter if \caption[]{A caption.}

```
9968          \ifbool{LWR@starredlongtable}%
9969          {%
9970              \ifblank{#2}% TOC entry
9971              {}%
9972              {%
9973                  \refstepcounter{\LTcaptype}%
9974                  \protected@edef\@currentlabel{%
9975                      \@nameuse{p@\LTcaptype}\@nameuse{the\LTcaptype}%
9976                  }%
9977              }%
9978          }{}%
```

Create an HTML caption. Afterwards, maybe make a LOT entry.

```
9979          \LWR@figcaption%
9980          \LWR@isolate{\@nameuse{fnum@\LTcaptype}}%
9981          \CaptionSeparator%
9982          \LWR@isolate{#3}%
9983          \endLWR@figcaption%
```

See if an optional caption was given:

```
9984          \ifblank{#2}% TOC entry empty
```

if the optional caption was given, but empty, do not form a TOC entry

```
9985          {}%
```

If the optional caption was given, but might only be []:

```
9986          {% TOC entry not empty
9987              \IfNoValueTF{#2}% No TOC entry?
```

The optional caption is []:

```
9988              {% No TOC entry
9989                  \addcontentsline%
9990                  {\@nameuse{ext@\LTcaptype}}%
9991                  {\LTcaptype}%
9992                  {%
9993                      \protect\numberline%
9994              {\LWR@isolate{\@nameuse{p@\LTcaptype}}\@nameuse{the\LTcaptype}}%
9995                      {\ignorespaces \LWR@isolate{#3}\protect\relax}%
9996                  }%
9997              }% end of No TOC entry
```

The optional caption has text enclosed:

```
9998              {% yes TOC entry
9999                  \addcontentsline%
10000                 {\@nameuse{ext@\LTcaptype}}%
10001                 {\LTcaptype}%
10002                 {%
10003                     \protect\numberline%
10004             {\LWR@isolate{\@nameuse{p@\LTcaptype}}\@nameuse{the\LTcaptype}}%
```

```
10005                        {\ignorespaces \LWR@isolate{#2}\protect\relax}%
10006                   }%
10007              }% end of yes TOC entry
10008          }% end of TOC entry not empty
10009       }% end of no star
```

Skip any trailing @ or ! columns for this cell:

```
10010       \booltrue{LWR@skipatbang}%
10011       }% end of \LWR@domulticolumn
10012   \defaddtocounter{LWR@tableLaTeXcolindex}{\value{LWR@tabletotalLaTeXcols}}%
10013       \defaddtocounter{LWR@tableLaTeXcolindex}{-1}
10014
10015 }
```

75.22.5   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.

```
10016 \newcounter{LWR@tabhtmlcolindex}
10017 \newcounter{LWR@tabhtmlcolend}
10018 \newcounter{LWR@tabhtmlcoltotal}
```

\LWR@subtabularhtmlcolumns {⟨index⟩}

Factored from \LWr@tabularhtmlcolumns, which follows.

```
10019 \newcommand*{\LWR@subtabularhtmlcolumns}[1]{%
```

Temporarily define a macro equal to the @ specification for this column:

```
10020       \edef\LWR@atbangspec{\LWR@getexparray{LWR@colatspec}{#1}}%
```

If the @ specification is not empty, add to the count:

```
10021       \ifdefempty{\LWR@atbangspec}%
10022          {}%
10023          {\defaddtocounter{LWR@tabhtmlcoltotal}{1}}%
```

Likewise for the ! columns:

```
10024       \edef\LWR@atbangspec{\LWR@getexparray{LWR@colbangspec}{#1}}%
10025       \ifdefempty{\LWR@atbangspec}%
10026          {}%
10027          {\defaddtocounter{LWR@tabhtmlcoltotal}{1}}%
10028 }
```

\LWR@tabularhtmlcolumns {⟨*starting LATEX column*⟩} {⟨*number LATEX columns*⟩}

Compute the total number of HTML columns being spanned, considering the start-ing LATEX table column and the number of LATEX 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.

```
10029 \newcommand*{\LWR@tabularhtmlcolumns}[2]{%
```

Count the starting index, compute ending index, and begin with the count being the LATEX span, to which additional @ and ! columns may be added:

```
10030    \defcounter{LWR@tabhtmlcolindex}{#1}%
10031    \defcounter{LWR@tabhtmlcoltotal}{#2}%
10032    \defcounter{LWR@tabhtmlcolend}{#1}%
10033    \defaddtocounter{LWR@tabhtmlcolend}{#2}%
```

If at the left edge, add the at/bang columns for the left edge:

```
10034    \ifnumcomp{\value{LWR@tabhtmlcolindex}}{=}{1}{%
10035        \LWR@subtabularhtmlcolumns{leftedge}%
10036    }{}%
```

Walk across the LATEX columns looking for @ and ! columns:

```
10037    \whileboolexpr{%
10038        test {%
10039        \ifnumcomp{\value{LWR@tabhtmlcolindex}}{<}{\value{LWR@tabhtmlcolend}}%
10040        }%
10041    }%
10042    {%
10043        \LWR@subtabularhtmlcolumns{\arabic{LWR@tabhtmlcolindex}}%
10044        \defaddtocounter{LWR@tabhtmlcolindex}{1}%
10045    }% whiledo
10046 }
```

```
10047 \end{warpHTML}
```

75.23 Multirow if not loaded

A default defintion in case **multirow** is not loaded. This is used during table parsing.

```
10048 \begin{warpHTML}
10049 \newcommand{\multirow}[2][c]{}
10050 \end{warpHTML}
```

75.24 Multicolumnrow

A print-mode version is defined here, and is also used during HTML output while inside a lateximage.

See section 429 for the HTML versions.

for HTML & PRINT: 10051 \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 re-
quires several tests to determine which version is loaded and thus which options
are available.

```
10052 \AtBeginDocument{
```

\@ifundefined{@xmultirow} determines if multirow was never loaded.

Null action if not loaded:

```
10053 \@ifundefined{@xmultirow}
10054 {
10055 \DeclareExpandableDocumentCommand{\LWR@print@multicolumnrow}%
10056     {+m +m +O{c} +m +O{0} +m +O{0pt} +m}%
10057     {}%
10058 }% no version of multirow was loaded
10059 {% \@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:

```
10060 \IfPackageLoadedTF{multirow}{% v2.0 or newer
10061 \IfPackageAtLeastTF{multirow}{2016/09/01}% 2016/09/27 for v2.0
10062 {% v2.0+:
10063 \DeclareExpandableDocumentCommand{\LWR@print@multicolumnrow}%
10064     {+m +m +O{c} +m +O{0} +m +O{0pt} +m}%
10065     {\multicolumn{#1}{#2}{\@xmultirow[#3]{#4}[#5]{#6}[#7]{#8}}}%
10066 }
10067 {% loaded but older, probably not executed:
10068 \DeclareExpandableDocumentCommand{\LWR@print@multicolumnrow}%
10069     {+m +m +O{c} +m +O{0} +m +O{0pt} +m}%
10070     {\multicolumn{#1}{#2}{\@xmultirow{#4}[#5]{#6}[#7]{#8}}}%
10071 }
10072 }% 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.

```
10073 {% v1.6 or older did not \ProvidePackage
10074 \DeclareExpandableDocumentCommand{\LWR@print@multicolumnrow}%
10075     {+m +m +O{c} +m +O{0} +m +O{0pt} +m}%
10076     {\multicolumn{#1}{#2}{\@xmultirow{#4}[#5]{#6}[#7]{#8}}}%
10077 }
10078
10079 }% \@ifundefined{@xmultirow}
10080
10081 \providecommand*{\multicolumnrow}{\LWR@print@multicolumnrow}
10082
10083 }% AtBeginDocument
```

10084 `\end{warpall}`

75.25   Utility macros inside a table

10085 `\begin{warpHTML}`

Used to prevent opening a tabular data cell if the following token is one which does not create tabular data:

10086 `\newcommand*{\LWR@donothing}{}`

In case array is not loaded:

10087 `\let\firsthline\relax`
10088 `\let\lasthline\relax`
10089 `\newcommand*{\firsthline}{}`
10090 `\newcommand*{\lasthline}{}`

In case bigdelim is not loaded:

10091 `\newcommand*{\ldelim}{}`
10092 `\newcommand*{\rdelim}{}`

10093 `\end{warpHTML}`

75.26   Special-case tabular markers

10094 `\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.

10095 `\newcommand*{\TabularMacro}{}`

10096 `\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
```

10097 \begin{warpHTML}

10098 \newcommand*{\ResumeTabular}{%
10099   \boolfalse{LWR@exitingtabular}%
10100   \boolfalse{LWR@tabularmutemods}%
10101   \boolfalse{LWR@tabularfinalrow}%
10102   \LWR@getmynexttoken%
10103 }

10104 \end{warpHTML}

10105 \begin{warpprint}

10106 \newcommand*{\ResumeTabular}{}

10107 \end{warpprint}

75.27   Checking for a new table cell

10108 \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:

10109 \newcommand*{\LWR@tabledatacolumntag}%
10110 {%
10111   \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:

10112   \global\let\LWR@mynextaction\LWR@tabledatasinglecolumntag%

If find \end, exit the tabular:

10113   \ifdefequal{\LWR@mynexttoken}{\end}%
10114   {%
10115   \booltrue{LWR@tabularfinalrow}%
10116   \booltrue{LWR@exitingtabular}%
10117   }{}%

longtable can have a caption in a cell

```
10118    \ifdefequal{\LWR@mynexttoken}{\caption}%
10119        {\global\let\LWR@mynextaction\LWR@donothing}{}%
```

Look for other things which would not start a table cell:

```
10120    \ifdefequal{\LWR@mynexttoken}{\multicolumn}%
10121        {\global\let\LWR@mynextaction\LWR@donothing}{}%
10122    \ifdefequal{\LWR@mynexttoken}{\multirow}%
10123        {\global\let\LWR@mynextaction\LWR@donothing}{}%
10124    \ifdefequal{\LWR@mynexttoken}{\multicolumnrow}%
10125        {\global\let\LWR@mynextaction\LWR@donothing}{}%
10126    \ifdefequal{\LWR@mynexttoken}{\noalign}%
10127        {\global\let\LWR@mynextaction\LWR@donothing}{}%
```

If an \mrowcell, this is a cell to be skipped over:

```
10128    \ifdefequal{\LWR@mynexttoken}{\mrowcell}%
10129        {\global\let\LWR@mynextaction\LWR@donothing}{}%
```

If an \mcolrowcell, this is a cell to be skipped over:

```
10130    \ifdefequal{\LWR@mynexttoken}{\mcolrowcell}%
10131        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10132    \ifdefequal{\LWR@mynexttoken}{\TabularMacro}%
10133        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10134    \ifdefequal{\LWR@mynexttoken}{\hline}%
10135        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10136    \ifdefequal{\LWR@mynexttoken}{\firsthline}%
10137        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10138    \ifdefequal{\LWR@mynexttoken}{\lasthline}%
10139        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10140    \ifdefequal{\LWR@mynexttoken}{\toprule}%
10141        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10142    \ifdefequal{\LWR@mynexttoken}{\midrule}%
10143        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10144    \ifdefequal{\LWR@mynexttoken}{\cmidrule}%
10145        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10146    \ifdefequal{\LWR@mynexttoken}{\morecmidrules}%
10147        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10148    \ifdefequal{\LWR@mynexttoken}{\specialrule}%
10149        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10150    \ifdefequal{\LWR@mynexttoken}{\cline}%
10151        {\global\let\LWR@mynextaction\LWR@donothing}{}%
```

```
10152    \ifdefequal{\LWR@mynexttoken}{\bottomrule}%
10153        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10154    \ifdefequal{\LWR@mynexttoken}{\hhline}%
10155        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10156    \ifdefequal{\LWR@mynexttoken}{\rowcolor}%
10157        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10158    \ifdefequal{\LWR@mynexttoken}{\arrayrulecolor}%
10159        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10160    \ifdefequal{\LWR@mynexttoken}{\doublerulesepcolor}%
10161        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10162    \ifdefequal{\LWR@mynexttoken}{\warpprintonly}%
10163        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10164    \ifdefequal{\LWR@mynexttoken}{\warpHTMLonly}%
10165        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10166    \ifdefequal{\LWR@mynexttoken}{\ldelim}%
10167        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10168    \ifdefequal{\LWR@mynexttoken}{\rdelim}%
10169        {\global\let\LWR@mynextaction\LWR@donothing}{}%
```

For arydshln:

```
10170    \ifdefequal{\LWR@mynexttoken}{\hdashline}%
10171        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10172    \ifdefequal{\LWR@mynexttoken}{\cdashline}%
10173        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10174    \ifdefequal{\LWR@mynexttoken}{\firsthdashline}%
10175        {\global\let\LWR@mynextaction\LWR@donothing}{}%

10176    \ifdefequal{\LWR@mynexttoken}{\lasthdashline}%
10177        {\global\let\LWR@mynextaction\LWR@donothing}{}%
```

Ignore an empty line between rows:

```
10178    \ifdefequal{\LWR@mynexttoken}{\par}%
10179        {%
10180            \global\let\LWR@mynextaction\LWR@donothing%
10181        }{}%
```

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:

```
10182    \LWR@traceinfo{LWR@tabledatacolumntag: done, about to do LWR@mynextaction}%
10183    \LWR@mynextaction%
10184 }
```

10185 `\end{warpHTML}`

75.28 `\mrowcell`

10186 `\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.

10187 `\newcommand*{\mrowcell}{}`

10188 `\end{warpall}`

75.29 `\mcolrowcell`

10189 `\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.

10190 `\newcommand*{\mcolrowcell}{}`

10191 `\end{warpall}`

75.30 HTML tabular environment

10192 `\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.

10193 `\providecommand*{\toprule}[1][]{\hline}`
10194 `\providecommand*{\midrule}[1][]{\hline}`
10195 `\providecommand*{\cmidrule}{\cline}`
10196 `\providecommand*{\bottomrule}[1][]{\hline}`
10197 `\providecommand*{\addlinespace}[1][]{}`
10198 `\providecommand*{\morecmidrules}{}`
10199 `\providecommand*{\specialrule}[3]{\hline}`

\noalign {⟨*text*⟩} Redefined for use inside tabular.

10200 `\LetLtxMacro\LWR@orignoalign\noalign`
10201
10202 `\newcommand{\LWR@tabularnoalign}[1]{%`
10203   `\advance\rownum\m@ne%`
10204   `\LetLtxMacro\LWR@save@xcolorrowHTMLcolor\LWR@xcolorrowHTMLcolor%`
10205   `\renewcommand*{\LWR@xcolorrowHTMLcolor}{}%`
10206   `\multicolumn{\value{LWR@tabletotalLaTeXcols}}{l}{#1} \\`
10207   `\LetLtxMacro\LWR@xcolorrowHTMLcolor\LWR@save@xcolorrowHTMLcolor%`
10208   `% \@rowc@lors%`
10209   `\LWR@getmynexttoken%`
10210 `}`

\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.

```
10211 \AtBeginDocument{
10212
10213 \IfPackageLoadedTF{lwarp-tabls}
10214 {
10215     \newcommand*{\LWR@HTMLhline}[1][]{%
10216         \ifbool{FormatWP}%
10217             {\LWR@docmidrule{1-\arabic{LWR@tabletotalLaTeXcols}}}%
10218             {\defaddtocounter{LWR@hlines}{1}}%
10219         \LWR@getmynexttoken}%
10220 }
10221 {
10222     \newcommand*{\LWR@HTMLhline}{%
10223         \ifbool{FormatWP}%
10224             {\LWR@docmidrule{1-\arabic{LWR@tabletotalLaTeXcols}}}%
10225             {\defaddtocounter{LWR@hlines}{1}}%
10226         \LWR@getmynexttoken}%
10227 }
10228
10229 }% AtBeginDocument
```

\LWR@HTMLcline {⟨columns⟩}

```
10230 \NewDocumentCommand{\LWR@HTMLcline}{m}%
10231 {%
10232     \LWR@docmidrule{#1}%
10233     \LWR@maybenewtablerow%
10234     \LWR@getmynexttoken%
10235 }%
```

\LWR@tabular@warpprintonly {⟨contents⟩}

Only process the contents if producing printed output. Modified inside a tabular
to grab the next token.

```
10236 \newcommand{\LWR@tabular@warpprintonly}[1]{%
10237     \ifbool{warpingprint}{#1}{}%
10238     \LWR@getmynexttoken%
10239 }
```

\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.

```
10240 \AtBeginDocument{
10241 \@ifundefined{NoAutoSpacing}%
10242 {% no babel-french
10243     \newcommand*{\LWR@nullifyNoAutoSpacing}{}
10244 }% no babel-french
10245 {% yes babel-french
10246     \newcommand*{\LWR@nullifyNoAutoSpacing}{%
10247         \NoAutoSpacing%
10248         \renewcommand*{\NoAutoSpacing}{}%
10249         \renewcommand*{\LWR@FBcancel}{}%
10250     }
```

```
10251 }% yes babel-french
10252 }% AtBeginDocument
```

tabular (*env.*) <direction> [⟨*vertposition*⟩] {⟨*colspecs*⟩}

The <direction> is from plext for Japanese documents, and is ignored.

```
10253 \StartDefiningTabulars
10254
10255 \NewDocumentCommand{\LWR@HTML@@tabular}{d<> o m}
10256 {%
10257     \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.

```
10258     \LWR@spanwarnformat{tabular}%
10259     \addtocounter{LWR@tabulardepth}{1}%
```

Not yet started a table row:

```
10260     \boolfalse{LWR@startedrow}%
```

Not yet doing any rules:

```
10261     \defcounter{LWR@hlines}{0}%
10262     \defcounter{LWR@hdashedlines}{0}%
10263     \boolfalse{LWR@doingtbrule}%
10264     \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.

```
10265     \LWR@nullifyNoAutoSpacing%
```

Have not yet found the end of tabular command. Unmute the @ and ! columns.

```
10266     \boolfalse{LWR@exitingtabular}%
10267     \boolfalse{LWR@tabularmutemods}%
```

Not adding final row for the lower border:

```
10268     \boolfalse{LWR@tabularfinalrow}%
```

Error if failed to use \mrowcell or \mcolrowcell when needed.

```
10269     \boolfalse{LWR@usedmultirow}%
10270     \boolfalse{LWR@foundmrowcell}%
```

In case of nesting:

```
10271    \renewcommand*{\LWR@multicoltext}{}%
10272    \booltrue{LWR@intabularmetadata}%
```

New PDF page, unless in a \multirow:

```
10273    \ifbool{LWR@in@multirow@par}%
10274        {\leavevmode\LWR@orignewline}%
10275        {\LWR@forcenewpage}%
```

In case of nesting, locally no longer in a \multirow:

```
10276    \boolfalse{LWR@in@multirow@par}%
```

Create the table tag:

```
10277    \LWR@htmlblocktag{table}%
```

Parse the table columns:

```
10278    \LWR@parsetablecols{#3}%
```

Table col spec is: \LWR@tablecolspec which is a string of llccrr, etc.

Do not place the table inside a paragraph:

```
10279    \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.

```
10280    \LWR@htmltag{tr style="display:none"}%
10281    \LWR@htmltag{th}.\LWR@htmltag{/th}%
10282    \LWR@htmltag{/tr}%
10283    \LWR@orignewline%
10284    \LWR@forceemptyline%
```

Track column #:

```
10285    \defcounter{LWR@tableLaTeXcolindex}{1}%
```

Have not yet added data in this column:

```
10286    \global\boolfalse{LWR@tabularcelladded}%
```

Start looking for midrules:

```
10287    \LWR@clearmidrules%
```

\\ becomes a macro to end the table row:

```
10288    \LetLtxMacro{\\}{\LWR@tabularendofline}%
```

\warpprintonly inside a tabular must grab the next token.

```
10289    \LetLtxMacro\warpprintonly\LWR@tabular@warpprintonly%
```

The following adjust for colortbl.

```
10290    \LetLtxMacro\arrayrulecolor\arrayrulecolornexttoken%
10291    \LetLtxMacro\doublerulesepcolor\doublerulesepcolornexttoken%
10292    \def\LWR@columnHTMLcolor{}%
10293    \def\LWR@rowHTMLcolor{}%
10294    \def\LWR@cellHTMLcolor{}%
10295    \@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.

```
10296    \ifdefvoid{\LWR@ruleHTMLcolor}%
10297        {\edef\LWR@vertruleHTMLcolor{black}}%
10298        {\edef\LWR@vertruleHTMLcolor{\LWR@origpound\LWR@ruleHTMLcolor}}%
```

Tracking the depth of cell color <div>s:

```
10299    \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.

```
10300    \LWR@traceinfo{LWR@@HTML@tabular: redefining macros}%
10301    \LetLtxMacro\noalign\LWR@tabularnoalign%
10302    \LetLtxMacro\hline\LWR@HTMLhline%
10303    \LetLtxMacro\cline\LWR@HTMLcline%

10304    \DeclareDocumentCommand{\hdashline}{o}{%
10305        \ifbool{FormatWP}%
10306            {\LWR@docdashline{1-\arabic{LWR@tabletotalLaTeXcols}}}%
10307            {\defaddtocounter{LWR@hdashedlines}{1}}%
10308        \LWR@getmynexttoken%
10309    }%

10310    \DeclareDocumentCommand{\cdashline}{m}{%
10311        \LWR@docdashline{##1}\LWR@getmynexttoken%
10312    }%

10313    \DeclareDocumentCommand{\firsthdashline}{o}{%
10314        \ifbool{FormatWP}%
10315            {\LWR@docdashline{1-\arabic{LWR@tabletotalLaTeXcols}}}%
10316            {\defaddtocounter{LWR@hdashedlines}{1}}%
10317        \LWR@getmynexttoken%
10318    }%

10319    \DeclareDocumentCommand{\lasthdashline}{o}{%
10320        \ifbool{FormatWP}%
10321            {\LWR@docdashline{1-\arabic{LWR@tabletotalLaTeXcols}}}%
10322            {\defaddtocounter{LWR@hdashedlines}{1}}%
10323        \LWR@getmynexttoken%
10324    }%
```

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.

```
10325    \renewcommand{\multicolumn}{\LWR@htmlmulticolumn}%
10326    \renewcommand*{\mrowcell}{%
10327        \LWR@maybenewtablerow%
10328        \LWR@tabularleftedge%
10329        \booltrue{LWR@skippingmrowcell}%
10330        \booltrue{LWR@foundmrowcell}%
10331    }%
10332    \renewcommand*{\mcolrowcell}{%
10333        \LWR@maybenewtablerow%
10334        \booltrue{LWR@skippingmcolrowcell}%
10335        \booltrue{LWR@foundmrowcell}%
10336    }%
10337    \LetLtxMacro\caption\LWR@longtabledatacaptiontag%
```

Reset for new processing:

```
10338    \boolfalse{LWR@tableparcell}%
10339    \boolfalse{LWR@skippingmrowcell}%
10340    \boolfalse{LWR@skippingmcolrowcell}%
10341    \boolfalse{LWR@skipatbang}%
10342    \boolfalse{LWR@emptyatbang}%
```

Set & for its special meaning inside the tabular:

```
10343    \StartDefiningTabulars%
10344    \protected\gdef&{\LWR@tabularampersand}%
```

Locally force any minipages to be fullwidth, until the end of the `tabular`:

```
10345    \booltrue{LWR@forceminipagefullwidth}%
```

Nest one level deeper of tabular paragraph handling:

```
10346    \addtocounter{LWR@tabularpardepth}{1}%
```

Look ahead for a possible table data cell:

```
10347    \LWR@traceinfo{LWR@@HTML@tabular: about to LWR@getmynexttoken}%
10348    \LWR@getmynexttoken%
10349 }%
```

Ending the environment:

```
10350 \newcommand*{\LWR@HTML@endtabular}
10351 {%
10352    \LWR@traceinfo{LWR@HTML@endtabular}%
```

Unnest one level of tabular paragraph handling:

```
10353    \addtocounter{LWR@tabularpardepth}{-1}%
```

Finish a row which is not yet done:

```
10354    \ifboolexpr{%
```

```
10355        test {%
10356            \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{<}%
10357                {\value{LWR@tabletotalLaTeXcols}}
10358        } or %
10359        (%
10360            bool{LWR@intabularmetadata} and%
10361            not bool{LWR@tabularcelladded} and%
10362            test {%
10363                \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{=}%
10364                    {\value{LWR@tabletotalLaTeXcols}}%
10365            }%
10366        )%
10367    }%
10368    {%


10369        \booltrue{LWR@tabularfinalrow}%
10370        \LWR@tabularfinishrow%
10371        \boolfalse{LWR@tabularfinalrow}%
10372    }%
10373    {%
10374        \LWR@closetabledatacell%
10375    }%
10376    \ifbool{LWR@startedrow}%
10377        {\LWR@htmltag{/tr}\LWR@orignewline}%
10378        {}%
```

xcolor row color support:

```
10379    \@rowc@lors%


10380    \LWR@htmlblocktag{/table}%
10381    \boolfalse{LWR@intabularmetadata}%
```

Unnest one level of tabular:

```
10382    \addtocounter{LWR@tabulardepth}{-1}%
```

Restore & to its usual meaning:

```
10383    \ifnumequal{\value{LWR@tabulardepth}}{0}{%
10384        \protected\gdef&{\LWR@origampmacro}%
10385        \StopDefiningTabulars%
10386    }{}%
```

Error if used \multirow or \multicolumnrow without using \mrowcell or \mcolrowcell.

```
10387    \ifbool{LWR@usedmultirow}{%
10388        \ifbool{LWR@foundmrowcell}%
10389            {\relax}%
10390            {%
10391                \PackageError{lwarp}%
10392                {%
10393                When using \protect\multirow, \protect\multicolumnrow,\MessageBreak
10394                    or the bigdelim package,\MessageBreak
10395                place \protect\mrowcell\space or \protect\mcolrowcell\MessageBreak
10396                    in empty cells which are to be skipped.\MessageBreak
10397                    See the Lwarp package documentation:\MessageBreak
```

```
10398                      "Special cases and limitations" -> "Tabular"
10399                }%
10400                {%
10401                      See the Lwarp package documentation:\MessageBreak
10402                      "Special cases and limitations" -> "Tabular".
10403                }%
10404            }%
10405     }{}%

10406     \LWR@traceinfo{LWR@HTML@endtabular finished}%
10407 }
10408
10409 \csletcs{LWR@HTML@endtabular*}{LWR@HTML@endtabular}
10410
10411 \StopDefiningTabulars
```

siunitx may redefine `tabular`, so set the following later:

```
10412 \AtBeginDocument{
10413     \LetLtxMacro\LWR@origendtabular\endtabular
10414     \csletcs{LWR@origendtabular*}{endtabular*}
10415     \LWR@formatted{@tabular}
10416     \LWR@formatted{endtabular}
10417     \LWR@formatted{endtabular*}
10418 }

10419 \end{warpHTML}
```

76   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.

10420 \begin{warpHTML}

76.1   Setup

`\@currentlabelname`   To remember the most recently defined section name, description, or caption, for `\nameref`.

```
10421 \def\@currentlabelname{\linkhomename}%
```

`\LWR@stripperiod` {⟨*text*⟩} [⟨.⟩]

Table 16: Cross-referencing data structures

Original LATEX: (print and HTML)

   \refstepcounter: Steps the couunter and sets \@currentlabel.

   \@currentlabel: \p@<ctr>\the<ctr> Updated by \refstepcounter.

   \label: Writes to the .aux file:
      \newlabel{<label>}{{\@currentlabel}{\thepage}}

   \newlabel: When the .aux file is read, sets \r@<label>.

   \r@<label>: Set to: {{\@currentlabel}{\thepage}}

   \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 76.3), and another .aux entry (section 76.2). If
      memoir is used, its \@mem@old@label points to lwarp's version, and cleveref
      patches.

   \newlabel: Unchanged. When the .aux file is read, sets \r@<label>@lwarp.

   \r@<label>@lwarp: Set to {{section_name}{file_name}{depth}{number}}:

      \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.

      \LWR@lateximagedepthref: The lateximagedepth for this label.

      \LWR@lateximagenumberref: The lateximagenumber for this label.

   \nameref: Emualted from hyperref for lwarp. See section 76.4.

   \ref and \nameref: Adds HTML tags. See section 76.4.

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 60.4 for use with footnotes.

   \label: 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 202.

   \label inside math: See section 83.7.1.

Footnotes: See \noteentry in section 60.4.

Removes a trailing period.

```
10422 \def\LWR@stripperiod#1.\ltx@empty#2\@nil{#1}%
```

\LWR@setlatestname {⟨*object name*⟩}

Removes \label, strips any final period, and remembers the result.

```
10423 \newcommand*{\LWR@setlatestname}[1]{%
```

Remove \label and other commands from the name, the strip any final period.
See gettitlestring.

```
10424    \GetTitleStringExpand{#1}%
10425    \edef\@currentlabelname{\detokenize\expandafter{\GetTitleStringResult}}%
10426    \edef\@currentlabelname{%
10427        \expandafter\LWR@stripperiod\@currentlabelname%
10428        \ltx@empty.\ltx@empty\@nil%
10429    }%
10430 }
```

76.2   New lwarp labels.

*_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>}}
```

See:
http://tex.stackexchange.com/questions/57194/
        [extract-section-number-from-equation-reference](http://tex.stackexchange.com/questions/57194/)

\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 cause the warning.

```
10431 \def\LWR@setref#1#2#3{%
10432    \ifx#1\relax%
10433        ??%
10434    \else%
10435        \expandafter#2#1%
10436    \fi}
```

\LWR@nameref {⟨*label*⟩} Returns the section name for this label:

```
10437 \newcommand*{\LWR@nameref}[1]{%
10438    \begingroup%
10439    \LWR@nullifyfootnotes%
10440    \expandafter\LWR@setref\csname r@#1@lwarp\endcsname\LWR@firstoffive{#1}%
10441    \endgroup%
10442 }
```

`\LWR@currentautosecpageref` {⟨*label*⟩} Returns the `LWR@currentautosecpage` for this label:

```
10443 \newcommand*{\LWR@currentautosecpageref}[1]{%
10444     \expandafter\LWR@setref\csname r@#1@lwarp\endcsname\LWR@secondoffive{#1}%
10445 }
```

`\LWR@htmlfileref` {⟨*label*⟩} Returns the file number or name for this label:

```
10446 \newcommand*{\LWR@htmlfileref}[1]{%
10447     \expandafter\LWR@setref\csname r@#1@lwarp\endcsname\LWR@thirdoffive{#1}%
10448 }
```

`\LWR@lateximagedepthref` {⟨*label*⟩} Returns the `lateximagedepth` for this label:

```
10449 \newcommand*{\LWR@lateximagedepthref}[1]{%
10450     \expandafter\LWR@setref\csname r@#1@lwarp\endcsname\LWR@fourthoffive{#1}%
10451 }
```

`\LWR@lateximagenumberref` {⟨*label*⟩} Returns the `lateximagenumber` for this label:

```
10452 \newcommand*{\LWR@lateximagenumberref}[1]{%
10453     \expandafter\LWR@setref\csname r@#1@lwarp\endcsname\LWR@fifthoffive{#1}%
10454 }
```

`\LWR@write@lwarplabel` {⟨*label*⟩} Sanitize the name and then creates the label:

```
10455 \newcommand*{\LWR@write@lwarplabel}[1]{%
10456     \LWR@traceinfo{LWR@write@lwarplabel !#1!}%
10457     \LWR@setlatestname{\@currentlabelname}%
10458        \@bsphack%
10459        \protected@write\@auxout{}%
10460            {%
10461                \string\newlabel{#1@lwarp}{%
10462                    {\@currentlabelname}%
10463                    {\theLWR@currentautosecpage}%
10464                    {%
10465                        \ifbool{FileSectionNames}%
10466                            {\LWR@thisfilename}%
10467                            {\arabic{LWR@htmlfilenumber}}%
10468                    }%
10469                    {\arabic{LWR@lateximagedepth}}%
10470                    {\arabic{LWR@lateximagenumber}}%
10471                }%
10472            }%
10473        \@esphack%
10474 }
```

76.3   Labels

`\LWR@label@subcreatetag` Creates the tag from `\LWR@sanitized`.

```
10475 \newcommand*{\LWR@label@subcreatetag}{%
10476     \LWR@htmltag{a \LWR@print@mbox{id=\textquotedbl\LWR@sanitized\textquotedbl}}%
10477        \LWR@htmltag{/a}%
10478 }
```

\LWR@label@inmathcomment

```
10479 \newcommand*{\LWR@label@inmathcomment}{%
10480     \ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
10481     {%
```

The combined LATEX & HTML label is printed in a \mbox field:

```
10482         \mbox{%
```

Shift the label over to the right side of the environment to avoid over-printing the math:

```
10483             \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.)

```
10484                 \LWR@htmlclosecomment%
10485                 \LWR@label@subcreatetag%
10486                 \LWR@htmlopencomment%
10487             }% mbox
10488     }% mathjax
10489     {%
10490         \LWR@label@subcreatetag%
10491     }%
10492 }
```

\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.

```
10493 \newcommand*{\LWR@label@createtag}[1]{%
10494     \LWR@traceinfo{LWR@label@createtag !#1!}%
```

Create an HTML id tag unless are inside a lateximage, since it would appear in the image:

```
10495     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
10496     {}%
10497     {% not lateximage
```

If not doing a lateximage, create an HTML ID tag.

```
10498         \LWR@sanitize{#1}%
10499         \ifbool{LWR@insidemathcomment}%
10500         {% inside HTML math comment
10501             \LWR@label@inmathcomment%
10502         }% inside HTML math comment
10503         {% not inside HTML math comment
10504             \ifbool{LWR@doingstartpars}%
10505             {% pars allowed
10506                 \ifbool{LWR@doingapar}%
10507                 {% par started
10508                     \LWR@label@subcreatetag%
```

```
10509                        }% par started
10510                        {% par not started
10511                            \LWR@stoppars%
10512                            \LWR@label@subcreatetag%
10513                            \LWR@startpars%
10514                        }% par not started
10515                    }% pars allowed
10516                    {% pars not allowed
10517                            \LWR@label@subcreatetag%
10518                    }% pars not allowed
10519             }% not inside HTML math comment
10520     }% not lateximage
10521 }
```

\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.

> If memoir is used, it's \@mem@old@label is pointed here.

> clevereref later encases this to add its own cross-referencing, and also patches
> memoir.

```
10522 \newcommand*{\LWR@new@label}[1]{%
10523     \LWR@traceinfo{LWR@new@label: starting}%
10524     \LWR@traceinfo{LWR@new@label: !#1!}%
10525 % \@bsphack%
```

> Create a traditional LATEX label, as modified by cleveref:

```
10526     \LWR@orig@label{#1}%
```

> Create a special label which holds the section number, section name, LWR@htmlfilenumber,
> LWR@lateximagedepth, and LWR@lateximagenumber:

```
10527     \LWR@traceinfo{%
10528         LWR@new@label: filesectionnames is %
10529         \ifbool{FileSectionNames}{true}{false}%
10530     }%
10531     \LWR@traceinfo{%
10532         LWR@new@label: LWR@thisfilename is !\LWR@thisfilename!%
10533     }%
10534     \LWR@traceinfo{%
10535         LWR@new@label: LWR@htmlfilenumber is \arabic{LWR@htmlfilenumber}%
10536     }%
10537     \LWR@write@lwarplabel{#1}%
10538     \LWR@label@createtag{#1}%
10539     % \@esphack%
10540     \LWR@traceinfo{LWR@new@label: done}%
10541 }
```

76.4   References

\LWR@addlinktitle

```
10542 \newcommand*{\LWR@addlinktitle}{%
10543     \ifdefvoid{\LWR@ThisAltText}{}{ % space
10544         title=\textquotedbl\LWR@ThisAltText\textquotedbl\ % space
10545         \gdef\LWR@ThisAltText{}%
10546     }%
10547 }
```

\LWR@startref {⟨*label*⟩}   (Common code for \ref and \nameref.)

Open an HTML tag reference to a filename, # character, and a label.

```
10548 \newcommand*{\LWR@startref}[1]
10549 {%
10550     \LWR@sanitize{#1}%
10551     \LWR@traceinfo{LWR@startref A: !#1!}%
```

Create the filename part of the link:

```
10552     \LWR@htmltag{a href=\textquotedbl%
10553     \LWR@traceinfo{LWR@startref B}%
10554     \LWR@print@mbox{\LWR@htmlrefsectionfilename{#1}}%
10555     \LWR@traceinfo{LWR@startref C}%
10556     \LWR@origpound%
```

Create the destination id:

See if LWR@lateximagedepth is unknown:

```
10557     \LWR@traceinfo{LWR@startref D: !#1!}%
10558     \ifcsundef{r@#1@lwarp}%
```

"??" if LWR@lateximagedepth is unknown, so create a link with an unknown destination:

```
10559     {%
10560         \LWR@traceinfo{LWR@startref D0: ??}%
10561         ??%
10562     }%
```

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.)

```
10563     {%
10564         \ifnumgreater{\LWR@lateximagedepthref{#1}}{0}%
10565             {%

10566                 lateximage-\BaseJobname-\LWR@lateximagenumberref{#1}%
10567             }%
10568             {%
10569                 \LWR@traceinfo{LWR@startref D3}%
```

\detokenize is used to allow underscores in the labels:

```
10570                    \LWR@print@mbox{\LWR@sanitized}%
10571              }%
10572        }%
10573        \LWR@traceinfo{LWR@startref E}%
```

Closing quote:

```
10574        \textquotedbl%
```

Maybe add a title:

```
10575        \LWR@addlinktitle%
10576        }%
10577        \LWR@traceinfo{LWR@startref F}%
10578 }
```

\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)".

```
10579 \NewDocumentCommand{\LWR@subnewref}{m m}{%
10580        \LWR@traceinfo{LWR@subnewref #1 #2}%
10581        \LWR@startref{#1}%
10582        \LWR@print@ref{#2}%
10583        \LWR@htmltag{/a}%
10584 }
```

\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:

```
10585 \NewDocumentCommand{\LWR@HTML@ref}{s m}{%
10586        \LWR@traceinfo{LWR@HTML@ref !#2!}%
10587        \IfBooleanTF{#1}%
10588              {\LWR@print@ref{#2}}%
10589              {\LWR@subnewref{#2}{#2}}%
10590 }
10591
10592 \LWR@formatted{ref}
10593
10594
10595 \NewDocumentCommand{\LWR@HTML@Ref}{s m}{%
10596        \LWR@traceinfo{LWR@HTML@Ref !#2!}%
10597        \IfBooleanTF{#1}%
10598              {\LWR@print@Ref{#2}}%
10599              {\LWR@subnewref{#2}{#2}}%
10600 }
10601
10602 \LWR@formatted{Ref}
```

\LWR@refwithsection * {⟨*label*⟩}

> Creates a reference, using the section number as the text. Used for back references.

```
10603 \NewDocumentCommand{\LWR@refwithsection}{s m}{%
10604     \LWR@traceinfo{LWR@refwithsection !#2!}%
```

> If starred, just use the text without a hyperlink:

```
10605     \IfBooleanTF{#1}%
10606       {\LWR@print@ref{\BaseJobname-autopage-\LWR@currentautosecpageref{#2}}}%
10607         {%
```

> Open the reference:

```
10608         \LWR@startref{#2}%
```

> Add the text of the link.

> Check for and handle an undefined reference:

```
10609         \edef\@tempa{\LWR@currentautosecpageref{#2}}%
10610         \ifdefstring{\@tempa}{??}%
10611           {??}%
```

> For a defined reference:

```
10612           {%
```

> Set \@tempa to \r@<label>, which is {section number}{page number}.

```
10613             \edef\@tempa{%
10614                 \csname
10615               r@\BaseJobname-autopage-\LWR@currentautosecpageref{#2}%
10616                 \endcsname%
10617             }%
```

> Check the section number alone:

```
10618             \edef\@tempa{\expandafter\@firstoftwo\@tempa}%
```

> If the reference has no section number print an asterisk:

```
10619             \expandafter\ifblank\expandafter{\@tempa}%
10620               {*}
```

> If there is a section number, print it:

```
10621                 {%
10622                   \LWR@print@ref{%
10623                 \BaseJobname-autopage-\LWR@currentautosecpageref{#2}%
10624                   }%
10625                 }%
10626             }%
```

> Close the reference:

```
10627              \LWR@htmltag{/a}%
10628          }%
10629 }
```

For MATHJAX:

```
10630 \CustomizeMathJax{\let\LWRref\ref}
10631 \CustomizeMathJax{\renewcommand{\ref}{\ifstar\LWRref\LWRref}}
```

\pagerefPageFor Text for page references.

```
10632 \newcommand*{\pagerefPageFor}{see }
```

\pageref * {⟨*label*⟩}   Create an internal document reference, or just the unlinked number
if starred, per hyperref.

```
10633 \NewDocumentCommand{\LWR@new@pageref}{s m}{%
10634     \IfBooleanTF{#1}%
10635         {(\pagerefPageFor\LWR@print@ref{#2})}%
10636         {(\cpageref{#2})}%
10637 }
```

\nameref {⟨*label*⟩}

```
10638 \newrobustcmd*{\nameref}[1]{%
10639     \LWR@traceinfo{nameref}%
10640     \LWR@startref{#1}%
10641     \LWR@traceinfo{nameref B}%
10642     \LWR@nameref{#1}%
10643     \LWR@traceinfo{nameref C}%
10644     \LWR@htmltag{/a}%
10645     \LWR@traceinfo{nameref: done}%
10646 }
```

\Nameref {⟨*label*⟩} In print, adds the page number. In HTML, does not.

```
10647 \LetLtxMacro\Nameref\nameref
```

76.5   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.

```
10648 % DO NOT TELL OTHER PACKAGES TO ASSUME HYPERREF, lest they attempt to patch it:
10649 % \EmulatesPackage{hyperref}[2015/08/01]% Disabled.  Do not do this.
```

Emulates hyperref:

\@currentHref Added to support backref.

```
10650 \AtBeginDocument{
10651     \def\@currentHref{\BaseJobname-autopage-\theLWR@previousautopagelabel}%
10652 }
```

\LWR@linkcatcodes Sets catcodes before processing macros which have hyperlinks as arguments.

```
10653 \newcommand*{\LWR@linkcatcodes}{%
10654     \catcode`\#=12%
10655     \catcode`\%=12%
10656     \catcode`\&=12%
10657     \catcode`\~=12%
10658     \catcode`\_=12%
```

For babel-french:

```
10659     \LWR@hook@processingtags%
10660 }
```

\LWR@linkmediacatcodes Sets catcodes before processing macros which have hyperlinks as arguments. Modified for multimedia links.

```
10661 \newcommand*{\LWR@linkmediacatcodes}{%
10662     \catcode`\#=12%
10663     \catcode`\%=12%
10664 %   \catcode`\&=12% left alone for splitting flash variables
10665     \catcode`\~=12%
10666     \catcode`\_=12%
```

For babel-french:

```
10667     \LWR@hook@processingtags%
10668 }
```

\LWR@subhyperref {⟨URL⟩}

Starts a link for \LWR@hrefb. A group must have been opened first, with nullified catcodes. The text name is printed afterwards, after the group is closed and catcodes restored.

```
10669 \NewDocumentCommand{\LWR@subhyperref}{m}{%
10670     \LWR@traceinfo{LWR@subhyperref !#1!}%
10671     \LWR@sanitize{#1}%
10672     \LWR@htmltag{%
10673         a href=\textquotedbl\LWR@sanitized\textquotedbl\ % space
10674         \LWR@addlinktitle % space
10675         target=\textquotedbl\_{}blank\textquotedbl\ % space
10676     }%
10677 }
```

\LWR@subhyperreftext {⟨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.

```
10678 \newcommand{\LWR@subhyperreftext}[1]{%
10679     #1%
10680     \LWR@htmltag{/a}%
10681     \LWR@ensuredoingapar%
10682 }
```

\LWR@subhyperrefclass {⟨URL⟩} {⟨text⟩} {⟨htmlclass⟩}

```
10683 \NewDocumentCommand{\LWR@subhyperrefclass}{m +m m}{%
10684     \LWR@htmltag{%
10685         a % space
10686       href=\textquotedbl\begingroup\@sanitize#1\endgroup\textquotedbl\ % space
10687         class=\textquotedbl#3\textquotedbl\ % space
10688         \LWR@addlinktitle % space
10689     }\LWR@orignewline%
10690     #2%
10691     \LWR@htmltag{/a}%
10692     \LWR@ensuredoingapar%
10693 }
```

\LWR@href [⟨options⟩] {⟨URL⟩} {⟨text⟩}

Create a link with accompanying text:

```
10694 \DeclareDocumentCommand{\LWR@hrefb}{O{} m}{%
10695     \LWR@ensuredoingapar%
10696     \LWR@subhyperref{#2}%
10697     \endgroup% restore catcodes
10698     \LWR@subhyperreftext%
10699 }
10700
10701 \newrobustcmd*{\LWR@href}{%
10702     \begingroup%
10703     \LWR@linkcatcodes%
10704     \LWR@hrefb%
10705 }
```

\LWR@nolinkurl {⟨URL⟩}

Print the name of the link without creating the link:

```
10706 \newcommand*{\LWR@nolinkurlb}[1]{%
10707     \LWR@ensuredoingapar%
10708     \def\LWR@templink{#1}%
10709     \@onelevel@sanitize\LWR@templink%
10710     \LWR@templink%
10711     \endgroup%
10712 }
10713
10714 \newrobustcmd*{\LWR@nolinkurl}{%
10715     \begingroup%
10716     \LWR@linkcatcodes%
10717     \LWR@nolinkurlb%
10718 }
```

\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 redefined by lwarp-url.

```
10719 \DeclareDocumentCommand{\LWR@urlb}{m}{%
10720     \LWR@ensuredoingapar%
10721     \def\LWR@templink{#1}%
10722     \@onelevel@sanitize\LWR@templink%
10723     \LWR@href{\LWR@templink}{\LWR@templink}%
10724     \endgroup%
10725 }
10726
10727 \newrobustcmd*{\LWR@url}{%
10728     \begingroup%
10729     \LWR@linkcatcodes%
10730     \LWR@urlb%
10731 }
```

\LWR@subinlineimage {⟨*1: <alt> tag*⟩} {⟨*2: class*⟩} {⟨*3: filename*⟩} {⟨*4: extension*⟩} {⟨*5: css style*⟩} {⟨*6: aria role*⟩}

Factored from `lateximage`.

```
10732 \newcommand*{\LWR@subinlineimage}[6]{%
10733     \ifblank{#6}%
10734         {\renewcommand*{\LWR@tempone}{}}%
10735         {\renewcommand*{\LWR@tempone}{role="#6"\LWR@indentHTML}}%
10736     \ifblank{#1}%
10737     {%
10738         \LWR@htmltag{img \LWR@indentHTML
10739             src=\textquotedbl#3.#4\textquotedbl \LWR@indentHTML
10740             alt=\textquotedbl#3\textquotedbl \LWR@indentHTML
10741             \LWR@tempone
10742             style=\textquotedbl#5\textquotedbl \LWR@indentHTML
10743             class=\textquotedbl#2\textquotedbl \LWR@orignewline
10744         }%
10745     }%
10746     {%
10747         \LWR@htmltag{img \LWR@indentHTML
10748             src=\textquotedbl#3.#4\textquotedbl \LWR@indentHTML
10749             alt=\textquotedbl#1\textquotedbl \LWR@indentHTML
10750             \LWR@tempone
10751             style=\textquotedbl#5\textquotedbl \LWR@indentHTML
10752             class=\textquotedbl#2\textquotedbl \LWR@orignewline
10753         }%
10754     }%
10755 }
```

```
10756 \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.

77 Floats

Floats are supported, although partially through emulation.

Table 17 shows the data structure associated with each <type> of float.

77.1 Float environment

for HTML output: 10757 \begin{warpHTML}

\LWR@floatbegin {⟨*type*⟩} [⟨*placement*⟩]   Begins a \newfloat environment.

10758 \NewDocumentCommand{\LWR@floatbegin}{m o}{%

Warn if starting a float inside a <span>:

10759   \LWR@spanwarninvalid{float}%

10760   \ifbool{FormatWP}{\newline}{}%
10761   \LWR@stoppars%

There is a new float, so increment the unique float counter:

```
10762    \addtocounter{LWR@thisautoid}{1}%
10763    \booltrue{LWR@freezethisautoid}%
```

```
10764    \begingroup%
```

Settings while inside the environment:

```
10765    \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.

```
10766    \LWR@htmltag{%
10767        figure id=\textquotedbl%
10768            \LWR@print@mbox{autoid-\arabic{LWR@thisautoid}}%
10769        \textquotedbl\ % space
10770        class=\textquotedbl#1 \@nameuse{LWR@floatstyle@#1}\textquotedbl%
10771    }%
10772    \ifbool{FormatWP}{%
10773        \LWR@orignewline%
10774        \LWR@BlockClassWP{}{}{wp#1}%
10775    }{}%
```

Update the caption type:

```
10776    \renewcommand*{\@captype}{#1}%
```

Mark the float for a word processor conversion:

```
10777    \LWR@startpars%
10778    \ifboolexpr{bool{FormatWP} and bool{WPMarkFloats}}{%
10779
10780    === begin #1 ===
10781
10782    }{}%
```

After each \LWR@floatbegin, look for \centering, etc next, using \LWR@floatalignment.

```
10783 }
```

For koma-script. The following does not work for tables.

```
10784 \AtBeginDocument{
10785
10786 \IfPackageLoadedTF{tocbasic}{
10787
10788 \appto\figure@atbegin{%
10789    \LWR@futurenonspacelet\LWR@mynexttoken\LWR@floatalignment%
10790 }
10791
10792 }{}% tocbasic
10793
10794 }% AtBeginDocument
```

\@xfloat Support packages which create floats directly.

\@xdlbfloat

Look for \centering, etc using \LWR@floatalignment.

```
10795 \AtBeginDocument{
10796     \def\@xfloat #1[#2]{%
10797         \LWR@floatbegin{#1}[#2]
10798         \LWR@futurenonspacelet\LWR@mynexttoken\LWR@floatalignment%
10799     }
10800     \def\@xdblfloat #1[#2]{%
10801         \LWR@floatbegin{#1}[#2]
10802         \LWR@futurenonspacelet\LWR@mynexttoken\LWR@floatalignment%
10803     }
10804 }
```

\LWR@floatend Ends a \newfloat environment.

```
10805 \newcommand*{\LWR@floatend}{%
```

If saw a \centering, finish the center environment:

```
10806     \LWR@endfloatalignment%
```

Mark the float end for a word processor conversion:

```
10807     \ifboolexpr{bool{FormatWP} and bool{WPMarkFloats}}{%
10808
10809     === end ===
10810
10811     }{}%
10812     \LWR@stoppars%
```

Close an HTML figure tag:

```
10813     \ifbool{FormatWP}{\endLWR@BlockClassWP}{}%
10814     \LWR@htmlelementend{figure}%
10815     \endgroup%
10816     \boolfalse{LWR@freezethisautoid}%
10817     \LWR@startpars%
10818     \ifbool{FormatWP}{\newline}{}%
10819 }
```

\end@float Support packages which create floats directly.

\end@dlbfloat

```
10820 \AtBeginDocument{
10821     \let\end@float\LWR@floatend
10822     \let\end@dblfloat\LWR@floatend
10823 }
```

77.2 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.

```
10824 \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.

```
10825 \newcounter{LWR@thisautoidWP}
```

LWR@freezethisautoid (*bool*)   Prevents multiple increments of \LWR@thisautoid inside a float.

```
10826 \newbool{LWR@freezethisautoid}
10827 \boolfalse{LWR@freezethisautoid}
```

\LWR@forcenewautoidanchor Adds a new <autoid> anchor.

```
10828 \newcommand*{\LWR@forcenewautoidanchor}{%
10829     \addtocounter{LWR@thisautoid}{1}%
10830     \ifbool{LWR@doingapar}%
10831     {%
10832         \LWR@htmltag{a id=\textquotedbl%
10833             \LWR@print@mbox{autoid-\arabic{LWR@thisautoid}}%
10834             \textquotedbl\ }% space
10835         \LWR@htmltag{/a }%
10836     }%
10837     {%
10838         \LWR@stoppars%
10839         \LWR@htmltag{a id=\textquotedbl%
10840             \LWR@print@mbox{autoid-\arabic{LWR@thisautoid}}%
10841         \textquotedbl\ }% space
10842         \LWR@htmltag{/a }%
10843         \LWR@startpars%
10844     }%
10845 }
```

\LWR@newautoidanchor Sometimes adds a new <autoid> anchor.

```
10846 \newcommand*{\LWR@newautoidanchor}{%
10847     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
10848         {}%
10849         {\ifbool{LWR@freezethisautoid}{}{\LWR@forcenewautoidanchor}}%
10850 }
```

\@captype   Remembers which float type is in use.

```
10851 \newcommand*{\@captype}{}
```

\LWR@floatalignmentname Set to center, flushleft, or flushright if saw \centering, \raggedright, or \raggedleft.

```
10852 \newcommand*{\LWR@floatalignmentname}{}
```

\LWR@floatalignment If sees a \centering, \raggedleft, or \raggedright, creates a center, flushright, or flushleft environment.

```
10853 \newcommand*{\LWR@floatalignment}{%
10854     \ifdefstrequal{\LWR@mynexttoken}{\centering}{%
10855         \center%
10856         \renewcommand*{\LWR@floatalignmentname}{center}%
10857     }{}%
```

```
10858        \ifdefstrequal{\LWR@mynexttoken}{\raggedright}{%
10859            \flushleft%
10860            \renewcommand*{\LWR@floatalignmentname}{flushleft}%
10861        }{}%
10862        \ifdefstrequal{\LWR@mynexttoken}{\raggedleft}{%
10863            \flushright%
10864            \renewcommand*{\LWR@floatalignmentname}{flushright}%
10865        }{}%
10866 }
```

\LWR@endfloatalignment Closes an environment from \LWR@floatalignment.

```
10867 \newcommand*{\LWR@endfloatalignment}{%
10868    \ifdefvoid{\LWR@floatalignmentname}%
10869        {}%
10870        {\@nameuse{end\LWR@floatalignmentname}}%
10871    \renewcommand*{\LWR@floatalignmentname}{}%
10872 }
```

77.3   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.

```
10873 \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.

```
10874 \AtBeginDocument{
10875    \IfPackageLoadedTF{caption}{}{
10876        \let\LWR@orig@caption\@caption%
10877        \long\def\@caption#1[#2]{%
```

Warn if using a caption inside a <span>:

```
10878            \LWR@spanwarnformat{caption}%

10879            \LWR@setlatestname{#2}%
10880            \LWR@orig@caption{#1}[{#2}]% also takes third argument
10881        }%
10882
10883        \renewcommand{\@makecaption}[2]{%
10884            \LWR@traceinfo{@makecaption}%
10885            \caption@begin{\@captype}%
10886            \LWR@isolate{#1}%
10887            \edef\LWR@tempone{#1}%
10888            \ifdefvoid{\LWR@tempone}{}{\CaptionSeparator}%
10889            \LWR@isolate{#2}%
```

```
10890          \caption@end%
10891          \LWR@traceinfo{@makecaption: done}%
10892       }%
10893    }
10894 }
```

77.4 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.

```
10895 \newcounter{LWR@nextautoid}
10896 \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.

```
10897 \newcommand*{\LWRsetnextfloat}[2]{%
10898    \setcounter{LWR@nextautopage}{#1}%
10899    \setcounter{LWR@nextautoid}{#2}%
10900 }
```

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.

```
10901 \newenvironment*{LWR@figcaption}
10902    {%
10903       \ifbool{FormatWP}{%
10904          \BlockClass[font-style:italic]{figurecaption}%
10905       }{%
10906          \BlockClass{figurecaption}%
10907       }%
```

Inside the caption, temporarily prevent underfull \hbox warnings, such as when the caption contains a math SVG image.

```
10908          \hbadness=10000\relax%
```

```
10909    }%
10910    {\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.

```
10911 \newcommand*{\LWR@HTML@caption@begin}[1]
10912 {%
10913     \LWR@traceinfo{LWR@HTML@caption@begin}%
```

> Keep par and minipage changes local:

```
10914     \begingroup%
```

> No need for a minipage or \parbox inside the caption:

```
10915     \RenewDocumentEnvironment{minipage}{O{t} o O{t} m}{}{}%
10916     \RenewDocumentCommand{\parbox}{O{t} O{} O{t} m +m}{##5}%
```

> Enclose the original caption code inside an HTML tag:

```
10917     \LWR@figcaption%
10918     \LWR@traceinfo{LWR@HTML@caption@begin: about to LWR@origcaption@begin}%
10919     \LWR@print@caption@begin{#1}%
10920     \LWR@traceinfo{LWR@HTML@caption@begin: done}%
10921 }
```

\LWR@HTML@caption@end Low-level patches to create HTML tags for captions.

```
10922 \newcommand*{\LWR@HTML@caption@end}
10923 {%
10924     \LWR@traceinfo{LWR@HTML@caption@end}%
10925     \LWR@print@caption@end%
```

> Closing tag:

```
10926     \endLWR@figcaption%
10927     \endgroup%
10928     % \leavevmode% avoid bad space factor (0) error
10929     \LWR@traceinfo{LWR@HTML@caption@end: done}%
10930 }
```

\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.

```
10931 \AtBeginDocument{
10932     \providecommand{\caption@begin}[1]{}
10933     \LWR@formatted{caption@begin}
10934
10935     \providecommand{\caption@end}{}
10936     \LWR@formatted{caption@end}
10937 }
```

\captionlistentry Tracks the float number for this caption used outside a float. Patched to create an
HTML anchor.

```
10938 \AtBeginDocument{%
10939 \IfPackageLoadedTF{caption}{
10940     \let\LWR@origcaptionlistentry\captionlistentry
10941
10942     \renewcommand*{\captionlistentry}{%
10943         \LWR@ensuredoingapar%
10944         \LWR@origcaptionlistentry%
10945     }
10946
10946     \def\LWR@LTcaptionlistentry{%
10947         \LWR@ensuredoingapar%
10948         \LWR@forcenewautoidanchor%
10949         \bgroup%
10950         \@ifstar{\egroup\LWR@LT@captionlistentry}% gobble *
10951             {\egroup\LWR@LT@captionlistentry}%
10952     }%
10953
10954     \def\LWR@LT@captionlistentry#1{%
10955         \caption@listentry\@firstoftwo[\LTcaptype]{#1}%
10956     }%
10957 }% caption loaded
10958 {% caption not loaded
10959     \newcommand{\captionlistentry}[2][]{}%
10960     \newcommand{\LWR@LT@captionlistentry}[2][]{}%
10961 }
10962 }% 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

```
10963 \let\LWR@origaddcontentsline\addcontentsline
10964
10965 \renewcommand*{\addcontentsline}[3]{%
10966     \ifstrequal{#1}{toc}{}{% not TOC
10967     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
10968         {}%
10969         {\LWR@newautoidanchor}%
10970     \ifcsvoid{ext@#2}{\csdef{ext@#2}{#1}}{}%
10971     \addtocontents{\@nameuse{ext@#2}}{%
10972         \protect\LWRsetnextfloat%
10973         {\arabic{LWR@currentautosecfloatpage}}%
10974         {\arabic{LWR@thisautoid}}%
10975     }%
10976     }% not TOC
10977     \LWR@origaddcontentsline{#1}{#2}{#3}%
10978 }
```

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.

```
10979 \RequirePackage{capt-of}
10980
10981 \AtBeginDocument{
10982     \let\LWR@origcaptionof\captionof
10983
10984     \renewcommand*{\captionof}{%
10985         \LWR@stoppars%
10986         \LWR@origcaptionof%
10987     }
10988 }% AtBeginDocument
```

```
10989 \end{warpHTML}
```

78   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 gener-
ate HTML output.

for HTML output `10990 \begin{warpHTML}`

78.1   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.

```
10991 \newcommand*{\LWR@myshorttoc}[1]{%
10992     \LWR@traceinfo{LWR@myshorttoc: #1}%
```

Only if the file exists:

```
10993     \IfFileExists{\jobname.#1}{%
10994         \LWR@traceinfo{LWR@myshorttoc: loading}%
```

⚠   Many of the commands in the file will have @ characters in them, so @ must be

made a regular letter.

```
10995        \begingroup%
10996        \makeatletter%
```

Disable \ref to avoid nested HTML references.

```
10997        \LetLtxMacro\ref\LWR@print@ref%
10998        \LWR@disablepinyin%
```

Read in the TOC file:

```
10999        \@input{\jobname.#1}%
11000        \endgroup%
11001    }%
11002    {}%
11003    \LWR@traceinfo{LWR@myshorttoc: done}%
11004 }
```

\LWR@subtableofcontents {⟨*toc/lof/lot*⟩} {⟨*sectionstarname*⟩}

Places a TOC/LOF/LOT at the current position.

```
11005 \NewDocumentCommand{\LWR@subtableofcontents}{m m}{%
```

Closes previous levels:

```
11006    \@ifundefined{chapter}%
11007        {\LWR@closeprevious{section}}%
11008        {\LWR@closeprevious{chapter}}%
```

Prints any pending footnotes so that they appear above the potentially large TOC:

```
11009    \LWR@printpendingfootnotes%
```

Place the list into its own chapter (if defined) or section:

```
11010    \@ifundefined{chapter}{\section*{#2}}{\chapter*{#2}}%
```

Create a new HTML nav containing the TOC/LOF/LOT:

```
11011    \LWR@htmlelementclass{nav}{#1}%
```

Create the actual list:

```
11012    \LWR@myshorttoc{#1}%
```

Close the nav:

```
11013    \LWR@htmlelementclassend{nav}{#1}%
11014 }
```

\@starttoc {⟨*ext*⟩}

Patch \@starttoc to encapsulate the TOC inside HTML tags:

```
11015 \let\LWR@orig@starttoc\@starttoc
11016
11017 \renewcommand{\@starttoc}[1]{
11018     \LWR@htmlelementclass{nav}{#1}%
11019     \LWR@orig@starttoc{#1}%
11020     \LWR@htmlelementclassend{nav}{#1}%
11021 }
```

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.)

```
11022 \newbool{LWR@copiedsidetoc}
11023 \boolfalse{LWR@copiedsidetoc}
```

\tableofcontents Patch \tableofcontents, etc. to print footnotes first. **newfloat** uses \listoffigures for all future float types.

```
11024 \AtBeginDocument{
11025
11026 \LetLtxMacro\LWR@origtableofcontents\tableofcontents
11027
11028 \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:

```
11029     \ifboolexpr{bool{FormatWP} and bool{WPMarkTOC}}{
11030
11031     === table of contents ===
11032
11033     }
11034     {
```

Copy the .toc file to .sidetoc for printing the sidetoc. The original .toc file is renewed when \tableofcontents is finished.

```
11035         \ifbool{LWR@copiedsidetoc}{}{%
11036             \LWR@copyfile{\jobname.toc}{\jobname.sidetoc}%
11037             \booltrue{LWR@copiedsidetoc}%
11038         }%
11039         \LWR@printpendingfootnotes
```

Disable \ref to avoid nested html references.

```
11040         \begingroup%
11041         \LetLtxMacro\ref\LWR@print@ref%
11042         \LWR@disablepinyin%
11043         \LWR@origtableofcontents%
11044         \endgroup%
11045     }
11046 }% \tableofcontents
11047
11048 }% \AtBeginDocument
```

\listoffigures

```
11049 \let\LWR@origlistoffigures\listoffigures
11050
11051 \renewcommand*{\listoffigures}{
11052     \ifboolexpr{bool{FormatWP} and bool{WPMarkLOFT}}{
11053
11054     === list of figures ===
11055
11056     }
11057     {
11058         \LWR@printpendingfootnotes
```

Disable \ref to avoid nested HTML references.

```
11059         \begingroup%
11060         \LetLtxMacro\ref\LWR@print@ref%
11061         \LWR@disablepinyin%
11062         \LWR@origlistoffigures%
11063         \endgroup%
11064     }
11065 }
```

\listoftables

```
11066 \let\LWR@origlistoftables\listoftables
11067
11068 \renewcommand*{\listoftables}{
11069     \ifboolexpr{bool{FormatWP} and bool{WPMarkLOFT}}{
11070
11071     === list of tables ===
11072
11073     }
11074     {
11075         \LWR@printpendingfootnotes
```

Disable \ref to avoid nested HTML references.

```
11076         \begingroup%
11077         \LetLtxMacro\ref\LWR@print@ref%
11078         \LWR@disablepinyin%
11079         \LWR@origlistoftables%
11080         \endgroup%
11081     }
11082 }
```

78.2   TOC commands

\LWR@listof {⟨*type*⟩} {⟨*title*⟩}

Emulate the \listof command from the float package (section 276). Used to create lists of custom float types. Also used to redefine the standard LaTeX \listoffigures and \listoftables commands, and in tocloft and memoir.

```
11083 \NewDocumentCommand{\LWR@listof}{m +m}{%
11084     \@ifundefined{l@#1}{%
```

```
11085        \csdef{l@#1}##1##2{\hypertocfloat{1}{#1}{\@nameuse{ext@#1}}{##1}{##2}}%
11086      }{}%
11087      \LWR@subtableofcontents{\@nameuse{ext@#1}}{#2}%
11088      \expandafter\newwrite\csname tf@\csname ext@#1\endcsname\endcsname%
11089      \immediate\openout \csname tf@\csname ext@#1\endcsname\endcsname%
11090          \jobname.\@nameuse{ext@#1}\relax%
11091 }
```

78.3   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.

```
11092 \end{warpHTML}
```

for HTML & PRINT: `11093 \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.

```
11094 \newcounter{SideTOCDepth}
11095 \setcounter{SideTOCDepth}{1}
11096
11097 \AtEndDocument{%
11098     \ifnumcomp{\value{SideTOCDepth}}{<}{\value{FileDepth}}{
11099         \PackageWarningNoLine{lwarp}
11100         {%
11101             SideTOCDepth is less than FileDepth,\MessageBreak
11102             so some website pages may be inaccessible%
11103         }
11104     }{}
11105 }
```

\sidetocname Holds the default name for the sidetoc.

```
11106 \newcommand{\sidetocname}{Contents}
```

```
11107 \end{warpall}
```

for HTML output: `11108 \begin{warpHTML}`

\LWR@sidetoc Creates the actual side-TOC.

```
11109 \newcommand*{\LWR@sidetoc}{%
```

```
11110    \LWR@forcenewpage
11111    \LWR@stoppars
11112
```

The entire sideтoc is placed into a nav of class sidetoc.

```
11113    \LWR@htmlelementclass{div}{sidetoccontainer}
11114    \LWR@htmlelementclass{nav}{sidetoc}
11115
11116    \setcounter{tocdepth}{\value{SideTOCDepth}}
11117
```

The title is placed into a <div> of class sidetoctitle, and may contain paragraphs.

```
11118    \begin{BlockClass}{sidetoctitle}
11119    \ifcsvoid{thetitle}{}{\InlineClass{sidetocthetitle}{\thetitle}\par}
11120    \sidetocname
11121    \end{BlockClass}
```

The table of contents is placed into a <div> of class sidetoccontents.

```
11122    \begin{BlockClass}{sidetoccontents}
11123    \LinkHome
11124
11125    \LWR@myshorttoc{sidetoc}
11126    \end{BlockClass}
11127    \LWR@htmlelementclassend{nav}{sidetoc}
11128    \LWR@htmlelementclassend{div}{sidetoccontainer}
11129 }
```

78.4   Low-level тoc line formatting

\numberline {⟨*number*⟩}

(Called from each line in the .aux, .lof files.)

Record this section number for further use:

```
11130 \newcommand*{\LWR@numberline}[1]{%
11131    \LWR@sectionnumber{#1}\quad%
11132 }
11133
11134 \LetLtxMacro\numberline\LWR@numberline
```

\LWR@maybetocdata   Replaced by tocdata. Adds author name.

```
11135 \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 is depth

#2 is section, subsection, etc.

#3 the text of the caption

#4 page number

```
11136 \NewDocumentCommand{\hypertoc}{m m +m m}{%
11137     \LWR@traceinfo{hypertoc !#1!#2!#3!#4!}%
```

Respond to tocdepth:

```
11138     \ifnumcomp{#1}{>}{\value{tocdepth}}%
11139         {}%
11140         {%
11141             \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.

```
11142             \LWR@subhyperrefclass{%
11143                 \LWR@htmlrefsectionfilename{\BaseJobname-autopage-#4}%
11144                     \LWR@origpound\LWR@print@mbox{autosec-#4}%
11145             }{#3}{toc#2}%


11146             \LWR@maybetocdata%


11147             \LWR@stoppars%
11148         }%
11149     \LWR@traceinfo{hypertoc done}%
11150 }
```

lofdepth (*Ctr*)   TOC depth for figures.

```
11151 \IfClassLoadedTF{memoir}{}{
11152     \newcounter{lofdepth}
11153     \setcounter{lofdepth}{1}
11154 }
```

lotdepth (*Ctr*)   TOC depth for tables.

```
11155 \IfClassLoadedTF{memoir}{}{
11156     \newcounter{lotdepth}
11157     \setcounter{lotdepth}{1}
11158 }
```

\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

```
11159 \newcommand{\hypertocfloat}[5]{%
```

If some float-creation package has not yet defined the float type's lofdepth counter, etc, define it here:

```
11160     \@ifundefined{c@#3depth}{%
11161         \newcounter{#3depth}%
11162         \setcounter{#3depth}{1}%
11163     }{}%
```

Respond to lofdepth, etc.:

```
11164     \LWR@traceinfo{hypertocfloat depth is #1 #3depth is \arabic{#3depth}}%
11165     \ifthenelse{\cnttest{#1}{<=}{\arabic{#3depth}}}%
11166         {%
11167             \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.

```
11168             \LWR@subhyperrefclass{%
11169             \LWR@htmlrefsectionfilename{%
11170                 \BaseJobname-autopage-\arabic{LWR@nextautopage}%
11171             }%
11172             \LWR@origpound\LWR@print@mbox{autoid-\arabic{LWR@nextautoid}}}%
11173             {#4}{toc#2}%


11174             \LWR@maybetocdata%


11175             \LWR@stoppars%
11176         }%
11177         {}%
11178 }
```

Automatically called by \contentsline:

\l@book {⟨*name*⟩} {⟨*page*⟩}

Uses \DeclareDocumentCommand in case the class does not happen to have a \book.

```
11179 \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.

```
11180 \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.

```
11181 \@ifundefined{chapter}
11182 {}
11183 {
11184 \DeclareDocumentCommand{\l@chapter}{m m}
11185     {\hypertoc{0}{chapter}{#1}{#2}}
11186 }
```

\l@section {⟨*name*⟩} {⟨*page*⟩}

```
11187 \renewcommand{\l@section}[2]{\hypertoc{1}{section}{#1}{#2}}
```

\l@subsection {⟨*name*⟩} {⟨*page*⟩}

```
11188 \renewcommand{\l@subsection}[2]{\hypertoc{2}{subsection}{#1}{#2}}
```

\l@subsubsection {⟨*name*⟩} {⟨*page*⟩}

```
11189 \renewcommand{\l@subsubsection}[2]{\hypertoc{3}{subsubsection}{#1}{#2}}
```

\l@paragraph {⟨*name*⟩} {⟨*page*⟩}

```
11190 \renewcommand{\l@paragraph}[2]{\hypertoc{4}{paragraph}{#1}{#2}}
```

\l@subparagraph {⟨*name*⟩} {⟨*page*⟩}

```
11191 \renewcommand{\l@subparagraph}[2]{\hypertoc{5}{subparagraph}{#1}{#2}}
```

\l@figure {⟨*name*⟩} {⟨*page*⟩}

```
11192 \renewcommand{\l@figure}[2]{\hypertocfloat{1}{figure}{lof}{#1}{#2}}
```

\l@table {⟨*name*⟩} {⟨*page*⟩}

```
11193 \renewcommand{\l@table}[2]{\hypertocfloat{1}{table}{lot}{#1}{#2}}
```

```
11194 \end{warpHTML}
```

79 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.

11195 `\begin{warpHTML}`

```
11196 \newcounter{LWR@autoindex}
11197 \setcounter{LWR@autoindex}{0}
11198
11199 \newcounter{LWR@autoglossary}
11200 \setcounter{LWR@autoglossary}{0}
```

`\IndexPageSeparator`   User-adjustable delimiters for page and range separators in the `*.ind` files.
`\IndexRangeSeparator`

```
11201 \newcommand*{\IndexPageSeparator}{, }
11202 \newcommand*{\IndexRangeSeparator}{--}
```

theindex (*env.*)

```
11203 \@ifundefined{chapter}
11204     {\newcommand*{\LWR@indexsection}[1]{\section*{#1}}}
11205     {\newcommand*{\LWR@indexsection}[1]{\chapter*{#1}}}
11206
11207
11208 \AtBeginDocument{
11209
11210 \renewenvironment*{theindex}{%
11211     \LWR@indexsection{\indexname}%
11212     \LetLtxMacro\item\LWR@indexitem%
11213     \LetLtxMacro\subitem\LWR@indexsubitem%
11214     \LetLtxMacro\subsubitem\LWR@indexsubsubitem%
11215 }{}
11216
11217 }% AtBeginDocument
```

`\LWR@indexitem` [⟨*index key*⟩]   The optional argument is added to support repeatindex.

```
11218 \newcommand{\LWR@indexitem}[1][\@empty]{
11219
11220     \InlineClass{indexitem}{\LWR@htmlcomment{}}#1%
11221 }
```

`\LWR@indexsubitem`

```
11222 \newcommand{\LWR@indexsubitem}{
11223
11224     \InlineClass{indexsubitem}{\LWR@htmlcomment{}}%
11225 }
```

`\LWR@indexsubsubitem`

```
11226 \newcommand{\LWR@indexsubsubitem}{
11227
11228     \InlineClass{indexsubsubitem}{\LWR@htmlcomment{}}%
11229 }
```

\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.

```
11230 \NewDocumentCommand{\LWR@xindex@modifyentry}{>{\SplitArgument{1}{|}}m}
11231    {\LWR@xindex@modifyentrysub#1}
```

Handle left and right parenthesis range argument, or add a hyperindexformat
clause.

```
11232 \newcommand*{\LWR@xindex@modifyentrysub}[2]{%
11233    \edef\LWR@tempone{#1}%
11234    \edef\LWR@temptwo{#2}%
11235    \IfValueTF{#2}{%
11236        \ifx#2(%
11237            \appto\LWR@tempone{|(}%
11238        \else%
11239            \ifx#2)%
11240                \appto\LWR@tempone{|)}%
11241            \else%
11242                \appto\LWR@tempone{%
11243                    |hyperindexformat\LWRleftbrace%
11244                    \LWRbackslash#2%
11245                    \LWRrightbrace%
11246                }%
11247            \fi%
11248        \fi%
11249    }%
11250    {}%
11251 }
```

LWR@xindex@tricked (*bool*)   Used to track xindex creation. See next.

```
11252 \newbool{LWR@xindex@tricked}
11253 \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.

```
11254 \def\LWR@wrindex#1{%
11255    \ifbool{LWR@xindex}{%
11256        \ifbool{LWR@xindex@tricked}{}{%
11257            \protected@write\@indexfile{}%
11258            {%
11259                \LWRpercent\space hyperpage\LWRrightbrace%
11260                \LWRpercent\space trick xindex to assume hyperref%
11261            }%
11262            \global\booltrue{LWR@xindex@tricked}%
11263        }%
11264        \LWR@xindex@modifyentry{#1}%
11265    }{%
11266        \def\LWR@tempone{#1}%
11267    }%
11268    \addtocounter{LWR@autoindex}{1}%
```

```
11269     \protected@write\@indexfile{}%
11270 {\string\indexentry{\LWR@tempone}{\arabic{LWR@autoindex}}}}%
```

The label is assigned after the file write to avoid conflict with cleveref.

```
11271     \label{LWRindex-\arabic{LWR@autoindex}}%
11272     \endgroup%
11273     \@esphack%
11274 }
11275
11276 \AtBeginDocument{
11277 \let\@wrindex\LWR@wrindex
11278 }
```

\@wrglossary {⟨*term*⟩}   Redefined to write the LWR@autoglossary counter instead of page.

```
11279 \def\@wrglossary#1{%
11280     \addtocounter{LWR@autoglossary}{1}%
11281     \LWR@new@label{LWRglossary-\theLWR@autoglossary}%
11282     \protected@write\@glossaryfile{}%
11283     {\string\glossaryentry{#1}{\theLWR@autoglossary}}%
11284     \endgroup%
11285     \@esphack%
11286 }
```

\LWR@indexnameref@anonref {⟨*LWR@autoindex*⟩}

Displays a reference link where there no \ref available.

```
11287 \newcommand*{\LWR@indexnameref@anonref}[1]{%
11288     \LWR@startref{LWRindex-#1}%
11289     (*)%
11290     \LWR@htmltag{/a}%
11291 }
```

\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.

```
11292 \newcommand*{\LWR@indexnameref@ref}[1]{%
11293     \edef\LWR@thisref{\csuse{r@LWRindex-#1}}%
11294     \ifdefvoid{\LWR@thisref}{}{%
11295         \edef\LWR@thisref{\expandafter\@firstoftwo\LWR@thisref}%
11296         \ifdefvoid{\LWR@thisref}%
11297             {\LWR@indexnameref@anonref{#1}}%
11298             {\ref{LWRindex-#1}}%
11299     }%
11300 }
```

\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.

```
11301 \newcommand*{\LWR@indexnameref@refnameref}[1]{%
```

```
11302     \edef\LWR@thisref{\csuse{r@LWRindex-#1}}%
11303     \ifdefvoid{\LWR@thisref}{}{%
11304         \edef\LWR@thisref{\expandafter\@firstoftwo\LWR@thisref}%
11305         \ifdefvoid{\LWR@thisref}{}{%
11306             \ifdefstring{\LWR@thisref}{(*)}%
11307                 {}%
11308                 {\ref{LWRindex-#1} }% space
11309         }%
11310     }%
11311     \nameref{LWRindex-#1}%
11312 }
```

\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 (*).

```
11313 \newcommand*{\LWR@indexnameref@cref}[1]{%
11314     \edef\LWR@thisref{\csuse{r@LWRindex-#1}}%
11315     \ifdefvoid{\LWR@thisref}{%
11316         \nameref{LWRindex-#1}%
11317     }{%
11318         \edef\LWR@thisref{\expandafter\@firstoftwo\LWR@thisref}%
11319         \ifdefvoid{\LWR@thisref}{%
11320             \nameref{LWRindex-#1}%
11321         }{%
11322             \ifdefstring{\LWR@thisref}{(*)}{%
11323                 \LWR@indexnameref@anonref{#1}%
11324             }{%
11325                 \cref{LWRindex-#1}%
11326             }%
11327         }%
11328     }%
11329 }
```

\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.

```
11330 \newcommand*{\LWR@indexnameref@crefnameref}[1]{%
11331     \edef\LWR@thisref{\csuse{r@LWRindex-#1}}%
11332     \ifdefvoid{\LWR@thisref}%
11333         {}%
11334         {%
11335             \edef\LWR@thisref{\expandafter\@firstoftwo\LWR@thisref}%
11336             \ifdefvoid{\LWR@thisref}%
11337                 {}%
11338                 {%
11339                     \ifdefstring{\LWR@thisref}{(*)}%
11340                         {}%
11341                         {\cref{LWRindex-#1}} % space
11342                 }%
11343         }%
11344     \nameref{LWRindex-#1}%
11345 }
```

\LWR@indexnameref {⟨*LWR@autoindex*⟩}

> Creates a hyperlink based on the given entry's autoindex.

```
11346 \newcommand*{\LWR@indexnameref}[1]{%
11347     {% group
```

> Temporarily redefine caption's \caption@xref because it was printing ?? in the indexes, and also causing error on expansion:

```
11348         \ifdef{\caption@xref}{%
11349             \renewcommand*{\caption@xref}[2]{(*)}%
11350         }{}%

11351         \ifdefstring{\LWR@IndexRef}{ref}{%
11352             \LWR@indexnameref@ref{#1}%
11353         }{%
11354         \ifdefstring{\LWR@IndexRef}{nameref}{%
11355             \nameref{LWRindex-#1}%
11356         }{%
11357         \ifdefstring{\LWR@IndexRef}{refnameref}{%
11358             \LWR@indexnameref@refnameref{#1}%
11359         }{%
11360         \ifdefstring{\LWR@IndexRef}{cref}{%
11361             \LWR@indexnameref@cref{#1}%
11362         }{%
11363         \ifdefstring{\LWR@IndexRef}{crefnameref}{%
11364             \LWR@indexnameref@crefnameref{#1}%
11365         }{%
11366         \ifdefstring{\LWR@IndexRef}{autoref}{%
11367             \LWR@indexnameref@cref{#1}%
11368         }{% text string
11369             \LWR@startref{LWRindex-#1}%
11370             \LWR@IndexRef%
11371             \LWR@htmltag{/a}%
11372         }}}}}}%
11373     }% group
11374 }
```

\LWR@doindexentrysubsub {⟨*range start: LWR@autoindex, or macros.*⟩} {⟨*range end or blank*⟩}

> Creates a hyperlink, or handles \see, \textbf, etc.

```
11375 \newrobustcmd{\LWR@doindexentrysubsub}[2]{%
11376     \IfInteger{#1}%
11377         {\LWR@indexnameref{#1}}%
11378         {#1}%
11379     \IfValueT{#2}{%
11380         \IndexRangeSeparator%
11381         \IfInteger{#2}%
11382             {\LWR@indexnameref{#2}}%
11383             {#2}%
11384     }%
11385 }
```

\LWR@doindexentrysub {⟨*range delimiter*⟩} {⟨*LWR@autoindex or macros, possible a range*⟩}

```
11386 \NewDocumentCommand{\LWR@doindexentrysub}{m >{\SplitArgument{1}{#1}}m}
```

11387   {\LWR@doindexentrysubsub#2}

\LWR@doindexentry {⟨*LWR@autoindex or macros, possible a range*⟩}

11388 \newcommand*{\LWR@doindexentry}[1]{%
11389   \relax% required
11390   \expandafter\LWR@doindexentrysub\expandafter{\IndexRangeSeparator}{#1}%
11391 }

\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{. . . }

11392 \newcommand{\LWR@hyperindexrefnullified}{%
11393   \renewrobustcmd{\emph}[1]{\LWR@HTML@emph{\LWR@doindexentry{##1}}}%
11394   \renewrobustcmd{\textbf}[1]{\LWR@HTML@textbf{\LWR@doindexentry{##1}}}%
11395   \renewrobustcmd{\texteb}[1]{\LWR@HTML@texteb{\LWR@doindexentry{##1}}}%
11396   \renewrobustcmd{\textlg}[1]{\LWR@HTML@textlg{\LWR@doindexentry{##1}}}%
11397   \renewrobustcmd{\textrm}[1]{\LWR@HTML@textrm{\LWR@doindexentry{##1}}}%
11398   \renewrobustcmd{\textsf}[1]{\LWR@HTML@textsf{\LWR@doindexentry{##1}}}%
11399   \renewrobustcmd{\texttt}[1]{\LWR@HTML@texttt{\LWR@doindexentry{##1}}}%
11400   \renewrobustcmd{\textup}[1]{\LWR@HTML@textup{\LWR@doindexentry{##1}}}%
11401   \renewrobustcmd{\textsc}[1]{\LWR@HTML@textsc{\LWR@doindexentry{##1}}}%
11402   \renewrobustcmd{\textulc}[1]{\LWR@HTML@textulc{\LWR@doindexentry{##1}}}%
11403   \renewrobustcmd{\textsi}[1]{\LWR@HTML@textsi{\LWR@doindexentry{##1}}}%
11404   \renewrobustcmd{\textit}[1]{\LWR@HTML@textit{\LWR@doindexentry{##1}}}%
11405   \renewrobustcmd{\textsl}[1]{\LWR@HTML@textsl{\LWR@doindexentry{##1}}}%
11406 }

\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 in-
serted, which are defined to use \hyperindexref.

The argument is split at commas, and also for ranges, then passed to \LWR@hyperindexrefsub.

11407 \newcommand*{\hyperindexref}[1]{%
11408   \relax% required
11409   \expandafter\LWR@hyperindexref@comma\expandafter{\IndexPageSeparator}{#1}%
11410 }

\LWR@hyperindexref@comma {⟨*separator*⟩} {⟨*list of args*⟩}

The list is split at commas, and passed to \LWR@hyperindexref@@comma.

11411 \NewDocumentCommand{\LWR@hyperindexref@comma}
11412   {m >{\SplitList{#1}} m}
11413   {%

Used to place the separtor between each entry, but not before the first.

```
11414          \def\LWR@hyperindexref@thiscomma{}%
11415          \def\LWR@hyperindexref@nextcomma{#1}%
```

Each comma-delimited entry is now passed individually to \LWR@hyperindexref@@comma.

```
11416          \ProcessList{#2}\LWR@hyperindexref@@comma%
11417       }
```

\LWR@hyperindexref@@comma {⟨*arg, perhaps with a range*⟩}

A comma separator is placed if not the first item, then the range is parsed.

```
11418 \newcommand*{\LWR@hyperindexref@@comma}[1]{%
11419    \LWR@hyperindexref@thiscomma%
11420    \renewcommand{\LWR@hyperindexref@thiscomma}{\LWR@hyperindexref@nextcomma}%
11421    \expandafter\LWR@hyperindexref@range\expandafter{\IndexRangeSeparator}{#1}%
11422 }
```

\LWR@hyperindexref@range {⟨*range delimiter*⟩} {⟨*arg*⟩}

```
11423 \NewDocumentCommand{\LWR@hyperindexref@range}
11424    {m >{\SplitArgument{1}{#1}} m}
11425    {\LWR@hyperindexrefsub#2}
```

\LWR@hyperindexrefsub {⟨*range start: LWR@autoindex*⟩} {⟨*range end, or* -NoValue-⟩}

Handles the start and end of a range, if applicable.

```
11426 \newcommand*{\LWR@hyperindexrefsub}[2]{%
11427    \LWR@hyperindexrefsubtwo{#1}%
11428    \IfValueT{#2}{%
11429        \IndexRangeSeparator%
11430        \LWR@hyperindexrefsubtwo{#2}%
11431    }%
11432 }
```

\LWR@hyperindexrefsubtwo {⟨*LWR@autoindex*⟩}

```
11433 \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.

```
11434    \edef\LWR@tempone{#1}%
11435    \IfBeginWith{\LWR@tempone}{ }{%
11436        \StrGobbleLeft{\LWR@tempone}{1}[\LWR@tempone]%
11437    }{}%
```

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.

```
11438    \IfInteger{\LWR@tempone}%
11439        {\LWR@indexnameref{\LWR@tempone}}%
11440        {%
11441            \begingroup%
```

```
11442              \LWR@hyperindexrefnullified%
11443              #1%
11444              \endgroup%
11445          }%
11446 }
```

\hyperpage Emulate hyperref.

```
11447 \LetLtxMacro\hyperpage\hyperindexref
```

\nohyperpage Emulate hyperref.

```
11448 \def\nohyperpage#1{}
```

\hyperindexformat Emulate hyperref.

```
11449 \def\hyperindexformat#1#2{%
11450     #1{\hyperpage{#2}}%
11451 }%
```

```
11452 \end{warpHTML}
```

for PRINT output: A null command for print mode, in case hyperref was not used:

```
11453 \begin{warpprint}
11454 \newcommand{\hyperindexref}[1]{#1}
11455 \end{warpprint}
```

for HTML & PRINT: For the glossaries package, try to prevent an error where \glo@name was not found:

```
11456 \begin{warpall}
11457 \providecommand{\glo@name}{}
11458 \end{warpall}
```

80 Bibliography presentation

for HTML output: 11459 \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*⟩}

```
11460 \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

```
11461 \AtBeginDocument{
11462
11463 \AtBeginEnvironment{thebibliography}{
11464
11465 \providecommand*{\LWR@newem}[1]{\textit{#1}}
11466
11467 \renewrobustcmd{\em}{%
11468     \begingroup
11469         \gdef\LWR@em@after{\LWR@em@finish\LWR@newem}%
11470         \afterassignment\LWR@em@after
11471         \toks@\bgroup
11472 }
11473
11474 \def\LWR@em@finish#1{%
11475     \xdef\LWR@em@after{\noexpand#1{\the\toks@}}%
11476     \endgroup
11477     \LWR@em@after\egroup
11478 }
11479
11480 }% \AtBeginEnvironment{thebibliography}
11481
11482 }% \AtBeginDocument
```

```
11483 \end{warpHTML}
```

81   Restoring original formatting

```
11484 \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.

```
11485 \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.

```
11486 \newcommand*{\LWR@restoreorigformatting}{%
11487     \LWR@traceinfo{LWR@restoreorigformatting}%
```

Numerous macros change their print/HTML meaning depending on \LWR@formatting:

```
11488     \renewcommand*{\LWR@formatting}{print}%
11489     \linespread{1}%
```

```
11490    \setbool{LWR@doingparhooks}{false}%

11491    \def\color@endgroup{\endgraf\endgroup}%

11492    \LetLtxMacro\hfil\LWR@orighfil%
11493    \let\hss\LWR@orighss%
11494    \let\llap\LWR@origllap%
11495    \let\rlap\LWR@origrlap%
11496    \let\hfilneg\LWR@orighfilneg%

11497    \let\,\LWR@origcomma% disable HTML short unbreakable space
11498    \let\textless\LWR@origtextless%
11499    \let\textgreater\LWR@origtextgreater%

11500    \let\&\LWR@origampersand%

11501    \LetLtxMacro\em\LWR@origem%
11502    \LetLtxMacro\normalfont\LWR@orignormalfont%
11503    \let\sp\LWR@origsp%
11504    \let\sb\LWR@origsb%
11505    \LetLtxMacro\underline\LWR@origunderline%
11506    \let~\LWR@origtilde%
```

\endtabular must be restored to its original, instead of relying on lwarp's \LWR@formatted mechanism:

```
11507    \LetLtxMacro\endtabular\LWR@origendtabular%
11508    \csletcs{endtabular*}{LWR@origendtabular*}%

11509    \LetLtxMacro\noalign\LWR@orignoalign%
11510    \LetLtxMacro\hline\LWR@orighline%

11511    \let\newline\LWR@orignewline%
11512    \LetLtxMacro\includegraphics\LWR@origincludegraphics%

11513    \LetLtxMacro\@ensuredmath\LWR@origensuredmath%

11514    \let\math\LWR@orig@math%
11515    \let\endmath\LWR@orig@endmath%
11516    \let\displaymath\LWR@orig@displaymath%
11517    \let\enddisplaymath\LWR@orig@enddisplaymath%
11518 %
11519    \LWR@restoreorigaccents%
11520    \LWR@restoreoriglists%

11521    \let\@mpfootnotetext\LWR@orig@mpfootnotetext%

11522    \LWR@hook@processingtags%
```

To enable MATHJAX-specific nullification, used for tcolorbox:

```
11523    \ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
11524        {\LWR@restoreMathJaxformatting}%
11525        {}%
11526 }
```

11527 `\end{warpHTML}`

82   Nullifying filename formatting

The following are used to nullify certain macros and environments while convert-ing section names to file names.

11528 `\begin{warpHTML}`

Also commonly used are \@empty, \@gobble, and \@firstofone.

11529 `\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.

```
11530 \catcode'\$=\active% redefining $ below
11531 \catcode'\_=12% redefining \_ below
11532 \newcommand*{\LWR@nullfonts}{%
```

Various built-in symbols.

```
11533     \renewcommand*{\$}{-}%
11534     \renewcommand*{\%}{-}%
11535     \renewcommand*{\_}{-}%
11536     \renewcommand*{\}}{-}%
11537     \renewcommand*{\{}{-}%
11538     \renewcommand*{\&}{-}% used to be 'and'
11539     \renewcommand*{\#}{-}%
11540     \renewcommand*{\,}{-}%
11541     \renewcommand*{~}{-}%
11542 %
11543 % accents:
11544     \renewcommand*{\`}[1]{##1}%
11545     \renewcommand*{\'}[1]{##1}%
11546     \renewcommand*{\^}[1]{##1}%
11547     \renewcommand*{\~}[1]{##1}%
11548     \renewcommand*{\=}[1]{##1}%
11549     \renewcommand*{\u}[1]{##1}%
11550     \renewcommand*{\.}[1]{##1}%
11551     \renewcommand*{\"}[1]{##1}%
11552     \renewcommand*{\H}[1]{##1}%
11553     \renewcommand*{\v}[1]{##1}%
11554     \renewcommand*{\d}[1]{##1}%
11555     \renewcommand*{\c}[1]{##1}%
11556     \renewcommand*{\b}[1]{##1}%
11557     \renewcommand*{\t}[1]{##1}%
11558 %
11559     \let\newline\LWR@dash%
11560     \let\textasciicircum\LWR@dash%
```

```
11561    \let\textasciitilde\LWR@dash%
11562    \let\textasteriskcentered\LWR@dash%
11563    \let\textbackslash\LWR@dash%
11564    \let\textbar\LWR@dash%
11565    \let\textbardbl\LWR@dash%
11566    \let\textbigcircle\LWR@dash%
11567    \let\textbraceleft\LWR@dash%
11568    \let\textbraceright\LWR@dash%
11569    \let\textbullet\LWR@dash%
11570    \let\textcopyright\LWR@dash%
11571    \let\textdagger\LWR@dash%
11572    \let\textdaggerdbl\LWR@dash%
11573    \let\textdollar\LWR@dash%
11574    \let\textellipsis\LWR@dash%
11575    \let\textemdash\LWR@dash%
11576    \let\textendash\LWR@dash%
11577    \let\textexclamdown\LWR@dash%
11578    \let\textgreater\LWR@dash%
11579    \let\textless\LWR@dash%
11580    \let\textordfeminine\LWR@dash%
11581    \let\textordmasculine\LWR@dash%
11582    \let\textparagraph\LWR@dash%
11583    \let\textperiodcentered\LWR@dash%
11584    \let\textpertenthousand\LWR@dash%
11585    \let\textperthousand\LWR@dash%
11586    \let\textquestiondown\LWR@dash%
11587    \let\textquotedblleft\LWR@dash%
11588    \let\textquotedblright\LWR@dash%
11589    \let\textquoteleft\LWR@dash%
11590    \let\textquoteright\LWR@dash%
11591    \let\textregistered\LWR@dash%
11592    \let\textsection\LWR@dash%
11593    \let\textsterling\LWR@dash%
11594    \let\texttrademark\LWR@dash%
11595    \let\textunderscore\LWR@dash%
11596    \let\textvisiblespace\LWR@dash%
11597    \let\copyright\LWR@dash%
11598    \let\dag\LWR@dash%
11599    \let\ddag\LWR@dash%
11600    \let\dots\LWR@dash%
11601    \let\P\LWR@dash%
11602    \let\pounds\LWR@dash%
11603    \let\S\LWR@dash%
11604 %
11605    \renewcommand*{\aa}{a}%
11606    \renewcommand*{\AA}{A}%
11607    \renewcommand*{\AE}{AE}%
11608    \renewcommand*{\ae}{ae}%
11609    \renewcommand*{\dh}{d}%
11610    \renewcommand*{\DH}{D}%
11611    \renewcommand*{\DJ}{D}%
11612    \renewcommand*{\dj}{d}%
11613    \renewcommand*{\IJ}{IJ}%
11614    \renewcommand*{\ij}{ij}%
11615    \renewcommand*{\L}{L}%
11616    \renewcommand*{\l}{l}%
11617    \renewcommand*{\NG}{NG}%
11618    \renewcommand*{\ng}{ng}%
11619    \renewcommand*{\O}{O}%
11620    \renewcommand*{\o}{o}%
```

```
11621    \renewcommand*{\oe}{oe}%
11622    \renewcommand*{\OE}{OE}%
11623    \renewcommand*{\ss}{ss}%
11624    \renewcommand*{\SS}{SS}%
11625    \renewcommand*{\th}{th}%
11626    \renewcommand*{\TH}{TH}%
11627 %
11628    \let\guillemotleft\@empty%
11629    \let\guilsinglleft\@empty%
11630    \let\quotedblbase\@empty%
11631    \let\textquotedbl\@empty%
11632    \let\guillemotright\@empty%
11633    \let\guilsinglright\@empty%
11634    \let\quotesinglbase\@empty%


11635    \renewcommand*{\HTMLunicode}[1]{}%
11636    \renewcommand*{\HTMLentity}[1]{}%


11637    \renewcommand{\textsuperscript}[1]{##1}%
11638    \renewcommand{\textsubscript}[1]{##1}%


11639    \renewcommand{\underline}[1]{##1}%


11640    \RenewDocumentCommand{\hspace}{s m}{}%


11641    \RenewDocumentCommand{\LWR@htmlspanclass}{o D(){} m +m}{{##4}}%
11642    \DeclareExpandableDocumentCommand{\InlineClass}{D(){}{} o m +m}{{##4}}%
```

Nullify math macros.

```
11643    \def\(##1\){}%
11644    \def\[##1\]{}%
11645    \RenewDocumentCommand{\LWR@subsingledollar}{s m m m}{}%
```

Nullify logos:

```
11646    \renewcommand*{\TeX}{TeX}%
11647    \renewcommand*{\LaTeX}{LaTeX}%
11648    \renewcommand*{\LaTeXe}{LaTeX2e}%
11649    \renewcommand*{\LuaTeX}{LuaTeX}%
11650    \renewcommand*{\LuaLaTeX}{LuaLaTeX}%
11651    \renewcommand*{\XeTeX}{XeTeX}%
11652    \renewcommand*{\XeLaTeX}{XeLaTeX}%
11653    \renewcommand*{\ConTeXt}{ConTeXt}%
11654    \renewcommand*{\BibTeX}{BibTeX}%
11655    \renewcommand*{\MakeIndex}{MakeIndex}%
11656    \renewcommand*{\AmS}{AmS}%
11657    \renewcommand*{\MiKTeX}{MiKTeX}%
11658    \renewcommand*{\LyX}{LyX}%
```

Use the simpler form with \texorpdfstring:

```
11659    \def\texorpdfstring{\expandafter\@secondoftwo}%
11660 }
11661 \catcode`\$=3%
11662 \catcode`\_=8%
```

`\FilenameNullify` {⟨*redefinitions*⟩}

Adds more nullifying definitions for filename generation.

```
11663 \newcommand*{\FilenameNullify}[1]{%
11664     \appto{\LWR@nullfonts}{#1}%
11665 }
```

```
11666 \end{warpHTML}
```

83 Math

83.1 Limitations

See Math, section 8.7.

83.2 HTML alt tag names

Redefinable names for the HTML alt tags, for translation according to the reader's native language.

for HTML & PRINT: 11667 `\begin{warpall}`

`\AltTextOpen` The opening part of HTML alt tag for an image. The default is a left parenthesis.
Default: (

```
11668 \newcommand*{\AltTextOpen}{(}
```

`\AltTextClose` The closing part of HTML alt tag for an image. The default is a right parenthesis.
Default: (

```
11669 \newcommand*{\AltTextClose}{)}
```

`\ImageAltText` The HTML alt tag for an image.
Default: image

```
11670 \newcommand*{\ImageAltText}{image}
```

`\MathImageAltText` The HTML alt tag for an SVG math image.
Default: "math image"

```
11671 \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.

```
11672 \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.

```
11673 \newcommand*{\ThisAltText}[1]{%
11674     \renewcommand{\LWR@ThisAltText}{#1}%
11675 }
```

\PackageDiagramAltText Appended to the lateximage HTML alt tag for the images generated by many
Default: "diagram" packages.

```
11676 \newcommand*{\PackageDiagramAltText}{diagram}
```

```
11677 \end{warpall}
```

83.3   Inline and display math

for HTML output: 11678 \begin{warpHTML}

LWR@externalfilecnt (*Ctr*) Counter for the external files which are generated and then referenced from the HTML:

```
11679 \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.

```
11680 \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.

```
11681 \newbool{LWR@insidemathcomment}
11682 \boolfalse{LWR@insidemathcomment}
```

LWR@xfakebold (*bool*) True if xfakebold \setBold is in use.

```
11683 \newbool{LWR@xfakebold}
11684 \boolfalse{LWR@xfakebold}
```

\LWR@orig@setBold Redefined by lwarp-xfakebold.

```
11685 \newcommand*{\LWR@orig@setBold}{}
```

\LWR@orig@unsetBold Redefined by lwarp-xfakebold.

```
11686 \newcommand*{\LWR@orig@unsetBold}{}
```

\LWR@applyxfakebold Redefined by lwarp-xfakebold.

```
11687 \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.

```
11688 \newcommand*{\LWR@setcurrentfont}{%
11689     \LWR@traceinfo{Using font family \LWR@f@family}%
11690     \@nameuse{LWR@print@\LWR@f@family family}%
11691     \LWR@traceinfo{Using font series \LWR@f@series}%
11692     \@nameuse{LWR@print@\LWR@f@series series}%
11693     \LWR@traceinfo{Using font shape \LWR@f@shape}%
11694     \@nameuse{LWR@print@\LWR@f@shape shape}%
11695     \LWR@traceinfo{Using font caps shape \LWR@f@shapecaps}%
11696     \@nameuse{LWR@print@\LWR@f@shapecaps shape}%
11697 }
```

\\$ 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.

```
11698 \let\LWR@origtextdollar\$
11699
11700 \renewcommand*{\$}{%
11701     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
11702         {\LWR@origtextdollar}%
11703         {\HTMLunicode{00024}}%
11704 }
```

lwarp_baseline_marker.png   A marker to be used to help *pdfcrop* identify the inline math baseline and width.
(*file*)   If either graphicx or graphics is loaded, this marker is placed at the lower left and
lwarp_baseline_marker.eps   lower right corners of the inline math. *pdfcrop* is then able to identify the width of
(*file*)   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.

```
11705 \AtBeginDocument{
11706
11707 \ifpdf
11708     \newcommand*{\LWR@baselinename}{lwarp_baseline_marker.png}
11709 \else
11710     \ifXeTeX
11711         \newcommand*{\LWR@baselinename}{lwarp_baseline_marker.png}
11712     \else
11713         \newcommand*{\LWR@baselinename}{lwarp_baseline_marker.eps}
```

```
11714    \fi
11715 \fi
11716
11717 \IfFileExists{\LWR@baselinename}%
11718 {
11719    \IfPackageLoadedTF{graphicx}{
11720        \newcommand*{\LWR@addbaselinemarker}{%
11721            \LWR@origincludegraphics{\LWR@baselinename}%
11722        }
11723    }{
11724        \IfPackageLoadedTF{graphics}{
11725            \newcommand*{\LWR@addbaselinemarker}{%
11726                \LWR@origincludegraphics{\LWR@baselinename}%
11727            }
11728        }{
11729            \newcommand*{\LWR@addbaselinemarker}{%
11730                \global\booltrue{LWR@warnbaselinemarker}%
11731            }
11732            \AtEndDocument{
11733                \ifbool{LWR@warnbaselinemarker}{
11734                    \PackageNoteNoLine{lwarp}{%
11735                        Load graphicx or graphics for improved\MessageBreak
11736                        SVG math sizing and baselines%
11737                    }
11738                }{}
11739            }
11740        }
11741    }
11742 }{% lwarp_baseline_marker.png or .eps is not present
11743    \newcommand*{\LWR@addbaselinemarker}{%
11744        \global\booltrue{LWR@warnbaselinemarker}%
11745    }
11746    \AtEndDocument{
11747        \ifbool{LWR@warnbaselinemarker}{
11748            \PackageWarningNoLine{lwarp}{%
11749                File \LWR@baselinename\space is not installed\MessageBreak
11750                alongside the lwarp-*.sty files, so\MessageBreak
11751                SVG math sizing and baselines may not be accurate}
11752        }{}
11753    }
11754 }
11755
11756 }% AtBeginDocument
```

LWR@warnbaselinemarker (*bool*)   True if the math baseline marker was ever called for, but graphics or graphicx were not loaded.

```
11757 \newbool{LWR@warnbaselinemarker}
11758 \boolfalse{LWR@warnbaselinemarker}
```

LWR@unknownmathsize (*bool*)   If Ti*k*Z 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.

```
11759 \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.

```
11760 \newcommand*{\LWR@singledollarmeasure}[1]{%
11761     \begingroup%
```

Temporarily disable formatting while measuring the image parameters:

```
11762     \LWR@restoreorigformatting%
11763     \RenewDocumentEnvironment{lateximage}{s o s o o d()}{}{}% inside group
11764     \LWR@print@normalsize%
```

Temporarily set font for the html pdf output:

```
11765     \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.

```
11766     \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.

```
11767     \boolfalse{LWR@unknownmathsize}%
11768     \ifmmode%
11769         \global\sbox{\LWR@singledollarbox}{#1}%
11770     \else%
11771         \ifbool{LWR@dynamicmath}{%
11772             \ifbool{mathjax}{%
11773                 \global\sbox{\LWR@singledollarbox}%
11774                     {\LWR@origensuredmath{#1}}%
11775             }{%
11776                 \global\sbox{\LWR@singledollarbox}{#1}%
11777             }%
11778         }{%
11779             \global\sbox{\LWR@singledollarbox}{#1}%
11780         }%
11781     \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.

```
11782      \global\sbox{\LWR@singledollarbox}{%
11783          \usebox{\LWR@singledollarbox}%
11784          \raisebox{-\dp\LWR@singledollarbox}{%
11785              \LWR@addbaselinemarker%
11786          }%
11787      }%
```

More low-level code to undo the counter change.

```
11788      \global\advance\c@LWR@lateximagedepth -1\relax% Due to AmS \text macro.
```

Measure the depth:

```
11789      \setlength{\LWR@singledollardepth}{%
11790          \LateximageFontScale\dp\LWR@singledollarbox%
11791      }%
```

Make the length a global change:

```
11792      \global\LWR@singledollardepth=\LWR@singledollardepth%
```

Likewise for width:

```
11793      \setlength{\LWR@singledollarwidth}{%
11794          \LateximageFontScale\wd\LWR@singledollarbox%
11795      }%
11796      \global\LWR@singledollarwidth=\LWR@singledollarwidth%
```

Likewise for total height:

```
11797      \setlength{\LWR@singledollarheight}{%
11798          \LateximageFontScale\ht\LWR@singledollarbox%
11799      }%
11800      \addtolength{\LWR@singledollarheight}{%
11801          \LateximageFontScale\dp\LWR@singledollarbox%
11802      }%
11803      \global\LWR@singledollarheight=\LWR@singledollarheight%
```

```
11804      \endgroup%
11805 }
```

\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.

```
11806 \newcommand*{\LWR@subsingledollarsvg}[4]{%
11807     \LWR@traceinfo{LWR@subsingledollartsvg}%
```

Measure the depth, width, and height of the math image:

```
11808     \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:

```
11809     \ifdimgreater{\LWR@singledollarwidth}{.7\LWR@singledollarheight}{%
11810         \def\LWR@singledollarstyle{%
11811             width:\LWR@convertto{em}{\the\LWR@singledollarwidth} em%
11812         }%
11813     }{%
11814         \def\LWR@singledollarstyle{%
11815             height:\LWR@convertto{em}{\the\LWR@singledollarheight} em%
11816         }%
11817     }%
```

If a very narrow width, use the height.

```
11818     \ifdimless{\LWR@singledollarwidth}{.2em}%
11819     {%
11820         \def\LWR@singledollarstyle{%
11821             height:\LWR@convertto{em}{\the\LWR@singledollarheight} em%
11822         }%
11823     }%
11824     {}%
```

If very wide and short, use the width:

```
11825     \ifdimless{\LWR@singledollarheight}{.2em}%
11826     {%
11827         \def\LWR@singledollarstyle{%
11828             width:\LWR@convertto{em}{\the\LWR@singledollarwidth} em%
11829         }%
11830     }%
11831     {}%
```

If there is significant text depth, add the depth to the style.

```
11832     \ifdimgreater{\LWR@singledollardepth}{0.05ex}{%
11833         \def\LWR@singledollardepthstyle{%
11834             \ ; % extra space
11835             \LWR@print@mbox{%
11836           vertical-align:-\LWR@convertto{em}{\the\LWR@singledollardepth} em%
11837             } % extra space
11838         }%
11839     }{%
11840         \def\LWR@singledollardepthstyle{}%
11841     }%
```

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.

```
11842    \ifbool{LWR@unknownmathsize}{%
11843        \def\LWR@singledollarstyle{}%
11844        \def\LWR@singledollardepthstyle{}%
11845    }{}%
```

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.

```
11846    \ifbool{LWR@dynamicmath}{%
11847        \LWR@traceinfo{subsingledollarsvg: dynamic}%
11848        \begin{lateximage}% no hashing
11849            [\MathImageAltText]% alt tag
11850            []% no add'l hashing
11851            [\LWR@singledollarstyle \LWR@singledollardepthstyle]% CSS
11852            (math)% ARIA
11853    }{% not dynamic math
11854        \LWR@traceinfo{subsingledollarsvg: static}%
11855        \IfValueTF{#1}{% #1 True
11856            \LWR@findcurrenttextcolor% sets \LWR@tempcolor
```

Support for xfakebold:

```
11857            \ifbool{LWR@xfakebold}%
11858                {\def\LWR@tempone{Y}}%
11859                {\def\LWR@tempone{N}}%

11860            \LWR@traceinfo{subsingledollarsvg about to lateximage}%
11861            \begin{lateximage}*% use hashing
11862                [#2]% alt
11863                *% do not add open/closing braces
11864                [% addl' hashing
11865                    #3%
11866                    FM\LWR@f@family%
11867                    SR\LWR@f@series%
11868                    SH\LWR@f@shape%
11869                    SHC\LWR@f@shapecaps%
11870                    CL\LWR@tempcolor%
11871                    FB\LWR@tempone% xfakebold
11872                ]%
11873                [\LWR@singledollarstyle \LWR@singledollardepthstyle]% CSS
11874                (math)% ARIA
11875                \LWR@traceinfo{subsingledollar did lateximage}%
11876        }{% #1 False
11877            \begin{lateximage}% no hashing
11878                [#2]% alt
11879                []% no add'l hashing
11880                [\LWR@singledollarstyle \LWR@singledollardepthstyle]% CSS
11881                (math)% ARIA
11882        }%
11883    }% 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.)

11884   \LWR@addbaselinemarker%

Support for xfakebold:

11885   \LWR@applyxfakebold%

Typeset the contents:

11886   \usebox{\LWR@singledollarbox}%

The closing baseline marker:

11887   \LWR@addbaselinemarker%

11888   \end{lateximage}%
11889 %
11890 }

\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.

11891 \newlength{\LWR@singledollarwidth}
11892 \newlength{\LWR@singledollarheight}
11893 \newlength{\LWR@singledollardepth}
11894
11895 \newsavebox{\LWR@singledollarbox}

```
11896
11897 \NewDocumentCommand{\LWR@subsingledollar}{s m m m}{%
11898     \LWR@traceinfo{LWR@subsingledollar !#2!}%

11899     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
11900     {%
11901         \LWR@traceinfo{LWR@subsingledollar: already in a lateximage}%
11902             #4% contents
11903     }%
11904     {% not in a lateximage
11905         \begingroup%
```

Support for xfakebold:

```
11906         \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:

```
11907         \ifboolexpr{%
11908             (
11909                 bool{mathjax} or
11910                 ( bool{FormatWP} and bool{WPMarkMath} )
11911             ) and
11912             ( not test {
11913                     \ifstrequal {#2}% from \ensuredmath
11914                         {\AltTextOpen\MathImageAltText\AltTextClose}
11915                 }
11916             ) and
11917             ( not bool{LWR@dynamicmath} )
11918         }%
```

For MATHJAX, print the math between \(and \):

```
11919         {%
11920             \LWR@traceinfo{LWR@subsingledollar: Mathjax}%
11921             {%
11922                 \textbackslash(%
11923                 {%
```

\ifmmode to avoid error about \ttfamily inside math mode in the case of nested math, ex. equation with tcolorbox with math.

```
11924                     \ifmmode\else\LWR@print@ttfamily\fi%
11925                     \LWR@HTMLsanitizedetokenized{\detokenize{#4}}%
11926                 }%
11927                 \textbackslash)%
11928             }%
11929         }% 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.

```
11930        {% not mathjax
11931          \LWR@traceinfo{%
11932          LWR@subsingledollar: NOT mathjax, or is ensuremath, or is dynamic%
11933          }%
11934          \LWR@subsingledollarsvg{#1}{#2}{#3}{#4}%
11935        }% not mathjax
11936        \endgroup%
11937    }% not in a lateximage
```

Clear the single-use alt text:

```
11938    \gdef\LWR@ThisAltText{}%
11939    \LWR@traceinfo{LWR@subsingledollar: done}%
11940 }
```

```
11941 \LetLtxMacro\LWR@origdollar$
11942 \LetLtxMacro\LWR@secondorigdollar$% balance for editor syntax highlighting
```

```
11943 \LetLtxMacro\LWR@origopenparen\(
11944 \LetLtxMacro\LWR@origcloseparen\)
11945 \LetLtxMacro\LWR@origopenbracket\[
11946 \LetLtxMacro\LWR@origclosebracket\]
```

$ Redefine the dollar sign to place math inside a lateximage, or use MATHJAX:

$$

```
11947 \begingroup
11948 \catcode`\$=\active%
11949 \protected\gdef${\@ifnextchar$\LWR@doubledollar\LWR@singledollar}%
```

Used by chemformula to escape single-dollar math:

```
11950 \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:

```
11951 \protected\gdef\LWR@doubledollar$#1$${%
```

If MATHJAX or formatting for a word processor, print the LATEX expression:

```
11952    \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.

```
11953    {% intentional blank line:
11954
11955      \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.

```
11956        \LWR@hidelatexequation{math}{#1}%
```

```
11957            \InlineClass{hidden}{\LWR@syncnotenumbers}%
11958            \textbackslash[%
11959            {%
11960                \LWR@print@ttfamily%
11961                \LWR@HTMLsanitizedetokenized{\detokenize{#1}}%
11962            }%
11963            \textbackslash]
11964            \InlineClass{hidden}{\LWR@syncnotenames}%
11965        }{%
11966            \textbackslash[%
11967            {%
11968                \LWR@print@ttfamily%
11969                \LWR@HTMLsanitizedetokenized{\detokenize{#1}}%
11970            }%
11971            \textbackslash]
11972        }%
11973
11974    }% mathjax
```

For svg, print the math inside a lateximage, with an <alt> tag of the LATEX code:

```
11975    {% not mathjax
11976        \begin{BlockClass}{displaymath}%
11977        \LWR@newautoidanchor%
11978        \booltrue{LWR@indisplaymathimage}%
11979        \begin{lateximage}%
11980        [%
11981            \textbackslash{[} % extra space
11982            \LWR@HTMLsanitizedetokenized{\detokenize{#1}} % extra space
11983            \textbackslash{]}%
11984        ]%
11985        *% do not add open/closing braces
11986        (math)% ARIA
```

Support for xfakebold:

```
11987            \LWR@applyxfakebold%

11988            \LWR@origdollar\LWR@origdollar#1\LWR@origdollar\LWR@origdollar%
11989        \end{lateximage}%
11990        \end{BlockClass}%
11991    }% not mathjax
```

Clear the single-use alt text:

```
11992    \gdef\LWR@ThisAltText{}%
11993 }%
```

\LWR@singledollar {⟨*math expression*⟩}

```
11994 \protected\gdef\LWR@singledollar#1${%
11995    \LWR@traceinfo{LWR@singledollar}%
11996    \ifbool{mathjax}{%
11997        \LWR@subsingledollar*%
11998        {% alt tag
11999            \textbackslash( %
12000            \LWR@HTMLsanitizedetokenized{\detokenize{#1}} % extra space
```

```
12001          \textbackslash)%
12002        }%
12003        {singledollar}% add'l hashing
12004        {#1}% contents
12005    }{% not mathjax
12006        \LWR@subsingledollar*%
12007        {% alt tag
12008            \textbackslash( %
12009            \LWR@HTMLsanitizedetokenized{\detokenize{#1}} % extra space
12010            \textbackslash)%
12011        }%
12012        {singledollar}% add'l hashing
12013        {\LWR@origensuredmath{#1}}% contents
12014    }% not mathjax
```

Clear the single-use `alt` text:

```
12015        \gdef\LWR@ThisAltText{}%
12016 }
```

`\(` Redefine to the above dollar macros.
`\[`
```
12017 \AtBeginDocument{
12018    \protected\gdef\(#1\){$#1$}
12019    \protected\gdef\[#1\]{$$#1$$}
12020 }
12021
12022 \endgroup% active $
```

```
12023 \AtBeginDocument{
12024 \LetLtxMacro\LWR@openbracketnormal\[
12025 \LetLtxMacro\LWR@closebracketnormal\]
12026 }
```

`\@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.

```
12027 \LetLtxMacro\LWR@origensuredmath\@ensuredmath
12028
12029 \renewcommand{\@ensuredmath}[1]{%
12030    \ifbool{mathjax}{%
12031        \LWR@subsingledollar*{\AltTextOpen\MathImageAltText\AltTextClose}%
12032        {%
12033         \protect\LWR@HTMLsanitizedetokenized{\detokenize\expandafter{#1}}%
12034        }%
12035        {%
12036            \relax%
12037            \LWR@origensuredmath{#1}%
12038        }%
12039    }{% SVG math
```

> If already inside a `lateximage` in math mode, continue as-is.

```
12040        \ifmmode%
```

```
12041              \LWR@origensuredmath{#1}%
12042         \else%
```

Create an inline math lateximage with a simple alt tag and additional hashing according to the contents.

```
12043              \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
12044                  {\LWR@origensuredmath{#1}}%
12045                  {%
12046                      \LWR@subsingledollar*%
12047                          {\AltTextOpen\MathImageAltText\AltTextClose}%
12048                          {%
12049                              \protect\LWR@HTMLsanitizedetokenized{%
12050                                  \detokenize\expandafter{#1}%
12051                              }%
12052                          }%
12053                          {\LWR@origensuredmath{#1}}%
12054                  }%
12055         \fi%
12056     }%
```

Clear the single-use alt text:

```
12057     \gdef\LWR@ThisAltText{}%
12058 }
```

Remember then remove the old math and displaymath environments:

```
12059 \let\LWR@orig@math\math
12060 \let\LWR@orig@endmath\endmath
12061
12062 \let\LWR@orig@displaymath\displaymath
12063 \let\LWR@orig@enddisplaymath\enddisplaymath
12064
12065 \let\math\relax
12066 \let\endmath\relax
12067
12068 \let\displaymath\relax
12069 \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.

```
12070 \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.

```
12071 \NewEnviron{LWR@displaymathnormal}{\expandafter\[\BODY\]\@ignoretrue}
```

Set the default displaymath to the normal version:

```
12072 \LetLtxMacro\displaymath\LWR@displaymathnormal%
12073 \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.

```
12074 \newenvironment{LWR@displaymathother}
12075 {%
12076     \begin{BlockClass}{displaymath}%
12077     \LWR@newautoidanchor%
12078     \booltrue{LWR@indisplaymathimage}%
12079     \begin{lateximage}[\MathImageAltText](math)% [alt](ARIA)
12080     \LWR@origdollar\LWR@origdollar%
12081 }
12082 {%
12083     \LWR@origdollar\LWR@origdollar%
12084     \end{lateximage}%
12085     \end{BlockClass}%
12086 }
```

LWR@equationother (*env.*) A version of displaymath which can handle complicated objects, but does not supply MATHJAX or HTML alt tags.

```
12087 \newenvironment{LWR@equationother}
12088 {%
12089     \begin{BlockClass}{displaymathnumbered}%
12090     \LWR@newautoidanchor%
12091     \booltrue{LWR@indisplaymathimage}%
12092     \begin{lateximage}[\MathImageAltText](math)% [alt](ARIA)
12093     \LWR@orig@equation%
12094 }
12095 {%
12096     \LWR@orig@endequation%
12097     \end{lateximage}%
12098     \end{BlockClass}%
12099 }
```

83.4 MATHJAX support

LWR@nextequation (*Ctr*)   Used to add one to compute the next equation number.

```
12100 \newcounter{LWR@nextequation}
```

Determing how to set MATHJAX section and equation numbers. Adjusts for various kinds of \theequation to determine \theMathJaxsection and \theMathJaxequation.

```
12101 \newcommand\LWR@article@theequation{\@arabic\c@equation}
12102
12103 \newcommand\LWR@book@theequation
12104   {\ifnum \c@chapter>\z@ \thechapter.\fi \@arabic\c@equation}
12105
12106
12107 \newcommand\LWR@chapter@theequation{\thechapter.\arabic{equation}}
12108 \newcommand\LWR@section@theequation{\thesection.\arabic{equation}}
12109 \newcommand\LWR@subsection@thequation{\thesubsection.\arabic{equation}}
12110
12111 \AtBeginDocument{
12112     % default per article class:
12113     \newcommand*{\theMathJaxsubequations}{0}
12114     \newcommand*{\theMathJaxsection}{}
```

```
12115     \newcommand*{\theMathJaxequation}{\arabic{equation}}
12116
12117     \ifdefstrequal{\theequation}{\LWR@article@theequation}
12118     {}{
12119     \ifdefstrequal{\theequation}{\LWR@book@theequation}{
12120      \renewcommand*{\theMathJaxsection}{\ifnum \c@chapter>\z@ \thechapter.\fi}
12121     }{
12122     \ifdefstrequal{\theequation}{\LWR@subsection@thequation}{
12123         \renewcommand*{\theMathJaxsection}{\thesubsection{}.}
12124     }{
12125     \ifdefstrequal{\theequation}{\LWR@section@thequation}{
12126         \renewcommand*{\theMathJaxsection}{\thesection{}.}
12127     }{
12128     \ifdefstrequal{\theequation}{\LWR@chapter@theequation}{
12129         \renewcommand*{\theMathJaxsection}{\thechapter{}.}
12130     }{% unknown format
12131         \PackageWarningNoLine{lwarp}
12132         {%
12133           Unknown equation tag format for \protect\theequation.\MessageBreak
12134             Article-style equation numbering will be used%
12135         }
12136     }}}}}
12137 }
```

\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.

```
12138 \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.

```
12139       \LWR@stoppars%
12140       \InlineClass{hidden}{
12141           \textbackslash(%
12142           \textbackslash{}seteqnumber%
12143           \{\theMathJaxsubequations\}%
12144           \{\theMathJaxsection\}%
12145           \{\theMathJaxequation\}%
12146           \textbackslash)%
12147       }
12148       \LWR@startpars%
12149 }
```

\LWR@hidelatexequation {⟨*environment*⟩} {⟨*contents*⟩}

> Creates the LATEX version of the equation inside an HTML comment.

```
12150 \NewDocumentCommand{\LWR@hidelatexequation}{m +m}{%
```

> Stop HTML paragraph handling and open an HTML comment:

```
12151     \LWR@stoppars
12152     \LWR@htmlopencomment
12153
```

> Start the LATEX math environment inside the HTML comment:

```
12154     \begingroup
12155     \@nameuse{LWR@orig@#1}
```

> While in the math environment, restore various commands to their LATEX meanings.

```
12156     \LWR@restoreorigformatting
12157     \booltrue{LWR@insidemathcomment}
```

> Temporarily prevent underfull \hbox warnings.

```
12158     \hbadness=10000\relax%
```

> See \LWR@htmlmathlabel in section 83.7.1.

> Print the contents of the equation:

```
12159     #2
```

> End the LATEX math environment inside the HTML comment:

```
12160     \@nameuse{LWR@orig@end#1}
12161     \endgroup
12162
```

> Close the HTML comment and resume HTML paragraph handling:

```
12163     \LWR@htmlclosecomment
12164     \boolfalse{LWR@insidemathcomment}
12165     \LWR@startpars
12166 }
```

\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.

```
12167 \NewDocumentCommand{\LWR@addmathjax}{m +m}{%
```

```
12168     \LWR@origtilde\LWR@orignewline
```

> Enclose the MATHJAX environment inside printed "\(" and "\)" characters. Print the environment name and contents, sanitizing for HTML special characters.

```
12169    {%
12170        \LWR@print@ttfamily%
12171        \textbackslash{}begin\{#1\}
```

The alignat environment takes a mandatory argument, which must be replicated here.

```
12172        \ifboolexpr{
12173            test {\ifstrequal{#1}{alignat}} or
12174            test {\ifstrequal{#1}{alignat*}} or
12175            test {\ifstrequal{#1}{alignat+}}
12176        }%
12177            {\{\arabic{LWR@maxfields@}\}}%
12178            {}%
```

The environment contents and \end:

```
12179        \LWR@orignewline%
12180        \LWR@HTMLsanitizeexpanded{\detokenize\expandafter{#2}}%
12181        \LWR@orignewline%
12182        \textbackslash{}end\{#1\}
12183    }%
```

```
12184    \LWR@orignewline
12185 }
```

83.5 Equation environment

Remember existing equation environment, after redefined by amsmath, if loaded.

```
12186 \AtBeginDocument{
12187    \let\LWR@orig@equation\equation
12188    \let\LWR@orig@endequation\endequation
12189    \csletcs{LWR@orig@equation*}{equation*}
12190    \csletcs{LWR@orig@endequation*}{endequation*}
12191 }
```

\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.

```
12192 \newcommand*{\LWR@doequation}[2]{%
12193
```

If mathjax or FormatWP, print the LATEX expression:

```
12194    \ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
```

MATHJAX output:

```
12195    {
```

Print commands to syncronize MATHJAX's equation number and format to the current LATEX chapter/section and equation number:

```
12196          \LWR@syncmathjax%
```

Print the LATEX math inside an HTML comment:

```
12197          \LWR@hidelatexequation{#2}{#1}
12198      }
```

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.

```
12199      {% not mathjax
```

Begin the lateximage with an <alt> tag containing the math source:

```
12200          \ifstrequal{#2}{equation*}{%
12201              \begin{BlockClass}{displaymath}%
12202          }{%
12203              \begin{BlockClass}{displaymathnumbered}%
12204          }%
12205          \LWR@newautoidanchor%
12206          \booltrue{LWR@indisplaymathimage}%
12207          \begin{lateximage}[%
12208              \ifstrequal{#2}{equation*}{%
12209                  \ifdefequal{\LWR@equationtag}{\theequation}{%
12210      %                                   no tag was given
12211              }{%
12212                  (\LWR@equationtag) % tag was given
12213              }%
12214          }{%
12215              (\LWR@equationtag) % automatic numbering
12216          }%
12217          \textbackslash{begin\{#2\}} % extra space
12218          \LWR@HTMLsanitizeexpanded{\detokenize\expandafter{#1}} % extra space
12219          \textbackslash{end\{#2\}}%
12220      ]*(math)% alt tag, ARIA
```

Support for xfakebold:

```
12221          \LWR@applyxfakebold%
```

Create the actual LATEX-formatted equation inside the lateximage using the contents of the environment.

```
12222          \@nameuse{LWR@orig@#2}%
12223          #1% contents collected by \collect@body
12224          \@nameuse{LWR@orig@end#2}%
12225          \end{lateximage}%
12226          \end{BlockClass}%
12227      }% not mathjax
```

Clear the single-use alt text:

```
12228      \gdef\LWR@ThisAltText{}%
12229 }
```

After the environment, if MᴀᴛʜJᴀx, print the math to the ʜᴛᴍʟ output for MᴀᴛʜJᴀx processing. If a footnote is used, sync the footnote counter before, then unsync after for non-equation environments, as defined next.

```
12230 \newcommand*{\LWR@doendequation}[1]{%
12231     \ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
12232     {%
12233         \IfSubStr\detokenize\expandafter{\BODY}}{\detokenize{note}}{%
12234             \InlineClass{hidden}{\LWR@syncnotenumbers}%
12235             \LWR@addmathjax{#1}{\BODY}%
12236             \InlineClass{hidden}{\LWR@syncnotenames}%
12237         }{%
12238             \LWR@addmathjax{#1}{\BODY}%
12239         }%
12240     }{}%
12241
```

Clear the single-use alt text:

```
12242     \gdef\LWR@ThisAltText{}%
12243 }
```

The following are used to syncronize footnote marks and related to MᴀᴛʜJᴀx if *note* is used inside the MᴀᴛʜJᴀx expression. The counter is read from LᴬTᴇX then defined into MᴀᴛʜJᴀx for use during the following equation. After the equation, the MᴀᴛʜJᴀx 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*⟩}

```
12244 \newcommand*{\LWR@synconenotenumber}[2]{%
12245     \textbackslash(
12246     \textbackslash{}def\textbackslash{}#1\{#2\}
12247     \textbackslash)
12248 }
```

\LWR@syncnotenumbers Assignments to make.

```
12249 \newcommand*{\LWR@syncnotenumbers}{\LWR@synconenotenumber{LWRfootnote}{\thefootnote}}
```

\LWR@synconenotename {⟨*MathJax variable*⟩} {⟨*text*⟩}

```
12250 \newcommand*{\LWR@synconenotename}[2]{%
12251     \textbackslash(
12252     \textbackslash{}def\textbackslash{}#1name\{#2\}
12253     \textbackslash)
12254 }
```

\LWR@syncnotenames Assignments to make.

```
12255 \newcommand*{\LWR@syncnotenames}{\LWR@synconenotename{LWRfootnote}{\footnotename}}
```

Remove existing equation environment:

```
12256 \AtBeginDocument{
12257     \let\equation\relax
12258     \let\endequation\relax
12259     \csletcs{equation*}{relax}
12260     \csletcs{endequation*}{relax}
12261 }
```

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.

```
12262 \AtBeginDocument{
12263     \NewEnviron{equation}%
12264         {\LWR@doequation{\BODY}{equation}}%
12265         [\LWR@doendequation{equation}]
12266
12267     \LetLtxMacro\LWR@equationnormal\equation
12268     \LetLtxMacro\endLWR@equationnormal\endequation
12269 }% AtBeginDocument
```

equation* (*env.*)

```
12270 \AtBeginDocument{
12271     \NewEnviron{equation*}%
12272         {\LWR@doequation{\BODY}{equation*}}%
12273         [\LWR@doendequation{equation*}]
12274
12275     \csletcs{LWR@equationnormalstar}{equation*}
12276     \csletcs{LWR@endequationnormalstar}{endequation*}
12277 }% AtBeginDocument
```

Remember the "less" version of equation, which uses MATHJAXand alt tags, but does not support complicated contents such as some Ti*k*Z expressions.

```
12278 \AtBeginDocument{
12279     \LetLtxMacro\LWR@equationless\equation
12280     \LetLtxMacro\endLWR@equationless\endequation
12281     \csletcs{LWR@equationlessstar}{equation*}
12282     \csletcs{LWR@endequationlessstar}{endequation*}
12283 }
```

83.6 \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 Ti*k*Z pictures, compilation will fail.

\displaymathother   When selecting \displaymathother, it is assumed that the contents are more
MATHJAX unsupported   complicated than "pure" math. An example is an elaborate Ti*k*Z 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 Ti*k*Z pictures are more likely to compile successfully.

\displaymathnormal   Use when display math environments have simple math which is to sent to Math-
simple math objects   Jax or included in HTML alt tags.

```
12284 \newcommand*{\displaymathnormal}{%
12285     \ifbool{LWR@origmathjax}{\booltrue{mathjax}}{\boolfalse{mathjax}}%
12286     \LetLtxMacro\[\LWR@openbracketnormal%
12287     \LetLtxMacro\]\LWR@closebracketnormal%
12288     \LetLtxMacro\displaymath\LWR@displaymathnormal%
12289     \LetLtxMacro\enddisplaymath\endLWR@displaymathnormal%
12290     \LetLtxMacro\equation\LWR@equationnormal%
12291     \LetLtxMacro\endequation\endLWR@equationnormal%
12292     \csletcs{equation*}{LWR@equationnormalstar}%
12293     \csletcs{endequation*}{LWR@endequationnormalstar}%
12294 }
```

\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.

```
12295 \newcommand*{\displaymathother}{%
12296     \boolfalse{mathjax}%
12297     \LetLtxMacro\displaymath\LWR@displaymathother%
12298     \LetLtxMacro\enddisplaymath\endLWR@displaymathother%
12299     \LetLtxMacro\[\LWR@displaymathother%
12300     \LetLtxMacro\]\endLWR@displaymathother%
12301     \LetLtxMacro\equation\LWR@equationother%
12302     \LetLtxMacro\endequation\endLWR@equationother%
12303     \csletcs{equation*}{displaymath}%
12304     \csletcs{endequation*}{enddisplaymath}%
12305 }
```

```
12306 \end{warpHTML}
```

for PRINT output: `12307 \begin{warpprint}`

Print-mode versions:

```
12308 \newcommand*{\displaymathnormal}{}
12309 \newcommand*{\displaymathother}{}
12310 \newcommand*{\theMathJaxsubequations}{0}
12311 \newcommand*{\theMathJaxsection}{}
12312 \newcommand*{\theMathJaxequation}{\arabic{equation}}
```

```
12313 \end{warpprint}
```

for HTML output: `12314 \begin{warpHTML}`

83.7 AMS Math environments

83.7.1 Support macros

LWR@amsmultline (*bool*)   True if processing a multline environment.

To compensate for `multline`-spefific 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`.

```
12315 \newbool{LWR@amsmultline}
12316 \boolfalse{LWR@amsmultline}
```

`\LWR@beginhideamsmath` Starts hiding LaTeX math inside an HTML comment.

```
12317 \newcommand*{\LWR@beginhideamsmath}{
12318     \LWR@stoppars
12319     \LWR@origtilde\LWR@orignewline
12320     \LWR@htmlopencomment
12321
12322     \begingroup
12323     \LWR@restoreorigformatting
```

Temporarily prevent underfull `\hbox` warnings.

```
12324     \hbadness=10000\relax%
```

```
12325     \booltrue{LWR@insidemathcomment}
12326 }
```

`\LWR@endhideamsmath` Ends hiding LaTeX math inside an HTML comment.

```
12327 \newcommand*{\LWR@endhideamsmath}{
12328     \endgroup
12329
12330     \LWR@htmlclosecomment
12331     \boolfalse{LWR@insidemathcomment}
12332     \LWR@orignewline
12333     \LWR@startpars
12334 }
```

83.7.2 Environment patches

The **amsmath** environments already collect their contents in `\@envbody` for further processing. `eqnarray` is not an $\mathcal{AMS}$ package, and thus requires special handling.

For SVG math: Each environment is encapsulated inside a `lateximage` environment, 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 $\mathcal{AMS}$ 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 $\mathcal{AMS}$ environment and thus its body is not automatically captured, so the **environ** package is used to capture the environment into `\BODY`.

```
12335 \let\LWR@origeqnarray\eqnarray
12336 \let\LWR@origendeqnarray\endeqnarray
```

To remember whether the starred environment was used, and thus whether to number the equations:

```
12337 \newbool{LWR@numbereqnarray}
12338 \booltrue{LWR@numbereqnarray}
```

Common code used by eqnarray and Beqnarray (from **fancybox**):

```
12339 \newcommand{\LWR@eqnarrayfactor}{%
```

If mathjax or FormatWP, print the LATEX expression:

```
12340    \ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
12341    {%
```

If MATHJAX, the environment contents (the \BODY) are executed in a HTML comment to trigger the correct equation number increment (if not starred), then are included verbatim in the output for MATHJAX to interpret:

```
12342        \LWR@syncmathjax%
12343        \boolfalse{LWR@amsmultline}%
12344        \ifbool{LWR@numbereqnarray}%
12345        {%
```

If numbering the equations, execute a copy inside an HTML comment block:

```
12346            \LWR@beginhideamsmath%
12347            \LWR@origeqnarray%
12348            \BODY%
12349            \LWR@origendeqnarray%
12350            \LWR@endhideamsmath%
```

Then print the (sanitized) contents to the output for MATHJAX to interpret:

```
12351            \LWR@addmathjax{eqnarray}{\BODY}%
12352        }%
12353        {% not LWR@numbereqnarray
```

If not numbering equations, just create the contents for MATHJAX:

```
12354            \LWR@addmathjax{eqnarray*}{\BODY}%
12355        }% LWR@numbereqnarray
12356    }% mathjax
12357    {% not mathjax
12358        \ifbool{LWR@numbereqnarray}%
12359        {%
```

For numbered SVG equations, first create a lateximage with an alt attribute containing sanitized copy of the source code:

```
12360            \begin{BlockClass}{displaymathnumbered}%
12361            \LWR@newautoidanchor%
12362            \booltrue{LWR@indisplaymathimage}%
12363        \begin{lateximage}[(\LWR@startingequationtag\textendash\LWR@equationtag)%
12364                \LWR@addmathjax{eqnarray}{\BODY}]*(math)%
```

Support for **xfakebold**:

```
12365            \LWR@applyxfakebold%
```

Create the image contents using an actual eqnarray:

```
12366            \LWR@origeqnarray%
12367            \BODY%
12368            \LWR@origendeqnarray%
12369            \end{lateximage}%
12370            \end{BlockClass}%
12371        }%
12372       {% not LWR@numbereqnarray
```

If not numbered, do the same, but an extra \nonumber seems to be required:

```
12373            \begin{BlockClass}{displaymath}%
12374            \LWR@newautoidanchor%
12375            \booltrue{LWR@indisplaymathimage}%
12376            \begin{lateximage}[\LWR@addmathjax{eqnarray*}{\BODY}]*(math)%
```

Support for xfakebold:

```
12377            \LWR@applyxfakebold%

12378            \def\@eqncr{\nonumber\@seqncr}
12379            \csuse{LWR@origeqnarray}%
12380            \BODY%
12381            \nonumber\csuse{LWR@origendeqnarray}%
12382            \end{lateximage}%
12383            \end{BlockClass}%
12384        }% LWR@numbereqnarray
12385    }% not mathjax
```

Default to number equations in the future:

```
12386    \booltrue{LWR@numbereqnarray}%
```

Clear the single-use alt text:

```
12387    \gdef\LWR@ThisAltText{}%
12388 }
```

eqnarray itself is made with a blank line before and after to force it to be on its own line:

```
12389 \RenewEnviron{eqnarray}
12390 {%
12391
12392 \LWR@eqnarrayfactor
12393
12394 }
```

The starred version is patched to turn off the numbering:

```
12395 \csgpreto{eqnarray*}{\boolfalse{LWR@numbereqnarray}}
```

```
12396 \end{warpHTML}
```

84 Lateximages

84.1 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 81.)

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 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}
```

84.2   Support counters and macros

12397 \begin{warpHTML}

LWR@lateximagenumber (*Ctr*)   Sequence the images.

```
12398 \newcounter{LWR@lateximagenumber}
12399 \setcounter{LWR@lateximagenumber}{0}
```

LWR@lateximagedepth (*Ctr*)   Do not create \lateximage inside of \lateximage.

```
12400 \newcounter{LWR@lateximagedepth}
12401 \setcounter{LWR@lateximagedepth}{0}
```

A few utility macros to write special characters:

```
12402 \edef\LWR@hashmark{\string#} % for use in \write
12403 \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.

```
12404 \newcounter{LWR@LIpage}
```

```
12405 \end{warpHTML}
```

84.3   Font size

12406 \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.*

```
12407 \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.

```
12408 \newcommand*{\LateximageFontScale}{1}
```

```
12409 \end{warpall}
```

84.4   Equation numbers

12410 \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.

```
12411 \newcounter{LWR@startingequation}
12412
12413 \@ifundefined{chapter}
12414 {
12415 \renewcommand{\theLWR@startingequation}{%
12416     \arabic{LWR@startingequation}%
12417 }
12418 }
12419 {% chapter defined
12420 \renewcommand{\theLWR@startingequation}{%
12421     \ifnumcomp{\value{chapter}}{>}{0}{\arabic{chapter}.}{}%
12422     \arabic{LWR@startingequation}%
12423 }
12424 }
```

LWR@isstartingequation (*bool*)   True for the first equation tag, false for later tags in the same environment.

```
12425 \newbool{LWR@isstartingequation}
```

\LWR@startingequationtag   Prints the starting equation number or tag.

```
12426 \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.

```
12427 \newcommand*{\LWR@equationtag}{}
```

Only if svg math, patch \tag after packages have loaded, in case someone else modified \tag.

```
12428 \AtBeginDocument{
12429
12430 \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.

```
12431 \NewDocumentCommand{\LWR@remembertag}{m}{%
12432     \ifbool{LWR@isstartingequation}%
12433     {%
12434         \global\boolfalse{LWR@isstartingequation}%
12435         \xdef\LWR@startingequationtag{#1}%
12436     }{}%
```

```
12437        \xdef\LWR@equationtag{#1}%
12438 }%


12439 }% not mathjax
12440 }% AtBeginDocument
```

84.5 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.

```
12441 \newcommand*{\LWR@amsmathbody}[1]
12442 {%
12443        \textbackslash\{begin\}\{#1\} % extra space
12444        \LWR@HTMLsanitizeexpanded{\detokenize\expandafter{\the\@envbody}}%
12445        \textbackslash\{end\}\{#1\}%
12446 }
```

\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.

```
12447 \newcommand*{\LWR@amsmathbodynumbered}[1]
12448 {%
12449        \ifnumcomp{\value{LWR@startingequation}}{=}{\value{equation}}%
12450            {(\LWR@equationtag)}%
12451            {(\LWR@startingequationtag\textendash\LWR@equationtag)} % extra space
12452        \LWR@amsmathbody{#1} % extra space
12453 }
```

84.6 lateximage environment

\LWR@lateximage@oneimageb {⟨*1: alt text*⟩} {⟨*2: filename*⟩} {⟨*3: css style*⟩} {⟨*4: aria role*⟩} Creates the image
for the lateximage.

```
12454 \newcommand{\LWR@lateximage@oneimageb}[4]{%
12455        \LWR@subinlineimage{#1}{lateximage}%
12456        {%
12457            \LWR@print@mbox{%
12458                \LWR@ImagesDirectory\OSPathSymbol%
12459                #2%
12460            }%
12461        }{svg}{#3}{#4}%
12462 }
```

\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.

```
12463 \newcommand{\LWR@lateximage@oneimage}[5]{%
12464        \LWR@traceinfo{LWR@lateximage@oneimage !#1!#2!#3!#4!#5!}%
12465        \ifdefvoid{\LWR@ThisAltText}{%
```

```
12466          \IfBooleanTF{#4}{%
12467              \LWR@lateximage@oneimageb{#1}{#2}{#3}{#5}%
12468          }{%
12469              \LWR@lateximage@oneimageb%
12470                  {\AltTextOpen#1\AltTextClose}%
12471                  {#2}{#3}{#5}%
12472          }%
12473      }{%
12474          \LWR@lateximage@oneimageb%
12475              {\AltTextOpen\LWR@ThisAltText\AltTextClose}%
12476              {#2}{#3}{#5}%
12477      }%
12478 }
```

lateximage (*env.*) * [⟨*2: <alt> tag*⟩] * [⟨*4: add'l hashing*⟩] [⟨*5: css style*⟩] (⟨*6: 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.

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 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.

*_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.

```
12479 \catcode'\$=\active%
12480
12481 \NewDocumentEnvironment{lateximage}{s O{\ImageAltText} s O{} O{} D(){}}%
12482 {%
12483 \LWR@traceinfo{lateximage !#1!#2!#3!#4!#5!#6!}%
12484 \LWR@traceinfo{lateximage: starting on \jobname.pdf page \arabic{page}}%
12485 \LWR@traceinfo{lateximage: entering depth is \arabic{LWR@lateximagedepth}}%
```

Nested `lateximages` remain one large `lateximage`:

```
12486 \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
```

If nesting inside an already-existing lateximage, simply record one more level. $\mathcal{AMS}$ packages redefine \addtocounter to do nothing if inside a \text, so lower-level TEX macros are used for tracking nested `lateximages`.

```
12487 {%
12488 %        \addtocounter{LWR@lateximagedepth}{1}%
12489       \global\advance\c@LWR@lateximagedepth 1\relax% Due to AmS \text macro.
12490 }%
```

Otherwise, this is the outer-most lateximage:

```
12491 {% start of outer-most lateximage
```

Remember the next equation number to be allocated, in case it must be printed in a multi-equation environment:

```
12492       \LWR@traceinfo{lateximage: starting outer-most lateximage}%
12493       \setcounter{LWR@startingequation}{\value{equation}}%
12494       \addtocounter{LWR@startingequation}{1}%
12495       \booltrue{LWR@isstartingequation}%
12496       \let\LWR@startingequationtag\theLWR@startingequation%
```

The default equation tag, unless overwritten by \tag:

```
12497       \let\LWR@equationtag\theequation%
```

Starting a new lateximage:

```
12498       \addtocounter{LWR@lateximagenumber}{1}%
12499       \LWR@traceinfo{lateximage: LWR@lateximagenumber is \arabic{LWR@lateximagenumber}}%
```

While inside a lateximage, locally do not use mathjax:

```
12500       \boolfalse{mathjax}%
```

Be sure that are doing a paragraph:

```
12501       \LWR@ensuredoingapar%
```

Inside the `lateximage`, temporarily prevent underfull \hbox warnings.

```
12502       \hbadness=10000\relax%
```

Next file:

```
12503       \addtocounter{LWR@externalfilecnt}{1}%
12504       \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:

```
12505       \setcounterpageref{LWR@LIpage}{%
12506           LWRlateximage-\BaseJobname-\arabic{LWR@lateximagenumber}%
```

```
12507    }%
12508    \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:

```
12509    \LWR@htmltag{span\LWR@indentHTML%
12510        id=\textquotedbl{}%
12511            lateximage-\BaseJobname-\arabic{LWR@lateximagenumber}%
12512            \textquotedbl\LWR@indentHTML
12513        class=\textquotedbl{}lateximagesource\textquotedbl\LWR@orignewline
12514    }%
```

Write instructions to the <ImagesDirectory>.txt file:

```
12515    \LWR@traceinfo{lateximage: about to write to \BaseJobname-images.txt}%
12516    \IfBooleanTF{#1}% starred
12517    {% hash
```

Compute and save the hashed file name for later use:

```
12518        \ifdefvoid{\LWR@ThisAltText}{%
12519            \IfBooleanTF{#3}{%
12520                \edef\LWR@hashedname{%
12521                    \LWR@mdfive{\detokenize\expandafter{#2}-!-#4}%
12522                }%
12523            }{%
12524                \edef\LWR@hashedname{%
12525                 \LWR@mdfive{\detokenize\expandafter{\AltTextOpen#2\AltTextClose}-!-#4}%
12526                }%
12527            }%
12528        }{%
12529            \edef\LWR@hashedname{%
12530             \LWR@mdfive{\detokenize\expandafter{\AltTextOpen\LWR@ThisAltText\AltTextClose}-!-#4}%
12531            }%
12532        }%
12533        \LWR@traceinfo{lateximage: hash is \LWR@hashedname}%
```

Write the page, hashing, and hashed name:

```
12534        \immediate\write\LWR@lateximagesfile{%
12535            |\arabic{LWR@LIpage}|true|\LWR@hashedname|%
12536        }%
12537    }% hash
12538    {% no hash
```

No hash, so write the page, no hashing, and the image number:

```
12539        \LWR@traceinfo{lateximage: hash false}%
12540        \immediate\write\LWR@lateximagesfile{%
12541        |\arabic{LWR@LIpage}|false|\LWR@ImagesName\arabic{LWR@externalfilecnt}|%
12542        }%
12543    }% no hash
```

Place an open comment tag. This will hide any traces of the lateximage PDF page which were picked up by *pdftotext*.

```
12544    \LWR@traceinfo{lateximage: about to create open comment}%
12545    \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.

```
12546     \addtocounter{LWR@lateximagedepth}{1}%
```

Start the new PDF page:

```
12547     \LWR@traceinfo{lateximage: about to create a new page}%
12548     \LWR@maybe@orignewpage%
```

If the current page is larger, typeset the image in a "standard" width page and font size:

```
12549     \LWR@traceinfo{lateximage: about to create minipage}%
12550     \setcounter{LWR@mpfootnote@store}{\value{mpfootnote}}
12551     \ifdimless{\linewidth}{6in}{%
12552         \LWR@print@minipage{\linewidth}%
12553     }{%
12554         \LWR@print@minipage{6in}%
12555     }%

12556     \ifnumgreater{\value{LWR@minipage@depth}}{0}%
12557         {\setcounter{mpfootnote}{\value{LWR@mpfootnote@store}}}%
12558         {}%
12559     \@nameuse{LWR@print@\LateximageFontSizeName}%
```

Temporarily restore formatting to its PDF definitions: Do not produce HTML tags for \hspace, etc. inside a lateximage.

```
12560     \LWR@traceinfo{lateximage: about to temporarily restore formatting}%
12561     \LWR@restoreorigformatting%
```

If not inside a minipage, use full-page footnotes instead of minipage footnotes. These become HTML footnotes.

```
12562     \ifnumgreater{\value{LWR@minipage@depth}}{0}%
12563         {}%
12564         {%
12565             \def\@mpfn{footnote}%
12566             \def\thempfn{\thefootnote}%
12567             \LetLtxMacro\@footnotetext\LWR@footnotetext%
12568         }%
```

Create the LWRlateximage<number> label:

```
12569     \LWR@traceinfo{lateximage: about to create label}%
12570     \LWR@orig@label{LWRlateximage-\BaseJobname-\arabic{LWR@lateximagenumber}}%
12571     \LWR@traceinfo{lateximage: finished creating the label}%
```

Adjust the rule color to match HTML:

```
12572     \ifdefvoid{\LWR@ruleHTMLcolor}{}{%
12573         \LWR@print@arrayrulecolor[HTML]{\LWR@ruleHTMLcolor}%
12574     }%
```

Enable print-mode math functions:

```
12575      \LetLtxMacro$\LWR@origdollar%
12576      \catcode`\$=3% math shift
12577      \LetLtxMacro\(\LWR@origopenparen%
12578      \LetLtxMacro\)\LWR@origcloseparen%
```

Only enable print-mode display math if are not already inside display math:

```
12579      \ifbool{LWR@indisplaymathimage}{}{% not in display math
12580          \LetLtxMacro\[\LWR@origopenbracket%
12581          \LetLtxMacro\]\LWR@origclosebracket%
12582          \let\equation\LWR@orig@equation%
12583          \let\endequation\LWR@orig@endequation%
12584          \csletcs{equation*}{LWR@orig@equation*}%
12585          \csletcs{endequation*}{LWR@orig@endequation*}%
12586      }% not in display math
```

For chemformula:

```
12587      \LetLtxMacro\LWR@newsingledollar$%
12588      \LetLtxMacro\LWR@newsingledollar$% syntax highlighting
```

```
12589 }% end of outer-most lateximage
12590 \LWR@traceinfo{lateximage: finished start of environment}%
12591 }% end of \begin{lateximage}
```

\endlateximage   When the lateximage environment closes:

```
12592 {% start of \end{lateximage}
12593 \LWR@traceinfo{lateximage: starting end of lateximage}%
```

Nested more than one deep?

```
12594 \LWR@traceinfo{lateximage: internal depth was \arabic{LWR@lateximagedepth}}%
12595 \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 $\mathcal{AMS}$ \text change of \addtocounter.

```
12596 {%
12597      \LWR@traceinfo{lateximage: unnesting}%
12598      \global\advance\c@LWR@lateximagedepth -1\relax%
12599 }%
```

If this is the outer-most lateximage:

```
12600 {% end of outer-most lateximage
```

Finish the lateximage minipage and start a new PDF page:

```
12601 \LWR@traceinfo{lateximage: ending outer-most lateximage}%
12602      \endLWR@print@minipage%
12603      \LWR@maybe@orignewpage%
```

Close the HTML comment which encapsulated any traces of the lateximage picked up by *pdftotext*:

```
12604      \LWR@print@vspace*{.5\baselineskip}%
```

```
12605        \LWR@htmlclosecomment%
12606        \LWR@traceinfo{lateximage: The page after the image is \arabic{page}}%
```

Create a link to the lateximage, allowing its natural height:

```
12607        \LWR@traceinfo{about to LWR@lateximage@oneimage !#2!}%
12608        \IfBooleanTF{#1}% starred
12609        {% hash
12610            \LWR@lateximage@oneimage{#2}{\LWR@hashedname}{#5}{#3}{#6}%
12611        }% hash
12612        {% no hash
12613         \LWR@lateximage@oneimage{#2}{\LWR@ImagesName\theLWR@externalfilecnt}{#5}{#3}{#6}%
12614        }% no hash
```

Be sure that are doing a paragraph:

```
12615        \LWR@ensuredoingapar%
```

Close the HTML span which has the *pdftotext* comment and also the link to the
. svg image:

```
12616        \LWR@htmltag{/span}%
12617        \ifbool{HTMLDebugComments}{%
12618            \LWR@htmlcomment{End of lateximage}%
12619        }{}%
```

Undo one lateximage level. This is not inside an $\mathcal{AMS}$ \text, so regular
\addtocounter may be used here.

```
12620        \addtocounter{LWR@lateximagedepth}{-1}%
```

Clear the single-use alt text:

```
12621        \gdef\LWR@ThisAltText{}%
12622 }% end of outer-most lateximage
12623 \LWR@traceinfo{lateximage: exiting depth is \arabic{LWR@lateximagedepth}}%
12624 \LWR@traceinfo{lateximage: done}%
12625 }%
12626 \catcode'\$=3% math shift
12627 \end{warpHTML}
```

```
12628 \begin{warpprint}
```

lateximage (*env.*) * [⟨<alt> *tag*⟩] * [⟨*add'l hashing*⟩] [⟨*css style*⟩]

Ignored in print mode.

```
12629 \NewDocumentEnvironment{lateximage}{s o s o o d()}
12630      {}{}
```

```
12631 \end{warpprint}
```

85   center, flushleft, flushright

```
12632 \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.

```
12633 \newenvironment*{LWR@HTML@center}
12634 {%
12635     \LWR@forcenewpage%
12636     \ifbool{FormatWP}%
12637         {\BlockClass[\LWR@print@mbox{text-align:center}]{center}}%
12638         {\BlockClass{center}}%
12639 }
12640 {\endBlockClass}
12641
12642 \LWR@formattedenv{center}
```

flushright (*env.*)

```
12643 \newenvironment*{LWR@HTML@flushright}
12644 {%
12645     \LWR@forcenewpage%
12646     \ifbool{FormatWP}%
12647         {\BlockClass[\LWR@print@mbox{text-align:right}]{flushright}}%
12648         {\BlockClass{flushright}}%
12649 }
12650 {\endBlockClass}
12651
12652 \LWR@formattedenv{flushright}
```

flushleft (*env.*)

```
12653 \newenvironment*{LWR@HTML@flushleft}
12654 {%
12655     \LWR@forcenewpage%
12656     \ifbool{FormatWP}%
12657         {\BlockClass[\LWR@print@mbox{text-align:left}]{flushleft}}%
12658         {\BlockClass{flushleft}}%
12659 }
12660 {\endBlockClass}
12661
12662 \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

```
12663 \newcommand*{\LWR@HTML@centering}{%
12664     \ifbool{HTMLDebugComments}{%
12665         \LWR@htmlcomment{centering}%
12666     }{}%
12667 }
12668 \LWR@formatted{centering}
```

\raggedleft

```
12669 \newcommand*{\LWR@HTML@raggedleft}{%
```

```
12670        \ifbool{HTMLDebugComments}{%
12671            \LWR@htmlcomment{raggedleft}%
12672        }{}%
12673 }
12674 \LWR@formatted{raggedleft}
```

\raggedright

```
12675 \newcommand*{\LWR@HTML@raggedright}{%
12676        \ifbool{HTMLDebugComments}{%
12677            \LWR@htmlcomment{raggedright}%
12678        }{}%
12679 }
12680 \LWR@formatted{raggedright}
```

\leftline {⟨*text*⟩}

```
12681 \renewcommand{\leftline}[1]{\begin{flushleft}#1\end{flushleft}}
```

\centerline {⟨*text*⟩}

```
12682 \renewcommand{\centerline}[1]{\begin{center}#1\end{center}}
```

\rightline {⟨*text*⟩}

```
12683 \renewcommand{\rightline}[1]{\begin{flushright}#1\end{flushright}}
```

```
12684 \end{warpHTML}
```

86 Preloaded packages

12685 \begin{warpHTML}

If the given package was loaded before or by lwarp, load the lwarp version as well.

\LWR@PreloadedPackage {⟨*packagename*⟩}

```
12686 \newcommand*{\LWR@PreloadedPackage}[1]{%
12687        \IfPackageLoadedTF{#1}%
12688            {%
12689                \AtBeginDocument{
12690                    \LWR@origRequirePackage{lwarp-#1}%
12691                }
12692            }%
12693            {}%
12694 }
```

If inputtrc was loaded before lwarp, as is usually done, explicitly load the lwarp patches now:

```
12695 \LWR@PreloadedPackage{inputtrc}
```

If **textcomp** was loaded before **lwarp**, perhaps as part of the font-related packages, explicitly load the lwarp patches now:

```
12696 \LWR@PreloadedPackage{textcomp}
```

If **xunicode** was loaded before **lwarp**, perhaps as part of the font-related packages, explicitly load the lwarp patches now:

```
12697 \LWR@PreloadedPackage{xunicode}
```

If **graphics** or **graphicx** were loaded before **lwarp**, perhaps by **xunicode**, explicitly load the lwarp patches now:

```
12698 \LWR@PreloadedPackage{graphics}
12699 \LWR@PreloadedPackage{graphicx}
```

tagpdf-base may have been preloaded by **pdfmanagement-testphase**

```
12700 \LWR@PreloadedPackage{tagpdf-base}
```

scalefnt may have been preloaded by **babel**

```
12701 \LWR@PreloadedPackage{scalefnt}
```

fontaxes must be preloaded so that **lwarp** may patch it for HTML.

```
12702 \LWR@PreloadedPackage{fontaxes}
```

Various font packages which may be loaded before **lwarp**:

```
12703 \LWR@PreloadedPackage{cmbright}
12704 \LWR@PreloadedPackage{fourier}
12705 \LWR@PreloadedPackage{kpfonts}
12706 \LWR@PreloadedPackage{kpfonts-otf}
12707 \LWR@PreloadedPackage{libertinust1math}
12708 \LWR@PreloadedPackage{pxfonts}
12709 \LWR@PreloadedPackage{txfonts}
12710 \LWR@PreloadedPackage{txgreeks}
12711 \LWR@PreloadedPackage{newpxmath}
12712 \LWR@PreloadedPackage{newtxmath}
12713 \LWR@PreloadedPackage{newtxsf}
12714 \LWR@PreloadedPackage{mathalpha}
12715 \LWR@PreloadedPackage{unicode-math}
```

```
12716 \LWR@PreloadedPackage{realscripts}
```

nfssext-cfr may be preloaded by **cfm-lm** or related font packages.

```
12717 \LWR@PreloadedPackage{nfssext-cfr}
```

ulem may be preloaded by **ctex**, **ctexart**, and related classes.

```
12718 \LWR@PreloadedPackage{ulem}
```

```
12719 \LWR@PreloadedPackage{xetexko}
```

geometry is preloaded by lwarp, and perhaps by various classes.

12720 \LWR@PreloadedPackage{geometry}

plext is preloaded by some CJK classes.

12721 \LWR@PreloadedPackage{plext}

stfloats is preloaded by ltj* classes.

12722 \LWR@PreloadedPackage{stfloats}

lltjext is preloaded by ltj* classes.

12723 \LWR@PreloadedPackage{lltjext}

luatexko must be loaded before lwarp.

12724 \LWR@PreloadedPackage{luatexko}

12725 \end{warpHTML}

87 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 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: 12726 `\begin{warpHTML}`

Options for siunitx:

```
12727 \newrobustcmd{\LWR@siunitx@textcelsius}{\HTMLentity{deg}C}
12728 \newrobustcmd{\LWR@siunitx@textdegree}{\HTMLentity{deg}}
12729 \newrobustcmd{\LWR@siunitx@textprime}{\HTMLunicode{2032}}
12730 \newrobustcmd{\LWR@siunitx@textdblprime}{\HTMLunicode{2033}}
12731 \newrobustcmd{\LWR@siunitx@textplanckbar}{\text{\textit{\HTMLunicode{210F}}}}
12732
12733 \appto\LWR@restoreorigformatting{%
12734 \renewrobustcmd{\LWR@siunitx@textcelsius}{\text{\ensuremath{^\circ}C}}%
12735 \renewrobustcmd{\LWR@siunitx@textdegree}{\text{\ensuremath{^\circ}}}%
12736 \renewrobustcmd{\LWR@siunitx@textprime}{\text{\ensuremath{^\prime}}}%
12737 \renewrobustcmd{\LWR@siunitx@textdblprime}{\text{\ensuremath{^{\prime\prime}}}}%
12738 \renewrobustcmd{\LWR@siunitx@textplanckbar}{\text{\ensuremath{\hbar}}}%
12739 }
```

12740 `\end{warpHTML}`

for PRINT output: The print version of `\HTMLDeclareSIUnit`.

```
12741 \begin{warpprint}
12742 \NewDocumentCommand{\HTMLDeclareSIUnit}{o +m m}{}
12743 \end{warpprint}
```

88 Graphics print-mode modifications

88.1 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 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 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`.

file extensions

⚠ case sensitive

`\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 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
⚠ image not displayed

A complication occurs if a file of the same name exists elsewhere in the TEX tree, 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
⚠ viewport

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 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 incorporated into graphicx itself.)

HTML class

With HTML output, \includegraphics accepts an optional class=xyz keyval com-

bination, 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.

88.2 Print-mode modifications

for PRINT output: For print output, accept and then discard the new class key:

```
12744 \begin{warpprint}
12745 \define@key{Gin}{class}{}
```

Print-mode additions for the overpic package. See section 460 for the HTML version.

```
12746 \AtBeginDocument{
12747 \IfPackageLoadedTF{overpic}{
12748 \newcommand*{\overpicfontsize}{12}
12749 \newcommand*{\overpicfontskip}{14}
12750 }{}
12751 }
12752 \end{warpprint}
```

89 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: 12753 \begin{warpall}

After xparse may have been loaded …

12754 \AtBeginDocument{

… and *only* if xcolor was loaded:

12755 \IfPackageLoadedTF{xcolor}{
12756 \LWR@traceinfo{patching xcolor}

The print version:

\colorboxBlock \colorboxBlock is the same as \colorbox:

12757 \LetLtxMacro\colorboxBlock\colorbox

The original definition is reused by the new versions:

12758 \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.)

12759 \newsavebox{\LWR@colorminipagebox}
12760
12761 \NewDocumentCommand{\LWR@print@fcolorbox}{o m o m +m}{%
12762   \LWR@traceinfo{LWR@print@fcolorbox #2 #4}%

Pre-load the contents into an LR box so that they can be used inside a \fcolorbox:

12763   \begin{lrbox}{\LWR@colorminipagebox}%
12764   #5%
12765   \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.

12766   \ifstrequal{#4}{none}%
12767   {% #4 none
12768   \LWR@traceinfo{background is none}%
12769   {% scope the \colorlet
12770   \colorlet{LWR@currentcolor}{.}%
12771   \color{#2}%
12772   \fbox{%
12773   \color{LWR@currentcolor}%
12774   \usebox{\LWR@colorminipagebox}%
12775   }% fbox
12776   }% colorlet
12777   }% #4 none
12778   {% #4 not none
12779   \LWR@traceinfo{background not none}%

```
12780        \IfValueTF{#1}%
12781        {%
12782            \IfValueTF{#3}%
12783         {\LWR@orig@print@fcolorbox[#1]{#2}[#3]{#4}{\usebox{\LWR@colorminipagebox}}}%
12784         {\LWR@orig@print@fcolorbox[#1]{#2}{#4}{\usebox{\LWR@colorminipagebox}}}%
12785        }%
12786        {% no value #1
12787            \IfValueTF{#3}%
12788         {\LWR@orig@print@fcolorbox{#2}[#3]{#4}{\usebox{\LWR@colorminipagebox}}}%
12789          {\LWR@orig@print@fcolorbox{#2}{#4}{\usebox{\LWR@colorminipagebox}}}%
12790        }% no value #1
12791        }% #4 not none
12792        \LWR@traceinfo{LWR@print@fcolorbox done}%
12793 }
```

```
12794 \renewrobustcmd*{\fcolorbox}{\LWR@print@fcolorbox}%
```

\fcolorboxBlock [⟨*framemodel*⟩] {⟨*framecolor*⟩} [⟨*boxmodel*⟩] {⟨*boxcolor*⟩} {⟨*text*⟩}

> In print mode, \fcolorboxBlock is the same as \fcolorbox.

```
12795 \newcommand*{\LWR@print@fcolorboxBlock}{\LWR@print@fcolorbox}
```

```
12796 \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:

```
12797 \NewDocumentEnvironment{fcolorminipage}{o m o m O{c} O{} o m}
12798 {%
12799        \LWR@traceinfo{*** fcolorminipage: #2 #4 #8}%
```

> Pre-load the contents into an LR box so that they can be used inside a \fcolorbox:

```
12800        \begin{lrbox}{\LWR@colorminipagebox}%
```

> If inner alignment is not given, use the outer alignment instead:

```
12801        \IfValueTF{#7}%
12802        {\begin{minipage}[#5][#6][#7]{#8}}%
12803        {\begin{minipage}[#5][#6][#5]{#8}}%
12804 }%
12805 {%
12806        \end{minipage}%
12807        \end{lrbox}%
12808        \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.

```
12809        \ifstrequal{#4}{none}%
12810        {% #4 none
```

```
12811        {% scope the \colorlet
12812            \colorlet{LWR@currentcolor}{.}%
12813            \color{#2}%
12814            \fbox{%
12815                \color{LWR@currentcolor}%
12816                \usebox{\LWR@colorminipagebox}%
12817            }% fbox
12818        }% colorlet
12819    }% #4 none
12820    {% #4 not none
12821        \IfValueTF{#1}%
12822        {%
12823        \IfValueTF{#3}%
12824    {\LWR@orig@print@fcolorbox[#1]{#2}[#3]{#4}{\usebox{\LWR@colorminipagebox}}}%
12825    {\LWR@orig@print@fcolorbox[#1]{#2}{#4}{\usebox{\LWR@colorminipagebox}}}%
12826        }%
12827        {% no value #1
12828        \IfValueTF{#3}%
12829    {\LWR@orig@print@fcolorbox{#2}[#3]{#4}{\usebox{\LWR@colorminipagebox}}}%
12830        {\LWR@orig@print@fcolorbox{#2}{#4}{\usebox{\LWR@colorminipagebox}}}%
12831        }% no value #1
12832    }% #4 not none
12833    \LWR@traceinfo{*** finished end fcolorminipage}%
12834 }
```

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.

```
12835 \LWR@formatted{colorbox}
12836 \LWR@formatted{colorboxBlock}
12837 \LWR@formatted{fcolorbox}
12838 \LWR@formatted{fcolorboxBlock}
12839 \LWR@formattedenv{fcolorminipage}
```

```
12840 \LWR@traceinfo{xcolor patches done}
12841 }{}% xcolor loaded
12842 }% AtBeginDocument
```

```
12843 \end{warpall}
```

90 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:
```
12844 \begin{warpprint}
```

```
12845 \AtBeginDocument{
12846 \IfPackageLoadedTF{chemmacros}{
```

Env polymerdelims

```
12847 \DeclareDocumentEnvironment{polymerdelims}{}
12848    {}{}
```

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 193.5.

```
12849 \DeclareDocumentEnvironment{redoxreaction}{m m}
12850     {\rule{0pt}{#1}}{\rule[-#2]{0pt}{#2}}

12851 }{}% chemmacros
12852 }% AtBeginDocument

12853 \end{warpprint}
```

91 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
```
12854 \begin{warpHTML}
12855
12856 \AtEndPreamble{
12857     \RequirePackage{cleveref}
12858 }
12859
12860 \end{warpHTML}
```

92 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
```
12861 \begin{warpHTML}
12862
12863 \LetLtxMacro\LWR@orig@label\label% includes memoir, before cleveref
12864 \LetLtxMacro\label\LWR@new@label
12865
12866 \LetLtxMacro\LWR@orig@pageref\pageref
12867 \LetLtxMacro\pageref\LWR@new@pageref
12868
12869 \end{warpHTML}
```

93   `picture` environment

picture (*env.*)   The `picture` environment is enclosed inside a `\lateximage`.

picture (*env.*)

```
12871 \BeforeBeginEnvironment{picture}{\begin{lateximage}[picture]}
12872
12873 \AfterEndEnvironment{picture}{\end{lateximage}}

12874 \end{warpHTML}
```

94   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
`\IgnoreMinipageWidths`   used 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: 12875 \begin{warpHTML}

94.1 Computed lengths

\LWR@minipagewidth (*Len*)   Used to convert the width into printable units.

12876 \newlength{\LWR@minipagewidth}

\LWR@minipageheight (*Len*)   Used to convert the height into printable units.

12877 \newlength{\LWR@minipageheight}

94.2 Virtual page size

LWR@virtualpagedepth (*Ctr*)   Used to only reset the line width at the outermost minipage.

12878 \newcounter{LWR@virtualpagedepth}
12879 \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.

12880 \NewDocumentEnvironment{LWR@setvirtualpage}{s O{1}}{%
12881   \ifnumequal{\value{LWR@virtualpagedepth}}{0}{%
12882   \IfBooleanT{#1}{\LWR@maybe@orignewpage}%
12883   \setlength{\linewidth}{6in/#2}%
12884   \setlength{\textwidth}{6in}%
12885   \setlength{\textheight}{9in}%
12886   }{}%
12887   \addtocounter{LWR@virtualpagedepth}{1}%
12888 }
12889 {\addtocounter{LWR@virtualpagedepth}{-1}}

94.3 Footnote handling

Also see section 60 for other forms of footnotes. Minipage footnotes are gathered in section 60.5, and then placed into the document in section 94.4.

94.4 Minipage handling

LWR@minipagefullwidth (*bool*) Should the next minipage have no HTML width?

```
12890 \newbool{LWR@minipagefullwidth}
12891 \boolfalse{LWR@minipagefullwidth}
```

LWR@forceminipagefullwidth (*bool*) Should the next minipage have no HTML width? Used to force full width for all minipages in an environment such as tabular or multicols, where the actual width depends on the browser width. Controlled by \useminipagewidths and \ignoreminipagewidths.

```
12892 \newbool{LWR@forceminipagefullwidth}
12893 \boolfalse{LWR@forceminipagefullwidth}
```

\minipagefullwidth Requests that the next minipage have no width tag in HTML:

for HTML output:
```
12894 \newcommand*{\minipagefullwidth}{\global\booltrue{LWR@minipagefullwidth}}
```

\UseMinipageWidths Locally requests that minipage widths be honored.

```
12895 \newcommand*{\UseMinipageWidths}{\boolfalse{LWR@forceminipagefullwidth}}
```

\IgnoreMinipageWidths Locally requests that minipage widths be ignored.

```
12896 \newcommand*{\IgnoreMinipageWidths}{\booltrue{LWR@forceminipagefullwidth}}
12897 \end{warpHTML}
```

for PRINT output:
```
12898 \begin{warpprint}
12899 \newcommand*{\minipagefullwidth}{}
12900 \newcommand*{\UseMinipageWidths}{}
12901 \newcommand*{\IgnoreMinipageWidths}{}
12902 \end{warpprint}
```

for HTML output:
```
12903 \begin{warpHTML}
```

LWR@minipagethispar (*bool*) Has a minipage been seen this paragraph? If true, prevents paragraph tags around horizontal space between minipages.

```
12904 \newbool{LWR@minipagethispar}
12905 \boolfalse{LWR@minipagethispar}
```

LWR@minipage@depth (*Ctr*) Used to track whether to change footnote styles in a lateximage inside an HTML minipage.

```
12906 \newcounter{LWR@minipage@depth}
12907 \setcounter{LWR@minipage@depth}{0}
```

LWR@mpfootnote@store (*Ctr*) Used to maintain minipage footnote number while nesting inside a lateximage.

```
12908 \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.

```
12909 \NewDocumentEnvironment{LWR@HTML@sub@minipage}{m m m m}
12910 {%
12911 \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.

```
12912 \begin{LWR@setvirtualpage}*%
```

Save the requested width now that \linewidth, etc. are adjusted to virtual size.

```
12913 \setlength{\LWR@minipagewidth}{#4}%
12914 \ifnumequal{\value{LWR@virtualpagedepth}}{1}{%
12915     \addtolength{\LWR@minipagewidth}{3em}% room for frames
12916 }{}%
12917 \LWR@traceinfo{computed width is \LWR@printlength{\LWR@minipagewidth}}%
```

Compute height:

```
12918 \setlength{\LWR@minipageheight}{\textheight}% default unless specified
12919 \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.

```
12920 \ifbool{FormatWP}{\newline}{}%
12921 \LWR@stoppars%
```

If FormatWP, add a text frame:

```
12922 \ifbool{FormatWP}{%
12923
12924 \addtocounter{LWR@thisautoidWP}{1}%
12925 \LWR@htmltag{%
12926     div id=\textquotedbl%
12927         \LWR@print@mbox{autoidWP-\arabic{LWR@thisautoidWP}}%
12928     \textquotedbl\ % space
12929     class=\textquotedbl{}wpminipage\textquotedbl%
12930 }%
12931
12932 }{}%
```

Create the <div> tag with optional alignment style:

```
12933 \LWR@traceinfo{minipage: creating div class}%
12934 \LWR@htmltag{div class=\textquotedbl{}minipage\textquotedbl\ style=\textquotedbl%
12935 \ifthenelse{\equal{#1}{t}}{\LWR@print@mbox{vertical-align:bottom} ; }{}%
12936 \ifthenelse{\equal{#1}{c}}{\LWR@print@mbox{vertical-align:middle} ; }{}%
12937 \ifthenelse{\equal{#1}{b}}{\LWR@print@mbox{vertical-align:top} ; }{}%
12938 \ifthenelse{\equal{#3}{t}}{\LWR@print@mbox{justify-content:flex-start} ; }{}%
12939 \ifthenelse{\equal{#3}{c}}{\LWR@print@mbox{justify-content:center} ; }{}%
```

```
12940 \ifthenelse{\equal{#3}{b}}{\LWR@print@mbox{justify-content:flex-end} ; }{}%
12941 \ifthenelse{\equal{#3}{s}}{\LWR@print@mbox{justify-content:space-between} ; }{}%
```

Print the width and optional height styles:

```
12942 \LWR@traceinfo{minipage: about to print the width of \LWR@printlength{\LWR@minipagewidth}}%
12943 \ifbool{LWR@minipagefullwidth}%
12944 {\global\boolfalse{LWR@minipagefullwidth}}%
12945 {%
12946     \ifbool{LWR@forceminipagefullwidth}%
12947         {}%
12948         {%
12949             \ifdimequal{#4}{\linewidth}%
12950                 {}%
12951                 {width:\LWR@printlength{\LWR@minipagewidth} ; }%
12952         }%
12953 }%
12954 \LWR@traceinfo{minipage: about to print the height}%
12955 \ifblank{#2}{}{height:\LWR@printlength{\LWR@minipageheight} ; }%
12956 \textquotedbl%
12957 }%
```

Finish with an empty line to start the contents on a new line.

```
12958
12959 % 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.

```
12960 \ifnumequal{\value{LWR@virtualpagedepth}}{1}{%
12961     \addtolength{\LWR@minipagewidth}{-3em}% undo frame padding
12962 }{}%
12963 \setlength{\linewidth}{\LWR@minipagewidth}%
```

\raggedright cancels hyphenation, which will be done by HTML instead.

```
12964 \LWR@print@raggedright%
```

```
12965     \LWR@newautopagelabel{page}%
```

Set minipage footnotes:

```
12966 \def\@mpfn{mpfootnote}%
12967 \def\thempfn{\thempfootnote}\c@mpfootnote\z@%
12968 \let\@footnotetext\@mpfootnotetext%
```

Track depth for lateximage footnote type:

```
12969 \addtocounter{LWR@minipage@depth}{1}%
```

Resume paragraph tag handling for the contents of the minipage:

```
12970 \LWR@startpars%
12971 \ifboolexpr{bool{FormatWP} and bool{WPMarkMinipages}}{%
12972
12973 === begin minipage ===
12974
12975 }{}%
12976 \LWR@traceinfo{minipage: finished starting the minipage}%
12977 }% finished \minipage
12978 {% \endminipage
```

Print pending minipage footnotes:

```
12979 \LWR@printpendingmpfootnotes%
```

End the environment with closing tag:

```
12980 \ifboolexpr{bool{FormatWP} and bool{WPMarkMinipages}}{%
12981
12982 === end minipage ===
12983
12984 }{}%
12985 \LWR@stoppars%
12986
12987 \ifbool{FormatWP}{%
12988
12989 \LWR@htmlelementend{div}%
12990
12991 }{}%
```

Wrapup:

```
12992 \addtocounter{LWR@minipage@depth}{-1}%

12993 \LWR@htmldivclassend{minipage}%
12994
12995 \end{LWR@setvirtualpage}%
12996 \LWR@startpars%
12997 \ifbool{FormatWP}{\newline}{}%
```

Prevent paragraph tags around horizontal white space until the start of the next paragraph:

```
12998 \global\booltrue{LWR@minipagethispar}%
12999 \LWR@traceinfo{LWR@minipage: done}%
13000 }
13001
13002 \NewDocumentEnvironment{LWR@HTML@minipage}{O{t} O{} O{t} m}
13003     {\LWR@HTML@sub@minipage{#1}{#2}{#3}{#4}}
13004     {\endLWR@HTML@sub@minipage}
13005
13006 \LWR@formattedenv{minipage}
```

94.5   \parbox, \mbox, \makebox, \framebox, \fbox, \raisebox

\parbox [⟨*pos*⟩] [⟨*height*⟩] [⟨*inner-pos*⟩] {⟨*width*⟩} {⟨*text*⟩}

A parbox uses the minipage code:

```
13007 \NewDocumentCommand{\LWR@HTML@parbox}{O{t} O{} O{t} m +m}
13008 {
13009 \LWR@traceinfo{parbox of width #4}%
13010 \begin{minipage}[#1][#2][#3]{#4}%
13011 #5
13012 \end{minipage}%
13013 }
13014
13015 \LWR@formatted{parbox}
```

\mbox {⟨*text*⟩}   Nullified for HTML.

```
13016 \newcommand*{\LWR@HTML@mbox}[1]{{#1}}
13017
13018 \LWR@formatted{mbox}
```

\LWR@@makebox@paren {⟨*width*⟩} , {⟨*height*⟩}

Adds to the style in \LWR@temptwo.

```
13019 \NewDocumentCommand{\LWR@@makebox@paren}{m m}{%
13020 \IfValueTF{#2}{%
13021     \setlength{\LWR@tempwidth}{#1\unitlength}%
13022     \setlength{\LWR@tempheight}{#2\unitlength}%
13023     \appto{\LWR@temptwo}{%
13024         \LWR@print@mbox{width:\LWR@printlength{\LWR@tempwidth}} ; % space
13025         \LWR@print@mbox{height:\LWR@printlength{\LWR@tempheight}} ; % space
13026     }%
13027 }{%
13028     \PackageError{lwarp}%
13029         {(width,height) is missing a comma ',' character}%
13030         {\protect\makebox\space and \protect\framebox\space accept
13031             a size in the format (width,height).}%
13032 }%
13033 }
```

\LWR@@makebox@align {⟨*alignment character*⟩}

Adds to the style in \LWR@temptwo.

```
13034 \newcommand*{\LWR@@makebox@align}[1]{%
13035     \def\LWR@align{center}%
13036     \ifstrequal{#1}{l}{\def\LWR@align{left}}{}%
13037     \ifstrequal{#1}{r}{\def\LWR@align{right}}{}%
13038     \ifstrequal{#1}{s}{\def\LWR@align{justify}}{}%
13039     \appto{\LWR@temptwo}{%
13040         \LWR@print@mbox{text-align:\LWR@align} ; %
13041     }%
13042 }
```

\makebox (⟨*width,height*⟩) [⟨*width*⟩] [⟨*pos*⟩] {⟨*text*⟩}

```
13043 \NewDocumentCommand{\LWR@HTML@makebox}{>{\SplitArgument{1}{,}}d() o o +m}{%
```

Build the style depending on arguments:

```
13044    \begin{LWR@setvirtualpage}%
13045        \def\LWR@temptwo{}%
13046        \IfValueTF{#1}%
13047        {% (width,height) ..
13048            \LWR@@makebox@paren #1%
13049            \IfValueT{#2}%
13050            {% (width,height) [posn]
13051                \LWR@@makebox@align{#2}%
13052            }%
13053        }%
13054        {% [width]
13055            \IfValueT{#2}% [width]
13056            {%
13057                \setlength{\LWR@tempwidth}{#2}%
13058                \ifdimgreater{\LWR@tempwidth}{0pt}{%
13059                    \appto{\LWR@temptwo}{%
13060                        width:\LWR@printlength{\LWR@tempwidth} ; % space
13061                    }%
13062                }{}%
13063            }%
13064        }%
13065        \IfValueT{#3}%
13066        {% [width] [posn]
13067            \LWR@@makebox@align{#3}%
13068        }%
13069        \InlineClass[%
13070            \LWR@print@mbox{display:inline-block} ; %
13071            \LWR@temptwo%
13072        ]%
13073        {makebox}%
13074        {#4}%
13075    \end{LWR@setvirtualpage}%
13076 }
13077 \LWR@formatted{makebox}
```

\framebox (⟨*width,height*⟩) [⟨*width*⟩] [⟨*pos*⟩] {⟨*text*⟩}

```
13078 \NewDocumentCommand{\LWR@HTML@framebox}{d() o o +m}{%
13079    \fbox{\makebox(#1)[#2][#3]{#4}}%
13080 }
13081
13082 \LWR@formatted{framebox}
```

\LWR@forceminwidth {⟨*legth*⟩}

Sets \LWR@atleastonept to be at least 1pt.

```
13083 \newlength{\LWR@atleastonept}
13084
13085 \newcommand*{\LWR@forceminwidth}[1]{%
13086 \setlength{\LWR@atleastonept}{#1}%
13087 \ifthenelse{%
13088    \lengthtest{\LWR@atleastonept>0pt}\AND%
13089    \lengthtest{\LWR@atleastonept<1pt}%
13090 }%
13091    {\setlength{\LWR@atleastonept}{1pt}}%
13092    {}%
```

```
13093 }
```

\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.

```
13094 \newcommand*{\LWR@fboxstyle}{%
13095 \LWR@findcurrenttextcolor%
13096 \LWR@traceinfo{LWR@fboxstyle B}%
13097 border:\LWR@printlength{\LWR@atleastonept} solid \LWR@origpound\LWR@tempcolor ; %
13098 padding:\LWR@printlength{\fboxsep} ; %
13099 color:\LWR@origpound\LWR@tempcolor%
13100 }
```

\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:

```
13101 \newcommand{\LWR@HTML@fbox}[1]{%
13102     \LWR@traceinfo{HTML fbox}%
13103     \LWR@forceminwidth{\fboxrule}%
13104     \LWR@traceinfo{HTML fbox B}%
13105     \InlineClass[%
13106         \LWR@print@mbox{display:inline-block} ; %
13107         \LWR@fboxstyle%
13108     ]{fbox}{#1}%
13109     \LWR@traceinfo{HTML fbox: done}%
13110 }
```

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:

```
13111 \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.

```
13112 \end{warpHTML}
13113
13114 \begin{warpall}
13115 \let\fboxBlock\fbox
13116 \end{warpall}
13117
13118 \begin{warpHTML}
```

Next, an HTML version:

```
13119 \newcommand{\LWR@HTML@fboxBlock}[1]{%
13120 \LWR@forceminwidth{\fboxrule}%
13121 \LWR@stoppars%
13122 \begin{BlockClass}[\LWR@fboxstyle]{fboxBlock}
13123 #1
13124 \end{BlockClass}
13125 \LWR@startpars%
```

```
13126 }
13127
13128 \LWR@formatted{fboxBlock}
13129
13130 \end{warpHTML}
```

fminipage (*env.*) [⟨*align*⟩] [⟨*height*⟩] [⟨*align*⟩] {⟨*width*⟩}

Creates a framed HTML <div> around its contents.

for HTML & PRINT: Print version:

```
13131 \begin{warpall}
13132
13133 \newsavebox{\LWR@fminipagebox}
13134
13135 \NewDocumentEnvironment{fminipage}{O{t} o O{t} m}
13136 {%
```

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:

```
13137 \IfValueTF{#3}%
13138 {\def\LWR@thisalign{#3}}
13139 {\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.

```
13140 \IfValueTF{#2}%
13141 {\minipage[#1][#2+2\fboxsep+2\fboxrule][\LWR@thisalign]{#4+2\fboxsep+2\fboxrule}}%
13142 {\minipage[#1]{#4+2\fboxsep+2\fboxrule}}%
```

Capture the contents of the environment:

```
13143 \begin{lrbox}{\LWR@fminipagebox}%
```

Nest the contents inside an inner minipage of the desired size:

```
13144 \IfValueTF{#2}%
13145 {\minipage[#1][#2][\LWR@thisalign]{#4}}%
13146 {\minipage[#1]{#4}}%
13147 }
13148 {%
```

Close the inner minipage and the LR box with the contents:

```
13149 \endminipage%
13150 \end{lrbox}%
```

Create a frame around the contents of the environment:

```
13151 \fbox{\usebox{\LWR@fminipagebox}}%
```

The entire thing is placed inside the outer minipage:

```
13152 \endminipage%
13153 }
13154 \end{warpall}
```

HTML version:

```
13155 \begin{warpHTML}
13156
13157 \NewDocumentEnvironment{LWR@HTML@fminipage}{O{t} o O{t} m}
13158 {%
13159 \LWR@traceinfo{fminipage #1 #2 #3 #4}%
```

Locally change to the virtual page size before processing the requested sizes:

```
13160 \begin{LWR@setvirtualpage}*%
13161 \setlength{\LWR@tempwidth}{#4}%
13162 \IfValueT{#2}{\setlength{\LWR@tempheight}{#2}}%
```

Use a rule of at least one pixel in width:

```
13163 \LWR@forceminwidth{\fboxrule}%
```

```
13164 \LWR@stoppars%
```

```
13165 \begin{BlockClass}[%
13166 \LWR@fboxstyle ; %
13167 \IfValueT{#2}{height:\LWR@printlength{\LWR@tempheight} ; }%
13168 \ifbool{LWR@minipagefullwidth}%
13169 {\global\boolfalse{LWR@minipagefullwidth}}%
13170 {%
13171     \ifbool{LWR@forceminipagefullwidth}%
13172         {}%
13173         {%
13174             \ifdimequal{\LWR@tempwidth}{\linewidth}%
13175                 {}%
13176                 {width:\LWR@printlength{\LWR@tempwidth} ; }%
13177         }%
13178 }%
13179 ]{fminipage}%
13180 }
13181 {%
13182 \end{BlockClass}%
13183 \end{LWR@setvirtualpage}%
```

Prevent paragraph tags around horizontal white space until the start of the next paragraph:

```
13184 \global\booltrue{LWR@minipagethispar}%
13185 \LWR@traceinfo{fminipage done}%
13186 }
13187
13188 \LWR@formattedenv{fminipage}
```

\raisebox {⟨*raiselen*⟩} [⟨*height*⟩] [⟨*depth*⟩] {⟨*text*⟩}

```
13189 \NewDocumentCommand{\LWR@HTML@raisebox}{m o o m}{%
13190 #4%
```

```
13191 }
13192
13193 \LWR@formatted{raisebox}

13194 \end{warpHTML}
```

95   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 convert &, <, and > to proper HTML entities.

verbatim   The various verbatim-related environments do not convert &, <, and >, so care must be taken to avoid accidentally including valid HTML code inside these environments. Adding a space on either side may be sufficient.

For high-level block and inline custom css classes, see section 52.10.

for HTML & PRINT:`13195 \begin{warpall}`

FixSmallCaps (*bool*)   User may set FixSmallCaps to true if small caps are being incorrectly rendered as all caps.

```
13196 \newbool{FixSmallCaps}
13197 \boolfalse{FixSmallCaps}

13198 \end{warpall}
```

for HTML output:`13199 \begin{warpHTML}`

\emph {⟨*text*⟩}

```
13200 \DeclareRobustCommand{\LWR@HTML@emph}[1]{%
13201     {%
13202         \LWR@HTML@itshape%
13203         \LWR@htmlspan{em}{#1}%
13204     }%
13205 }
13206
13207 \LWR@formatted{emph}
```

\textmd {⟨*text*⟩}

```
13208 \DeclareRobustCommand{\LWR@HTML@textmd}[1]{%
13209     {%
13210         \LWR@HTML@mdseries%
13211         \InlineClass(font-weight:normal){textmd}{#1}%
13212     }%
```

```
13213 }
13214
13215 \LWR@formatted{textmd}
```

\textbf {⟨*text*⟩}

```
13216 \DeclareRobustCommand{\LWR@HTML@textbf}[1]{%
13217     {%
13218         \LWR@HTML@bfseries%
13219         \LWR@htmlspan{b}{#1}%
13220     }%
13221 }
13222
13223 \LWR@formatted{textbf}
```

\texteb {⟨*text*⟩}   From nfssext-cfr.

```
13224 \IfPackageLoadedTF{nfssext-cfr}{
13225 \DeclareRobustCommand{\LWR@HTML@texteb}[1]{%
13226     {%
13227         \LWR@HTML@ebweight%
13228         \InlineClass{texteb}{#1}%
13229     }%
13230 }
13231
13232 \LWR@formatted{texteb}
13233 }{% if not loaded
13234     \providerobustcmd{\texteb}[1]{}
13235 }
```

\textlg {⟨*text*⟩}   From nfssext-cfr.

```
13236 \IfPackageLoadedTF{nfssext-cfr}{
13237 \DeclareRobustCommand{\LWR@HTML@textlg}[1]{%
13238     {%
13239         \LWR@HTML@lgweight%
13240         \InlineClass{textlg}{#1}%
13241     }%
13242 }
13243
13244 \LWR@formatted{textlg}
13245 }{% if not loaded
13246     \providerobustcmd{\textlg}[1]{}
13247 }
```

\textrm {⟨*text*⟩}

```
13248 \DeclareRobustCommand{\LWR@HTML@textrm}[1]{%
13249     {%
13250         \LWR@HTML@rmfamily%
13251         \InlineClass(font-family:serif){textrm}{#1}%
13252     }%
13253 }
13254
13255 \LWR@formatted{textrm}
```

\textsf {⟨*text*⟩}

```
13256 \DeclareRobustCommand{\LWR@HTML@textsf}[1]{%
13257     {%
13258         \LWR@HTML@sffamily%
13259         \InlineClass(font-family:sans){textsf}{#1}%
13260     }%
13261 }
13262
13263 \LWR@formatted{textsf}
```

\texttt {⟨*text*⟩}

```
13264 \DeclareRobustCommand{\LWR@HTML@texttt}[1]{%
13265     {%
13266         \LWR@HTML@ttfamily%
13267         \LWR@htmlspan{kbd}{#1}%
13268     }%
13269 }
13270
13271 \LWR@formatted{texttt}
```

\textup {⟨*text*⟩}

```
13272 \DeclareRobustCommand{\LWR@HTML@textup}[1]{%
13273     {%
13274         \LWR@HTML@upshape%
13275         \InlineClass(font-style:normal){textup}{#1}%
13276     }%
13277 }
13278
13279 \LWR@formatted{textup}
```

\textit {⟨*text*⟩}

```
13280 \DeclareRobustCommand{\LWR@HTML@textit}[1]{%
13281     {%
13282         \LWR@HTML@itshape%
13283         \LWR@htmlspan{i}{#1}%
13284     }%
13285 }
13286
13287 \LWR@formatted{textit}
```

\textsc {⟨*text*⟩}

```
13288 \DeclareRobustCommand{\LWR@HTML@textsc}[1]{%
13289     {%
13290         \LWR@HTML@scshape%
13291         \InlineClass{textsc}{#1}%
13292     }%
13293 }
13294
13295 \LWR@formatted{textsc}
```

\textulc {⟨*text*⟩}   From fontaxes.

```
13296 \DeclareRobustCommand{\LWR@HTML@textulc}[1]{%
13297     {%
13298         \LWR@HTML@ulcshape%
13299         \InlineClass{textulc}{#1}%
13300     }%
13301 }
13302
13303 \LWR@formatted{textulc}
```

\textsi {⟨*text*⟩}

```
13304 \@ifundefined{textsi}{
13305     \LetLtxMacro\LWR@print@textsi\LWR@print@textsc
13306 }{}
13307
13308 \DeclareRobustCommand{\LWR@HTML@textsi}[1]{%
13309     {%
13310         \LWR@HTML@sishape%
13311         \textsc{\textit{#1}}%
13312 %       \InlineClass(
13313 %           font-style: italic;
13314 %           font-variant: small-caps ;
13315 %           font-variant-numeric: oldstyle-nums ;
13316 %       ){textsi}{#1}%
13317     }%
13318 }
13319
13320 \LWR@formatted{textsi}
```

\textsl {⟨*text*⟩}

```
13321 \DeclareRobustCommand{\LWR@HTML@textsl}[1]{%
13322     {%
13323         \slshape%
13324         \InlineClass(font-style:oblique){textsl}{#1}%
13325     }%
13326 }
13327
13328 \LWR@formatted{textsl}
```

\textssc {⟨*text*⟩}

```
13329 \newrobustcmd{\LWR@HTML@textssc}[1]{\textsc{#1}}
13330 \LWR@formatted{textssc}
```

\textnormal {⟨*text*⟩}

```
13331 \DeclareRobustCommand{\LWR@HTML@textnormal}[1]{%
13332         \LWR@HTML@mdseries%
13333         \LWR@HTML@rmfamily%
13334         \LWR@HTML@upshape%
13335         \LWR@HTML@ulcshape%
13336         \InlineClass(%
13337             font-weight: normal;
13338             font-family: serif;
13339             font-style: normal;
13340             font-variant: normal;
```

```
13341          font-variant-numeric: normal ;
13342        ){textnormal}{#1}%
13343 }
13344
13345 \LWR@formatted{textnormal}

13346 \FilenameNullify{%
13347        \LetLtxMacro\emph\@firstofone%
13348        \LetLtxMacro\textmd\@firstofone%
13349        \LetLtxMacro\textbf\@firstofone%
13350        \LetLtxMacro\texteb\@firstofone%
13351        \LetLtxMacro\textlg\@firstofone%
13352        \LetLtxMacro\textrm\@firstofone%
13353        \LetLtxMacro\textsf\@firstofone%
13354        \LetLtxMacro\texttt\@firstofone%
13355        \LetLtxMacro\textup\@firstofone%
13356        \LetLtxMacro\textit\@firstofone%
13357        \LetLtxMacro\textsc\@firstofone%
13358        \LetLtxMacro\textulc\@firstofone%
13359        \LetLtxMacro\textsi\@firstofone%
13360        \LetLtxMacro\textsl\@firstofone%
13361        \LetLtxMacro\textssc\@firstofone%
13362        \LetLtxMacro\textnormal\@firstofone%
13363 }
```

Remembers the current font family, series, and shape. fontaxes support is integrated here.

```
13364 \newcommand*{\LWR@f@family}{rm}
13365 \newcommand*{\LWR@f@series}{md}
13366 \newcommand*{\LWR@f@shape}{up}
13367 \newcommand*{\LWR@f@shapecaps}{ulc}
```

\LWR@textcurrentfont {⟨text⟩}

Prints the text with the current font choices. Avoids nesting repeated font selections.

```
13368 \newcounter{LWR@textcurrentfontdepth}
13369 \setcounter{LWR@textcurrentfontdepth}{0}
13370
13371 \newcommand*{\LWR@textcurrentfont}[1]{%
13372     \ifnumcomp{\value{LWR@textcurrentfontdepth}}{>}{0}%
13373        {%
13374            \addtocounter{LWR@textcurrentfontdepth}{1}%
13375            #1%
13376            \addtocounter{LWR@textcurrentfontdepth}{-1}%
13377        }%
13378        {%
13379            \addtocounter{LWR@textcurrentfontdepth}{1}%
13380            \ifboolexpr{%
13381                test {\ifdefstring{\LWR@f@family}{rm}} and
13382                test {\ifdefstring{\LWR@f@series}{md}} and
13383                test {\ifdefstring{\LWR@f@shape}{up}} and
13384                test {\ifdefstring{\LWR@f@shapecaps}{ulc}}
13385            }%
13386                {\InlineClass{textnormal}{#1}}%
13387                {%
13388                    \InlineClass{%
```

```
13389                                         text\LWR@f@family\LWR@origtilde{}%
13390                                         text\LWR@f@series\LWR@origtilde{}%
13391                                         text\LWR@f@shape\LWR@origtilde{}%
13392                                         text\LWR@f@shapecaps%
13393                                     }%
13394                                     {#1}%
13395                             }%
13396                 \addtocounter{LWR@textcurrentfontdepth}{-1}%
13397         }%
13398 }
```

LWR@blocktextcurrentfont (*env.*) Prints the contents with the current font choices.

```
13399 \newenvironment*{LWR@blocktextcurrentfont}{%
13400 \LWR@stoppars%
13401 \BlockClass{%
13402         text\LWR@f@family\LWR@origtilde{}%
13403         text\LWR@f@series\LWR@origtilde{}%
13404         text\LWR@f@shape\LWR@origtilde{}%
13405         text\LWR@f@shapecaps%
13406     }%
13407 }{\endBlockClass\LWR@startpars}
```

\mdseries

```
13408 \newrobustcmd*{\LWR@HTML@mdseries}{%
13409     \LWR@print@mdseries%
13410     \renewcommand*{\LWR@f@series}{md}%
13411 }
13412 \LWR@formatted{mdseries}
```

\bfseries

```
13413 \newrobustcmd*{\LWR@HTML@bfseries}{%
13414     \LWR@print@bfseries%
13415     \renewcommand*{\LWR@f@series}{bf}%
13416 }
13417 \LWR@formatted{bfseries}
```

\ebweight From **nfssext-cfr**.

```
13418 \IfPackageLoadedTF{nfssext-cfr}{
13419 \newrobustcmd*{\LWR@HTML@ebweight}{%
13420     \LWR@print@ebweight%
13421     \renewcommand*{\LWR@f@series}{eb}%
13422 }
13423 \LWR@formatted{ebweight}
13424 }{}
```

\lgweight From **nfssext-cfr**.

```
13425 \IfPackageLoadedTF{nfssext-cfr}{
13426 \newrobustcmd*{\LWR@HTML@lgweight}{%
13427     \LWR@print@lgweight%
13428     \renewcommand*{\LWR@f@series}{lg}%
```

```
13429 }
13430 \LWR@formatted{lgweight}
13431 }{}
```

\rmfamily

```
13432 \newrobustcmd*{\LWR@HTML@rmfamily}{%
13433     \LWR@print@rmfamily%
13434     \renewcommand*{\LWR@f@family}{rm}%
13435 }
13436 \LWR@formatted{rmfamily}
```

\sffamily

```
13437 \newrobustcmd*{\LWR@HTML@sffamily}{%
13438     \LWR@print@sffamily%
13439     \renewcommand*{\LWR@f@family}{sf}%
13440 }
13441 \LWR@formatted{sffamily}
```

\ttfamily

```
13442 \newrobustcmd*{\LWR@HTML@ttfamily}{%
13443     \LWR@print@ttfamily%
13444     \renewcommand*{\LWR@f@family}{tt}%
13445 }
13446 \LWR@formatted{ttfamily}
```

The following use \AtBeginDocument due to the LATEX core \reinstall@nfss@defs,
which redefines these \AtBeginDocument. See **texdoc source2e**.

\upshape

```
13447 \newrobustcmd*{\LWR@HTML@upshape}{%
13448     \LWR@print@upshape%
13449     \renewcommand*{\LWR@f@shape}{up}%
13450 }
13451 \AtBeginDocument{\LWR@formatted{upshape}}
```

\itshape

```
13452 \newrobustcmd*{\LWR@HTML@itshape}{%
13453     \LWR@print@itshape%
13454     \renewcommand*{\LWR@f@shape}{it}%
13455 }
13456 \AtBeginDocument{\LWR@formatted{itshape}}
```

\scshape   Note: \LWR@print@scshape is not used here since some fonts, such as erewhon,
           copy/paste as all-caps.

```
13457 \newrobustcmd*{\LWR@HTML@scshape}{%
13458     \ifbool{FixSmallCaps}{}{%
13459         \LWR@print@scshape%
13460     }%
```

```
13461        \renewcommand*{\LWR@f@shapecaps}{sc}%
13462 }
13463 \AtBeginDocument{\LWR@formatted{scshape}}
```

\ulcshape From fontaxes.

```
13464 \@ifundefined{ulcshape}{
13465        \LetLtxMacro\ulcshape\upshape
13466 }{}
13467 \newrobustcmd*{\LWR@HTML@ulcshape}{%
13468        \LWR@print@ulcshape%
13469        \renewcommand*{\LWR@f@shapecaps}{ulc}%
13470 }
13471 \AtBeginDocument{\LWR@formatted{ulcshape}}
```

\sishape

```
13472 \@ifundefined{sishape}{
13473        \LetLtxMacro\sishape\scshape
13474 }{}
13475 \newrobustcmd*{\LWR@HTML@sishape}{%
13476        \ifbool{FixSmallCaps}{}{%
13477            \LWR@print@sishape%
13478        }%
13479        \renewcommand*{\LWR@f@shape}{it}
13480        \renewcommand*{\LWR@f@shapecaps}{sc}%
13481 }
13482 \AtBeginDocument{\LWR@formatted{sishape}}
```

\slshape

```
13483 \newrobustcmd*{\LWR@HTML@slshape}{%
13484        \LWR@print@slshape%
13485        \renewcommand*{\LWR@f@shape}{sl}%
13486 }
13487 \AtBeginDocument{\LWR@formatted{slshape}}
```

\sscshape

```
13488 \newrobustcmd{\LWR@HTML@sscshape}{\LWR@HTML@scshape}
13489 \AtBeginDocument{\LWR@formatted{sscshape}}
```

\normalfont

```
13490 \newrobustcmd*{\LWR@HTML@normalfont}{\rmfamily\mdseries\upshape\ulcshape}
13491 \LWR@formatted{normalfont}
```

```
13492 \FilenameNullify{%
13493        \LetLtxMacro\rmfamily\@empty%
13494        \LetLtxMacro\sffamily\@empty%
13495        \LetLtxMacro\ttfamily\@empty%
13496        \LetLtxMacro\bfseries\@empty%
13497        \LetLtxMacro\ebweight\@empty%
13498        \LetLtxMacro\lgweight\@empty%
13499        \LetLtxMacro\mdseries\@empty%
13500        \LetLtxMacro\upshape\@empty%
```

```
13501    \LetLtxMacro\slshape\@empty%
13502    \LetLtxMacro\sishape\@empty%
13503    \LetLtxMacro\scshape\@empty%
13504    \LetLtxMacro\itshape\@empty%
13505    \LetLtxMacro\ulcshape\@empty%
13506    \LetLtxMacro\sscshape\@empty%
13507    \LetLtxMacro\normalfont\@empty%
13508 }
```

\sp {⟨*text*⟩}

>   For siunitx. Must work in math mode.

```
13509 \renewcommand{\sp}[1]{\text{<sup>#1</sup>}{}}
```

\sb {⟨*text*⟩}

>   For siunitx. Must work in math mode.

```
13510 \renewcommand{\sb}[1]{\text{<sub>#1</sub>}{}}
```

\textsuperscript {⟨*text*⟩}

```
13511 \newrobustcmd{\LWR@HTML@textsuperscript}[1]{\LWR@htmlspan{sup}{#1}}
13512 \LWR@formatted{textsuperscript}
```

\@textsuperscript {⟨*text*⟩}

```
13513 \newcommand{\LWR@HTML@@textsuperscript}[1]{\LWR@htmlspan{sup}{#1}}
13514 \LWR@formatted{@textsuperscript}
```

\textsubscript {⟨*text*⟩}

```
13515    \newrobustcmd{\LWR@HTML@textsubscript}[1]{\LWR@htmlspan{sub}{#1}}
13516    \LWR@formatted{textsubscript}
```

\@textsubscript {⟨*text*⟩}

```
13517    \newcommand{\LWR@HTML@@textsubscript}[1]{\LWR@htmlspan{sub}{#1}}
13518    \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.

```
13519 \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.

```
13520 \AtBeginDocument{\let\fup\textsuperscript}
```

\underline {⟨*text*⟩}

```
13521 \renewcommand{\underline}[1]{%
13522     \InlineClass%
13523         (text-decoration:underline; text-decoration-skip: auto)%
13524         {underline}{#1}%
13525 }
```

\LWR@overline {⟨*text*⟩}

```
13526 \newcommand{\LWR@overline}[1]{%
13527     \InlineClass%
13528         (text-decoration:overline; text-decoration-skip: auto)%
13529         {overline}{#1}%
13530 }
```

\LWR@currenttextcolor The color to use for text and \rule, defaulting to black:

```
13531 \newcommand*{\LWR@currenttextcolor}{black}
```

\LWR@tempcolor The color converted to HTML colorspace.
\LWR@tempcolortwo
\LWR@tempcolorthree
```
13532 \newcommand*{\LWR@tempcolor}{}
13533 \newcommand*{\LWR@tempcolortwo}{}
13534 \newcommand*{\LWR@tempcolorthree}{}
```

\LWR@findcurrenttextcolor Sets \LWR@tempcolor to the current color.

```
13535 \newcommand*{\LWR@findcurrenttextcolor}{%
13536     \renewcommand{\LWR@tempcolor}{000000}%
13537 }
```

\LWR@textcurrentcolor {⟨*text*⟩} Like \textcolor but uses the current \color instead.

```
13538 \NewDocumentCommand{\LWR@textcurrentcolor}{m}{%
13539     \renewcommand*{\LWR@currenttextcolor}{black}%
13540     #1%
13541 }
```

```
13542 \end{warpHTML}
```

for PRINT output: 13543 \begin{warpprint}

\LWR@textcurrentfont {⟨*text*⟩}

Prints the text with the current font choices.

```
13544 \newcommand*{\LWR@textcurrentfont}[1]{#1}
```

LWR@blocktextcurrentfont (*env.*) Prints the contents with the current font choices.

```
13545 \newenvironment*{LWR@blocktextcurrentfont}{}{}
```

\FilenameNullify {⟨*macros to nullify*⟩}

```
13546 \newcommand*{\FilenameNullify}[1]{}
```

```
13547 \end{warpprint}
```

96   Skips, spaces, font sizes

13548 \begin{warpHTML}

\, and \thinspace may be redefined by other packages, so are redefined \AtBeginDocument here.

Direct-formatting space commands become HTML entities:

```
13549 \AtBeginDocument{%
13550 \renewrobustcmd*{\,}{\HTMLunicode{202f}}% HTML thin non-breakable space, not using LWR@formatted
13551 \newrobustcmd*{\LWR@HTML@thinspace}{\HTMLunicode{202f}}% HTML thin non-breakable space
13552 \LWR@formatted{thinspace}
13553 \newrobustcmd*{\LWR@HTML@negthinspace}{\HTMLunicode{202f}}   % HTML thin non-breakable space
13554 \LWR@formatted{negthinspace}
13555 \renewrobustcmd*{~}{\HTMLentity{nbsp}}% cannot use \LWR@formatted
13556 \newrobustcmd*{\LWR@HTML@textellipsis}{\HTMLunicode{2026}}
13557 \LWR@formatted{textellipsis}
13558 \newrobustcmd*{\LWR@HTML@vdots}{\HTMLunicode{22EE}}
13559 \LWR@formatted{vdots}
13560 }
```

Direct-formatting font sizes are remembered for future use:

```
13561 \newcommand*{\LWR@font@size}{normalsize}
13562
13563 \newrobustcmd*{\LWR@HTML@normalsize}{\renewcommand*{\LWR@font@size}{normalsize}}
13564 \LWR@formatted{normalsize}
13565
13566 \newrobustcmd*{\LWR@HTML@small}{\renewcommand*{\LWR@font@size}{small}}
13567 \LWR@formatted{small}
13568
13569 \newrobustcmd*{\LWR@HTML@footnotesize}{\renewcommand*{\LWR@font@size}{footnotesize}}
13570 \LWR@formatted{footnotesize}
13571
13572 \newrobustcmd*{\LWR@HTML@scriptsize}{\renewcommand*{\LWR@font@size}{scriptsize}}
13573 \LWR@formatted{scriptsize}
13574
13575 \newrobustcmd*{\LWR@HTML@tiny}{\renewcommand*{\LWR@font@size}{tiny}}
13576 \LWR@formatted{tiny}
13577
13578 \newrobustcmd*{\LWR@HTML@large}{\renewcommand*{\LWR@font@size}{large}}
13579 \LWR@formatted{large}
13580
13581 \newrobustcmd*{\LWR@HTML@Large}{\renewcommand*{\LWR@font@size}{Large}}
13582 \LWR@formatted{Large}
13583
13584 \newrobustcmd*{\LWR@HTML@LARGE}{\renewcommand*{\LWR@font@size}{LARGE}}
13585 \LWR@formatted{LARGE}
13586
```

```
13587 \newrobustcmd*{\LWR@HTML@huge}{\renewcommand*{\LWR@font@size}{huge}}
13588 \LWR@formatted{huge}
13589
13590 \newrobustcmd*{\LWR@HTML@Huge}{\renewcommand*{\LWR@font@size}{Huge}}
13591 \LWR@formatted{Huge}


13592 \DeclareDocumentCommand{\onecolumn}{}{}
13593
13594 \DeclareDocumentCommand{\twocolumn}{O{}}{
13595
13596 #1
13597
13598 }
```

\hfill

```
13599 \newcommand*{\LWR@HTML@hfill}{\qquad}
13600 \LWR@formatted{hfill}
```

\hrulefill

```
13601 \newcommand*{\LWR@HTML@hrulefill}{%
13602     \ifbool{LWR@doingapar}%
13603         {\rule{1in}{1pt}}%
13604         {%
13605             \LWR@findcurrenttextcolor%
13606             \ifdefstring{\LWR@tempcolor}{000000}%
13607             {%
13608                 \begin{BlockClass}{hrule}%
13609                 \end{BlockClass}%
13610             }%
13611             {%
13612                 \begin{BlockClass}[%
13613                     border-top: 1px solid \LWR@origpound\LWR@tempcolor % space
13614                 ]{hrule}%
13615                 \end{BlockClass}%
13616             }%
13617         }%
13618 }%
13619 \LWR@formatted{hrulefill}
```

\dotfill

```
13620 \newcommand*{\LWR@HTML@dotfill}{\dots}
13621 \LWR@formatted{dotfill}
```

\newpage

```
13622 \renewcommand*{\newpage}{
13623
13624 }
```

\newline Uses the HTML <br /> element.

```
13625 \newrobustcmd*{\LWR@newlinebr}{\unskip\LWR@htmltag{br /}\LWR@orignewline}%
13626 \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.

```
13627 \LetLtxMacro\LWR@origendofline\\
13628 \NewDocumentCommand{\LWR@endofline}{s O{0pt}}
13629 {%
13630 \newline%

13631 \setlength{\LWR@templengthone}{#2}%
13632 \ifdimgreater{\LWR@templengthone}{0pt}{\newline}{}%
13633 }
```

\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 small horizontal space between minipages, it is desirable to maintain this space if possible. lwarp tries to do this by remembering that a minipage has been seen, in which case paragraph tags are suppressed around \hspace, \enskip, \quad, and \qquad until the end of the paragraph, when the closing p tag is created.

\hspace
\enskip
\quad
\qquad

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.

```
13634 \newcommand*{\LWR@minipagestartpars}{%
13635     \ifbool{LWR@minipagethispar}{\LWR@startpars}{}%
13636 }
```

\LWR@minipagestoppars   Placed just after \hspace, \quad, or \qquad's HTML output.

```
13637 \newcommand*{\LWR@minipagestoppars}{%
13638     \ifbool{LWR@minipagethispar}{\LWR@stoppars}{}%
13639 }
```

\quad   Handles special minipage & horizontal space interactions. Uses 2003 EM SPACE to pass validation.

```
13640 \newrobustcmd*{\LWR@HTML@quad}{%
13641     \LWR@minipagestoppars%
13642     \HTMLunicode{2003}%
13643     \LWR@minipagestartpars%
13644 }
13645 \LWR@formatted{quad}
```

\qquad Handles special minipage & horizontal space interactions.

```
13646 \newrobustcmd*{\LWR@HTML@qquad}{\quad\quad}
13647 \LWR@formatted{qquad}
```

\enskip Handles special minipage & horizontal space interactions.

```
13648 \newrobustcmd*{\LWR@HTML@enskip}{%
13649     \LWR@minipagestoppars%
13650     \HTMLunicode{2002}%
13651     \LWR@minipagestartpars%
13652 }
13653 \LWR@formatted{enskip}
```

\LWR@tempwidth (*Len*)   Used to compute span width, height, raise for \hspace and \rule:

\LWR@tempheight (*Len*)
```
13654 \newlength{\LWR@tempwidth}
```
\LWR@tempraise (*Len*)
```
13655 \newlength{\LWR@tempheight}
13656 \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.

```
13657 \NewDocumentCommand{\LWR@HTML@hspace}{s m}{%
13658 \setlength{\LWR@tempwidth}{#2}%
```

If \fill, change to \qquad:

```
13659 \ifnum\gluestretchorder\LWR@tempwidth>0%
13660 \setlength{\LWR@tempwidth}{2em}%
13661 \fi%
```

Only if the width is greater than zero:

```
13662 \ifdimcomp{\LWR@tempwidth}{>}{0pt}{%
```

If had a minipage this paragraph, try to inline the white space without generating paragraph tags:

```
13663     \LWR@minipagestoppars%
```

Support the HTML thin wrappable space:

```
13664     \ifdimcomp{\LWR@tempwidth}{=}{.16667em}%
13665     {%
13666         \HTMLunicode{2009}% thin breakable space
13667     }%
```

Print the span with the converted width. Not rounded.

```
13668     {%
```

```
13669          \LWR@htmltagc{%
13670          span style=\textquotedbl{}width:\LWR@printlength{\LWR@tempwidth}; % extra space
13671                  display:inline-block\textquotedbl%
13672          }%
```

If formatting for a word processor, approximate with a number of \quads, in case a span of a given width is not supported:

```
13673          \ifbool{FormatWP}{%
13674              \setlength{\LWR@templengthone}{\LWR@tempwidth}%
13675              \whiledo{\lengthtest{\LWR@templengthone>1em}}{%
13676                  \quad%
13677                  \addtolength{\LWR@templengthone}{-1em}%
13678              }%
13679          }%
```

If NOT formatting for a word processor, include an empty comment to avoid an empty span:

```
13680          {\LWR@htmlcomment{}}%
```

Close the span:

```
13681          \LWR@htmltagc{/span}%
13682      }%
```

If had a minipage this paragraph, try to inline the white space without generating paragraph tags:

```
13683      \LWR@minipagestartpars%
13684 }{}% width greater than 0
13685 }%
13686 \LWR@formatted{hspace}
```

\LWR@vspace * {⟨*length*⟩} Nullified vspace.

```
13687 \NewDocumentCommand{\LWR@HTML@vspace}{s m}{}
13688
13689 \LWR@formatted{vspace}
```

\linebreak [⟨*num*⟩]   Inserts an HTML br tag.

```
13690 \renewcommand*{\linebreak}[1][]{\newline}
```

\nolinebreak [⟨*num*⟩]

```
13691 \renewcommand*{\nolinebreak}[1][]{}
```

\pagebreak [⟨*num*⟩]   Starts a new paragraph.

```
13692 \renewcommand*{\pagebreak}[1][]{
13693
13694 }
```

\nopagebreak [⟨*num*⟩]

```
13695 \renewcommand*{\nopagebreak}[1][]{}
```

\enlargethispage * {⟨*len*⟩}

```
13696 \RenewDocumentCommand{\enlargethispage}{s m}{}
```

\clearpage
\cleardoublepage

```
13697 \renewcommand*{\clearpage}{}
13698 \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.

```
13699 \newcommand*{\LWR@HTML@rule}[3][]{%
```

The width is copied into a temporary LATEX length, from which comparisons and conversions may be made:

```
13700 \setlength{\LWR@tempwidth}{#2}%
```

If it's zero-width then skip the entire rule:

```
13701 \ifthenelse{\lengthtest{\LWR@tempwidth=0pt}}%
13702 {}% zero- width
13703 {% non-zero width
```

If it's non-zero width, set a minimal thickness so that it more reliably shows in the browser:

```
13704     \ifthenelse{%
13705         \lengthtest{\LWR@tempwidth>0pt}\AND%
13706         \lengthtest{\LWR@tempwidth<1pt}%
13707     }%
13708         {\setlength{\LWR@tempwidth}{1pt}}%
13709         {}%
```

Likewise with height:

```
13710     \setlength{\LWR@tempheight}{#3}%
13711     \ifthenelse{%
13712         \lengthtest{\LWR@tempheight>0pt}\AND%
13713         \lengthtest{\LWR@tempheight<1pt}%
13714     }%
13715         {\setlength{\LWR@tempheight}{1pt}}%
13716         {}%
```

If had a minipage this paragraph, try to inline the rule without generating paragraph tags:

```
13717     \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.

```
13718    \LWR@findcurrenttextcolor%
13719    \LWR@htmltagc{%
13720    span\LWR@indentHTML%
13721    style=\textquotedbl%
```

The HTML background color is used to draw the filled rule according to the LaTeX foreground color set by \textcolor.

```
13722    \ifbool{FormatWP}{}{background:\LWR@currenttextcolor ; }%
```

The width and height are printed, converted to PT:

```
13723    width:\LWR@printlength{\LWR@tempwidth} ; %
13724    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.

```
13725    \ifblank{#1}%
13726    {}%
13727    {%
13728        \setlength{\LWR@tempraise}{0pt-#1}%
13729        \setlength{\LWR@tempraise}{\LWR@tempraise*2}%
13730        \LWR@indentHTML%
13731        -ms-transform: translate(0pt,\LWR@printlength{\LWR@tempraise}); %
13732        \LWR@indentHTML%
13733      -webkit-transform: translate(0pt,\LWR@printlength{\LWR@tempraise}); %
13734        \LWR@indentHTML%
13735        transform: translate(0pt,\LWR@printlength{\LWR@tempraise}); %
13736        \LWR@indentHTML%
13737    }%
```

Display inline-block to place the span inline with the text:

```
13738    display:inline-block;\textquotedbl\LWR@orignewline%
13739    }%
```

If formatting for a word processor, approximate with a number of underscores, in case a span of a given width is not supported:

```
13740        \ifbool{FormatWP}{%
13741            \setlength{\LWR@templengthone}{\LWR@tempwidth}%
13742            \whiledo{\lengthtest{\LWR@templengthone>1em}}{%
13743                \_{}%
13744                \addtolength{\LWR@templengthone}{-1em}%
13745            }%
13746        }%
```

If NOT formatting for a word processor, add a comment to avoid an empty <span>:

13747   {\LWR@htmlcomment{}}%

Close the span:

13748   \LWR@htmltagc{/span}%

If had a minipage this paragraph, try to inline the white space without generating
paragraph tags:

13749   \LWR@minipagestartpars%
13750 }% non-zero width
13751 }
13752
13753 \LWR@formatted{rule}

13754 \end{warpHTML}

97 \phantomsection

13755 \begin{warpHTML}

\LWR@phantomsection Emulate the hyperref \phantomsection command, often used to insert the bibli-
ography into the table of contents. Ignores \ForceHTMLTOC.

13756 \newrobustcmd*{\LWR@phantomsection}{%
13757   \begingroup%
13758   \boolfalse{LWR@forcinghtmltoc}%
13759   \section*{}%
13760   \endgroup%
13761 }

13762 \end{warpHTML}

98 \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.

13763 \begin{warpall}
13764 \newbool{LWR@warnXe}
13765 \boolfalse{LWR@warnXe}
13766
13767 \newrobustcmd*{\Xe}
13768   {%

```
13769        X\hspace{-.1667em}\raisebox{-.5ex}{E}%
13770        \global\booltrue{LWR@warnXe}%
13771    }
13772
13773 \AtBeginDocument{
13774    \IfPackageLoadedTF{graphics}{
13775        \IfPackageLoadedTF{metalogo}{}{
13776            \renewrobustcmd*{\Xe}
13777                {X\hspace{-.1667em}\raisebox{-.5ex}{\reflectbox{E}}}}
13778        }
13779    }{}
13780 }
13781
13782 \AtEndDocument{
13783    \ifbool{LWR@warnXe}{
13784        \PackageNoteNoLine{lwarp}{Load graphicx or graphics
13785            for improved XeTeX logo}
13786    }{}
13787 }
13788
13789 \providerobustcmd*{\XeTeX}{\mbox{\Xe\hspace{-.125em}\TeX}}
13790 \providerobustcmd*{\XeLaTeX}{\mbox{\Xe\hspace{-.125em}\LaTeX}}
13791 \providerobustcmd*{\AmS}{%
13792    \leavevmode\hbox{$\mathcal A\kern-.2em\lower.376ex%
13793    \hbox{$\mathcal M$}\kern-.2em\mathcal S$}%
13794 }
13795 \newrobustcmd*{\LyX}{\textsf{LyX}}
13796 \providerobustcmd*{\LuaTeX}{\mbox{Lua\TeX}}
13797 \providerobustcmd*{\LuaLaTeX}{\mbox{Lua\LaTeX}}
13798 \providerobustcmd*{\BibTeX}{\mbox{B\textsc{ib}\TeX}}
13799 \providerobustcmd*{\MakeIndex}{\mbox{\textit{MakeIndex}}}
13800 \providerobustcmd*{\ConTeXt}{\mbox{Con\TeX{}t}}
13801 \providerobustcmd*{\MiKTeX}{\mbox{MiK\TeX}}
13802 \end{warpall}
```

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.

```
13804 \newrobustcmd*{\LWR@HTML@TeX}
13805 {%
13806    \InlineClass{latexlogofont}%
13807    {%
13808        \InlineClass{latexlogo}%
13809        {%
13810            T%
13811            \InlineClass{latexlogosub}{e}%
13812            X%
13813        }%
13814    }%
13815 }
```

```
13816 \AtBeginDocument{\LWR@formatted{TeX}}% may have been patched by metalogo
```

\LaTeX LATEX, LATEX 2ε
\LaTeXe

```
13817 \newrobustcmd*{\LWR@HTML@LaTeX}
13818 {%
13819     \InlineClass{latexlogofont}%
13820     {%
13821         \InlineClass{latexlogo}%
13822         {%
13823             L%
13824             \InlineClass{latexlogosup}{a}%
13825             T%
13826             \InlineClass{latexlogosub}{e}%
13827             X%
13828         }%
13829     }%
13830 }
13831
13832 \AtBeginDocument{\LWR@formatted{LaTeX}}% may have been patched by metalogo
13833
13834
13835 \newrobustcmd*{\LWR@HTML@LaTeXe}
13836 {%
13837     \LaTeX%
13838     \InlineClass{latexlogofont}{%
13839         \InlineClass{latexlogotwoe}{%
13840             2%
13841             \InlineClass{latexlogotwoesub}{\HTMLunicode{03B5}}%
13842         }%
13843     }%
13844 }
13845 \AtBeginDocument{\LWR@formatted{LaTeXe}}% may have been patched by metalogo
```

\LuaTeX LuaTEX, LuaLATEX
\LuaLaTeX

```
13846 \newrobustcmd*{\LWR@HTML@LuaTeX}{\InlineClass{latexlogofont}{Lua}\TeX}
13847 \AtBeginDocument{\LWR@formatted{LuaTeX}}% may have been patched by metalogo
13848
13849 \newrobustcmd*{\LWR@HTML@LuaLaTeX}{\InlineClass{latexlogofont}{Lua}\LaTeX}
13850 \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.

```
13851 \newrobustcmd*{\LWR@HTML@Xe}
13852     {%
13853         X%
13854         \InlineClass{xelatexlogosub}{\HTMLunicode{18e}}%
13855     }
13856 \AtBeginDocument{\LWR@formatted{Xe}}% may have been patched by metalogo
13857
13858 \newrobustcmd*{\LWR@HTML@XeTeX}{\InlineClass{xelatexlogo}{\Xe}\TeX}
```

```
13859 \AtBeginDocument{\LWR@formatted{XeTeX}}% may have been patched by metalogo
13860
13861 \newrobustcmd*{\LWR@HTML@XeLaTeX}{\InlineClass{xelatexlogo}{\Xe}\LaTeX}
13862 \AtBeginDocument{\LWR@formatted{XeLaTeX}}% may have been patched by metalogo
```

\ConTeXt ConTeXt

```
13863 \newrobustcmd*{\LWR@HTML@ConTeXt}{%
13864     \InlineClass{latexlogofont}{Con}\TeX{}%
13865     \InlineClass{latexlogofont}{t}%
13866 }
13867 \LWR@formatted{ConTeXt}
```

\BibTeX BIBTEX, *MakeIndex*
\MakeIndex

```
13868 \newrobustcmd*{\LWR@HTML@BibTeX}
13869     {\InlineClass{latexlogofont}{B\textsc{ib}}\TeX}
13870 \LWR@formatted{BibTeX}
13871
13872 \newrobustcmd*{\LWR@HTML@MakeIndex}
13873     {\InlineClass{latexlogofont}{\textit{MakeIndex}}}
13874 \LWR@formatted{MakeIndex}
```

\AmS $\mathcal{A}_{\mathcal{M}}\mathcal{S}$

amslogo is a css class used for the $\mathcal{A}_{\mathcal{M}}\mathcal{S}$ logo.

```
13875 \AtBeginDocument{%
13876 \newrobustcmd*{\LWR@HTML@AmS}
13877 {%
13878     \InlineClass{amslogo}{%
13879         \textit{%
13880             A%
13881             \InlineClass{latexlogosub}{M}%
13882             S%
13883         }%
13884     }%
13885 }%
13886 \LWR@formatted{AmS}
13887 }
```

\MiKTeX MiKTEX

```
13888 \newrobustcmd*{\LWR@HTML@MiKTeX}{\InlineClass{latexlogofont}{MiK}\TeX}
13889 \LWR@formatted{MiKTeX}
```

\LyX LyX

lyxlogo is a css class used for the LyX logo.

```
13890 \newrobustcmd*{\LWR@HTML@LyX}{\InlineClass{lyxlogo}{LyX}}
13891 \LWR@formatted{LyX}
```

```
13892 \end{warpHTML}
```

99 Starting and stopping lwarp

13893 `\begin{warpHTML}`

`\LWR@LwarpStart` Automatically sets up the HTML-related actions for the start and end of the docu-
`\LWR@LwarpEnd` ment.

```
13894 \AfterEndPreamble{\LWR@LwarpStart}
13895 \AtEndDocument{\LWR@LwarpEnd}
13896 \DeclareHookRule{enddocument}{lwarp}{after}{legacy}
```

13897 `\end{warpHTML}`

100 Loading array

array is required for lwarp's column parsing. It and its patches are now loaded.

13898 `\begin{warpHTML}`
13899 `\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.

```
13900 \edef\LWR@detect@centeringarraybackslash{\centering\arraybackslash}
13901 \edef\LWR@detect@raggedrightarraybackslash{\raggedright\arraybackslash}
13902 \edef\LWR@detect@raggedleftarraybackslash{\raggedleft\arraybackslash}
13903 \def\LWR@detect@itshape{\itshape}
13904 \def\LWR@detect@bfseries{\bfseries}
13905 \def\LWR@detect@bfit{\bfseries\itshape}
13906 \end{warpHTML}
```

101 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.

13907 `\begin{warpHTML}`
```
13908 \AtBeginDocument{
13909     \IfPackageLoadedTF{everyshi}{
13910         \RequirePackage{lwarp-everyshi}
13911     }{}
13912 }
13913 \end{warpHTML}
```

102 Loading textcomp patches

textcomp has now been integrated into the LATEX core, so its patches are loaded
now.

13914 `\begin{warpHTML}`
      13915 `\RequirePackage{lwarp-textcomp}`
      13916 `\end{warpHTML}`

103   Loading amsmath, amsthm patches, centernot

amsmath, amsthm, and centernot may have been preloaded, such as by newtx, so their patches are loaded now.

13917 `\begin{warpHTML}`
      13918 `\IfPackageLoadedTF{amsthm}{`
      13919   `\RequirePackage{lwarp-amsthm}`
      13920 `}{}`

      13921 `\IfPackageLoadedTF{amsmath}{`
      13922   `\RequirePackage{lwarp-amsmath}`
      13923 `}{}`

amsthm may load centernot, so centernot must be checked second.

      13924 `\IfPackageLoadedTF{centernot}{`
      13925   `\RequirePackage{lwarp-centernot}`
      13926 `}{}`
      13927 `\end{warpHTML}`

104   Loading KOMA-SCRIPT class patches

Load patches to koma-script.

13928 `\begin{warpHTML}`

      13929 `\IfClassLoadedTF{scrbook}{\RequirePackage{lwarp-patch-komascript}}{}`
      13930 `\IfClassLoadedTF{scrartcl}{\RequirePackage{lwarp-patch-komascript}}{}`
      13931 `\IfClassLoadedTF{scrreprt}{\RequirePackage{lwarp-patch-komascript}}{}`

      13932 `\end{warpHTML}`

105   Loading MEMOIR class patches

Load patches to memoir.

13933 `\begin{warpprint}`
      13934 `\IfClassLoadedTF{memoir}{\LWR@origRequirePackage{xcolor}}{}`
      13935 `\end{warpprint}`

13936 `\begin{warpHTML}`
      13937 `\IfClassLoadedTF{memoir}{\RequirePackage{lwarp-patch-memoir}}{}`
      13938 `\end{warpHTML}`

106   ut* class patches

Load patches to uj* and ut* classes, as well as ltj* classes.

13939 \begin{warpHTML}

13940 \newcommand*{\LWR@patchujtclasses}{

uj/t does not use \partname

13941   \def\@partnameformat{}

13942   \def\@partcntformat##1{%
13943   \prepartname%
13944   \csname the##1\endcsname%
13945   \postpartname%
13946   \quad%
13947   }
13948   \@ifundefined{chapter}{}{
13949   \def\@chapcntformat##1{%
13950   \prechaptername%
13951   \csname the##1\endcsname%
13952   \postchaptername%
13953   \quad%
13954   }
13955   }
13956   \renewcommand*{\LWR@printchaptername}{}

Use decimal points instead of centered dots:

13957   \renewcommand{\thepart}{\@Roman\c@part}
13958   \@ifundefined{chapter}{
13959   \renewcommand{\thesection}{\@arabic\c@section}
13960   }{
13961   \renewcommand{\thechapter}{\@arabic\c@chapter}
13962   \renewcommand{\thesection}{\thechapter.\@arabic\c@section}
13963   }
13964   \renewcommand{\thesubsection}{\thesection.\@arabic\c@subsection}
13965   \renewcommand{\thesubsubsection}{%
13966   \thesubsection.\@arabic\c@subsubsection}
13967   \renewcommand{\theparagraph}{%
13968   \thesubsubsection.\@arabic\c@paragraph}
13969   \renewcommand{\thesubparagraph}{%
13970   \theparagraph.\@arabic\c@subparagraph}
13971   \@ifundefined{chapter}{
13972   \renewcommand{\thefigure}{\@arabic\c@figure}
13973   \renewcommand{\thetable}{\@arabic\c@table}
13974   }{
13975   \renewcommand{\thefigure}{%
13976   \ifnum\c@chapter>\z@\thechapter.\fi\@arabic\c@figure}
13977   \renewcommand{\thetable}{%
13978   \ifnum\c@chapter>\z@\thechapter.\fi\@arabic\c@table}
13979   }
13980 }
13981
13982 \IfClassLoadedTF{ujarticle}{\LWR@patchujtclasses}{}
13983 \IfClassLoadedTF{ujbook}{\LWR@patchujtclasses}{}
13984 \IfClassLoadedTF{ujreport}{\LWR@patchujtclasses}{}

```
13985 \IfClassLoadedTF{utarticle}{\LWR@patchujtclasses}{}
13986 \IfClassLoadedTF{utbook}{\LWR@patchujtclasses}{}
13987 \IfClassLoadedTF{utreport}{\LWR@patchujtclasses}{}
13988 \IfClassLoadedTF{ltjarticle}{\LWR@patchujtclasses}{}
13989 \IfClassLoadedTF{ltjbook}{\LWR@patchujtclasses}{}
13990 \IfClassLoadedTF{ltjreport}{\LWR@patchujtclasses}{}
13991 \IfClassLoadedTF{ltjsarticle}{\LWR@patchujtclasses}{}
13992 \IfClassLoadedTF{ltjsbook}{\LWR@patchujtclasses}{}
13993 \IfClassLoadedTF{ltjsreport}{\LWR@patchujtclasses}{}
13994 \IfClassLoadedTF{ltjskiyou}{\LWR@patchujtclasses}{}
13995 \IfClassLoadedTF{ltjspf}{\LWR@patchujtclasses}{}
13996 \IfClassLoadedTF{ltjtarticle}{\LWR@patchujtclasses}{}
13997 \IfClassLoadedTF{ltjtbook}{\LWR@patchujtclasses}{}
13998 \IfClassLoadedTF{ltjtreport}{\LWR@patchujtclasses}{}
```

```
13999 \end{warpHTML}
```

107   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.

14000 \begin{warpHTML}

\AtBeginDocument in case the user set FileSectionNames in the preamble.

```
14001 \AtBeginDocument{
14002     \IfPackageLoadedTF{ctexpatch}{%
14003         \def\@partcntformat#1{%
14004             \LWR@isolate{\CTEX@partname}~%
14005             \CTEX@part@aftername%
14006         }%
14007
14008         \def\@partnameformat{}
14009
14010         \def\@chapcntformat#1{%
14011             \LWR@isolate{\CTEX@chaptername}~%
14012             \CTEX@chapter@aftername%
14013         }%
14014
14015         \renewcommand*{\LWR@printchaptername}{}
14016     }{}
14017 }
```

```
14018 \end{warpHTML}
```

108   kotexutf patches

Patch for kotexutf, which is loaded before lwarp.

kotexutf's \@setref was conflicting with lwarp's cross references.

14019 \begin{warpHTML}

If kotexutf's version of \@setref is detected, it is reverted to the original.

```
14020 \AtBeginDocument{
14021 \IfPackageLoadedTF{kotexutf}{%
14022     \def\LWR@kotexutf@setref#1#2#3{%
14023         \@setref@dhucs@orig{#1}{#2}{#3}%
14024       \ifx#1\relax\else
14025         \bgroup
14026         \dhucs@make@cjkchar@null
14027         \edef\@temp{\expandafter#2#1}\global\josatoks\expandafter{\@temp}%
14028         \egroup
14029       \fi%
14030     }%
14031
14032     \ifdefequal{\@setref}{\LWR@kotexutf@setref}{
14033         \let\@setref\@setref@dhucs@orig
14034     }{}
14035 }{}
14036 }
```

```
14037 \end{warpHTML}
```

109 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.

```
14038 \begin{warpHTML}
14039 \AtBeginDocument{
14040
14041 \IfPackageLoadedTF{polyglossia}{
14042     \PackageNoteNoLine{lwarp}
14043     {%
14044         Polyglossia has been loaded.  Lwarp also uses cleveref.\MessageBreak
14045         See the cleveref documentation regarding\MessageBreak
14046         polyglossia support.  Some languages are not supported.\MessageBreak
```

```
14047          --- \MessageBreak
14048          If the error\MessageBreak
14049          \space\space Undefined control sequence ...
14050          \protect\__hook begindocument\MessageBreak
14051          occurs here, use the polyglossia macro:\MessageBreak
14052          \space\space\protect\setmainlanguage\protect{...\protect}
14053      }
14054 }{
14055      \IfPackageLoadedTF{babel}{
14056          \PackageNoteNoLine{lwarp}
14057          {%
14058              Babel has been loaded.  Lwarp also uses cleveref.\MessageBreak
14059              See the cleveref documentation regarding\MessageBreak
14060              babel support.  Some languages are not supported%
14061          }
14062      }{}
14063 }
14064
14065 }
14066 \end{warpHTML}
```

110   MATHJAX warnings

\LWR@mathjaxwarn {⟨*packagename*⟩} {⟨*More text.*⟩}

      Issue a warning that MATHJAX is emulated. To be done \AtBeginDocument.

```
14067 \newcommand*{\LWR@mathjaxwarn}[2]{%
14068      \IfPackageLoadedTF{lwarp-#1}{%
14069          \ifblank{#2}{%
14070              \PackageWarningNoLine{lwarp}
14071                  {%
14072                      Lwarp provides emulation for MathJax when used\MessageBreak
14073                          with the #1 package%
14074                  }
14075          }{%
14076              \PackageWarningNoLine{lwarp}
14077                  {%
14078                      Lwarp provides emulation for MathJax when used\MessageBreak
14079                          with the #1 package.\MessageBreak
14080                          #2%
14081                  }
14082          }%
14083      }{}%
14084 }
14085
14086 % \begin{macro}{\LWR@nomathjaxwarn} \marg{packagename} \marg{More text.}
14087 %
14088 % Issue a warning that \MathJax\ is not supported.
14089 % To be done \cs{AtBeginDocument}.
14090 %
14091 % \changes{v0.894}{2020/12/22}{Warn if using packages not supported by \MathJax.}
14092 % \changes{v0.895}{2021/01/08}{Improved \MathJax\ warning.}
14093 %    \begin{macrocode}
14094 \newcommand*{\LWR@nomathjaxwarn}[2]{%
14095      \IfPackageLoadedTF{lwarp-#1}{%
14096          \ifblank{#2}{%
```

```
14097              \PackageWarningNoLine{lwarp}
14098                 {%
14099                   Lwarp does not provide MathJax support for #1.\MessageBreak
14100                     Use SVG math by removing the Lwarp mathjax option%
14101                 }
14102          }{%
14103              \PackageWarningNoLine{lwarp}
14104                 {%
14105                   Lwarp does not provide MathJax support for #1.\MessageBreak
14106                     #2%
14107                 }
14108          }%
14109      }{}%
14110 }
```

\LWR@forceSVGmessage {⟨*packagename*⟩}

```
14111 \newcommand*{\LWR@forceSVGmessage}[1]{%
14112   SVG math output may be enabled for select math\MessageBreak
14113   expressions to preserve #1 visual\MessageBreak
14114   features for those particular expressions.\MessageBreak
14115   Before the chosen inline math, use \protect\inlinemathother\MessageBreak
14116   to begin using SVG math, and \protect\inlinemathnormal\MessageBreak
14117   afterward to resume using MathJax math.\MessageBreak
14118   Before display math, use \protect\displaymathother\MessageBreak
14119   to begin using SVG math, and use \protect\displaymathnormal\MessageBreak
14120   after to resume using MathJax for the following math.\MessageBreak
14121   Or, use SVG math for all expressions by removing\MessageBreak
14122   the mathjax option for the lwarp package%
14123 }
```

If MᴀᴛʜJᴀx is being used, issue a warning for certain packages.

```
14124 \AtBeginDocument{
14125   \ifbool{mathjax}{
14126       \LWR@nomathjaxwarn{aligned-overset}{}
14127       \LWR@nomathjaxwarn{amscdx}{\LWR@forceSVGmessage{amscdx}}
14128       \LWR@mathjaxwarn{arydshln}
14129          {In a math array, do not use the optional argument\MessageBreak
14130           for \protect\cdashline.\space\space
14131           Furthermore, \protect\cline\space is not\MessageBreak
14132           supported by MathJax}
14133       \LWR@nomathjaxwarn{autoaligne}{}
14134       \LWR@mathjaxwarn{autonum}
14135          {MathJax does not support equation+.\MessageBreak
14136           You may use the warpprint and warpHTML\MessageBreak
14137           environments to isolate the package load\MessageBreak
14138           and the equation+ environments}
14139       \LWR@mathjaxwarn{bigdelim}
14140          {Delimiters appear only of the first line}
14141       \LWR@nomathjaxwarn{boldtensors}{}
14142       \LWR@mathjaxwarn{booktabs}
14143          {\protect\cmidrule\space is not displayed}
14144       \LWR@mathjaxwarn{breqn}
14145          {Each environment becomes an SVG image}
14146       \LWR@mathjaxwarn{colortbl}
14147          {Colors are ignored in MathJax.\MessageBreak
14148           (Text mode tabular does support colortbl.)\MessageBreak
14149           \LWR@forceSVGmessage{colortbl}}
```

```
14150        \LWR@mathjaxwarn{delarray}{\LWR@forceSVGmessage{delarray}}
14151        \LWR@nomathjaxwarn{gauss}{\LWR@forceSVGmessage{gauss}}
14152        \LWR@mathjaxwarn{hhline}
14153            {A simple \protect\hline\space is used}
14154        \LWR@mathjaxwarn{isomath}
14155        {Some of the symbol font macros such as \protect\mathsfbfit\MessageBreak
14156            do not use a sans font because MathJax does not yet\MessageBreak
14157              have sans Greek.  Tensors may look like vectors%
14158            }
14159        \LWR@nomathjaxwarn{jkmath}{\LWR@forceSVGmessage{jkmath}}
14160        \LWR@mathjaxwarn{libertinust1math}
14161        {Some of the symbol font macros such as \protect\mathsfbfit\MessageBreak
14162            do not use a sans font because MathJax does not yet\MessageBreak
14163              have sans Greek.  Tensors may look like vectors%
14164            }
14165        \LWR@mathjaxwarn{mathtools}
14166            {See the Lwarp manual regarding the disallowspaces\MessageBreak
14167            and showonlyrefs options, the alignat environment,\MessageBreak
14168            and \protect\DeclarePairedDelimiter\space and related%
14169            }
14170        \LWR@mathjaxwarn{mathspec}
14171            {Double quotes are removed, even inside \protect\text}
14172        \LWR@mathjaxwarn{multirow}
14173            {Multirow works as expected in text mode, but\MessageBreak
14174            limited emulation is provided for MathJax math.\MessageBreak
14175          \protect\multirow\space ignores all arguments except\MessageBreak
14176            the text}
14177        \LWR@mathjaxwarn{nicematrix}
14178            {Keys/values are ignored in MathJax.\MessageBreak
14179            \protect\Cdots, etc. do not span multiple cells.\MessageBreak
14180            AutoNiceMatrix, etc. are not supported for MathJax.\MessageBreak
14181        \protect\CodeBefore, \protect\Body, and \protect\CodeAfter\MessageBreak
14182            \space\space also are not supported for MathJax.\MessageBreak
14183            \LWR@forceSVGmessage{nicematrix}%
14184            }
14185        \LWR@nomathjaxwarn{pb-diagram}{\LWR@forceSVGmessage{pb-diagram}}
14186 %      \LWR@mathjaxwarn{physics}
14187 % %          {The third-party extension is not used.\MessageBreak
14188 %          {The MathJax v3 extension is used.\MessageBreak
14189 %          See the Lwarp manual for details}
14190        \LWR@mathjaxwarn{siunitx}
14191        {Place \protect\sisetup\space before \protect\begin{document}.\MessageBreak
14192            Many optional arguments are ignored}
14193        \LWR@nomathjaxwarn{tensind}{}
14194        \LWR@mathjaxwarn{unicode-math}
14195            {Do not use embedded Unicode characters.\MessageBreak
14196            (Not all characters are encoded correctly.)\MessageBreak
14197            Some symbol fonts are not supported by MathJax,\MessageBreak
14198            and are only approximated.\MessageBreak
14199        Greek macros such as \protect\alpha\space respond to the math-style\MessageBreak
14200            option.  Latin symbols does not, per MathJax\MessageBreak
14201          limitations, unless placed inside \protect\symbit\space or similar}
14202        \LWR@nomathjaxwarn{unitsdef}{}
14203        \LWR@mathjaxwarn{witharrows}
14204            {Arrows can only point to the next line.\MessageBreak
14205            Text is only placed on a single line}
14206        \LWR@nomathjaxwarn{xy}
14207        {In text, xy works as-is.  SVG images will be generated.\MessageBreak
14208            \LWR@forceSVGmessage{xy}}
14209    }{}
```

14210 }

File 2   `lwarp-2in1.sty`

§ 111   Package   **2in1**

2in1 (*Pkg*)   2in1 is ignored.

for HTML output: 1 `\LWR@ProvidesPackageDrop{2in1}`

File 3   `lwarp-2up.sty`

§ 112   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`

§ 113   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`

§ 114   Package   **a4wide**

a4wide (*Pkg*)   a4wide is ignored.

for HTML output: 1 `\LWR@ProvidesPackageDrop{a4wide}[1994/08/30]`

File 6 **`lwarp-a5comb.sty`**

§ 115   Package   **a5comb**

a5comb (*Pkg*)   a5comb is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{a5comb}

File 7 **`lwarp-abstract.sty`**

§ 116   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 \IfClassLoadedTF{memoir}
18 {
19   \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`**

§ 117   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`**

§ 118 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 rede-
fined 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`

§ 119   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`

§ 120   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`**

§ 121   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`**

§ 122   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:

```
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 \renewcommand*\AC@acs[1]{%
66 %      \mbox{
67 \expandafter\AC@get\csname fn@#1\endcsname\@firstoftwo{#1}}
68 %  }
```

Fix for acronym labels in the captions of floats.

```
69 \renewcommand{\@starttoc}[1]{%
70     \LWR@htmlelementclass{nav}{#1}
71     \LetLtxMacro\@verridelabel\@gobble
72     \LWR@orig@starttoc{#1}
73     \LWR@htmlelementclassend{nav}{#1}
74 }
```

Modified for cleveref and lwarp:

```
75 \renewcommand*\AC@und@newl@bel[3]{%
76     \@ifundefined{#1@#3}%
77     {%
78         \global\expandafter\let\csname#2@#3\endcsname\@nnil
79         \global\expandafter\let\csname#2@#3@lwarp\endcsname\@nnil% lwarp
80         \global\expandafter\let\csname#2@#3@cref\endcsname\@nnil% lwarp
81     }%
82     {%
83         \global\expandafter\let\csname#1@#3\endcsname\relax
84         \global\expandafter\let\csname#1@#3@lwarp\endcsname\relax% lwarp
85         \global\expandafter\let\csname#1@#3@cref\endcsname\relax% lwarp
86     }%
87 }%
```

Improve paragraph handling:

```
88 \BeforeBeginEnvironment{acronym}{\LWR@stoppars}
89 \AfterEndEnvironment{acronym}{\LWR@startpars}
```

Create hyperlinks, even though hyperref is only emulated:

```
90 \AtBeginDocument{
91     \LetLtxMacro\AC@hyperlink\hyperlink
92     \LetLtxMacro\AC@hyperref\hyperref
93     \newcommand*\AC@raisedhypertarget[2]{%
94 %       \Hy@raisedlink{%
95             \hypertarget{#1}{}%
96 %         }%
97         #2}%
98     \LetLtxMacro\AC@hypertarget\AC@raisedhypertarget
99     \def\AC@phantomsection{%
100 %        \Hy@GlobalStepCount\Hy@linkcounter
101 %        \edef\@currentHref{section*.\the \Hy@linkcounter}%
102 %        \Hy@raisedlink{%
103 %          \hyper@anchorstart{\@currentHref}\hyper@anchorend
104 %        }%
105 %        \phantomsection%
106     }%
107 }
108
109 \appto\LWR@restoreorigformatting{%
110     \LetLtxMacro\AC@hyperlink\@secondoftwo%
111     \LetLtxMacro\AC@hyperref\LWR@nullify@hyperref%
112 }
```

File 14   `lwarp-adjmulticol.sty`

§ 123   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`

§ 124   Package   # addlines

(*Emulates or patches code by* WILL ROBERTSON.)

addlines (*Pkg*)   addlines is ignored.

for HTML output: 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`**

§ 125   Package   # afterpage

(*Emulates or patches code by* DAVID CARLISLE.)

afterpage (*Pkg*)   afterpage is emulated.

for HTML output: Discard all options for lwarp-afterpage:

1 \LWR@ProvidesPackageDrop{afterpage}[2014/10/28]

2 \newcommand{\afterpage}[1]{#1}

File 17   **`lwarp-algorithm2e.sty`**

§ 126   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.

for HTML output: 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`

§ 127   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 639.1.

```
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`

§ 128   Package   # alltt

(*Emulates or patches code by* Johannes Braams.)

alltt (*Pkg*)   alltt is patched for use by lwarp.

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`**

§ 129   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`.

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`**

§ 130   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 \LetLtxMacro\LWR@origmake@df@tag@@\make@df@tag@@
9 \LetLtxMacro\LWR@origmake@df@tag@@@\make@df@tag@@@
10
11 \renewcommand*{\make@df@tag@@}[1]{%
12     \LWR@remembertag{#1}%
13     \LWR@origmake@df@tag@@{#1}%
14 }
15
16 \renewcommand*{\make@df@tag@@@}[1]{%
17     \LWR@remembertag{#1}%
18     \LWR@origmake@df@tag@@@{#1}%
19 }
20
21 }% not mathjax
```

For nesting $\mathcal{A}_{\mathcal{M}}\mathcal{S}$ environments:

```
22 \newcounter{LWR@amsmathdepth}
23 \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.

```
24 \newcounter{LWR@maxfields@}
25
26 \xpatchcmd{\start@align}
```

```
27      {\maxfields@#3\relax}
28      {%
29          \maxfields@#3\relax%
30          \setcounter{LWR@maxfields@}{#3}%
31      }
32      {}
33      {\LWR@patcherror{amsmath}{start@align}}
```

\LWR@amsmathenv@@before   * {⟨*environment name*⟩}

* if the environment was starred.

Embeds the environment inside a lateximage.

```
34 \NewDocumentCommand{\LWR@amsmathenv@@before}{s m}{%
35      \IfBooleanTF{#1}{
36          \begin{BlockClass}{displaymath}
37      }{
38          \begin{BlockClass}{displaymathnumbered}
39      }
40      \LWR@newautoidanchor%
41      \booltrue{LWR@indisplaymathimage}%
42      \begin{lateximage}[\LWR@amsmathbodynumbered{#2}]*%
43      \LWR@applyxfakebold%
44 }
```

\LWR@amsmathenv@before   * {⟨*environment name*⟩}

* if the environment was starred.

Embeds the environment with MATHJAX or a lateximage.

```
45 \NewDocumentCommand{\LWR@amsmathenv@before}{s m}{%
46      \ifnumequal{\value{LWR@amsmathdepth}}{0}{%
47          \LWR@stoppars%
48      \ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
49          {
50              \LWR@syncmathjax
51              \boolfalse{LWR@amsmultline}
52              \ifstrequal{#2}{multline}{\booltrue{LWR@amsmultline}}{}
53              \ifstrequal{#2}{multline*}{\booltrue{LWR@amsmultline}}{}
```

⚠️   autonum's "+" environments are not supported by MATHJAX.

```
54              \LWR@beginhideamsmath
55          }
56          {
57              \IfBooleanTF{#1}{
58                  \LWR@amsmathenv@@before*{#2}
59              }{
60                  \LWR@amsmathenv@@before{#2}
61              }
62          }
63      }{}
64      \addtocounter{LWR@amsmathdepth}{1}
65 }
```

\LWR@amsmathenv@@after

Embeds the environment inside a lateximage.

```
66 \newcommand*{\LWR@amsmathenv@@after}{%
67      \end{lateximage}\end{BlockClass}\LWR@startpars%
```

68 }

\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.

```
69 \NewDocumentCommand{\LWR@amsmathenv@after}{s m}{%
70     \ifnumequal{\value{LWR@amsmathdepth}}{1}{%
71       \ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
72       {
73           \LWR@endhideamsmath
74           \boolfalse{LWR@amsmultline}
75           \LWR@addmathjax{#2}{\the\@envbody}
76       }
77       {\LWR@amsmathenv@@after}
```

Clear the single-use alt text:

```
78           \gdef\LWR@ThisAltText{}%
79     }{}
80     \addtocounter{LWR@amsmathdepth}{-1}
81 }
```

multline (*env.*)

```
82 \BeforeBeginEnvironment{multline}{\LWR@amsmathenv@before{multline}}
83
84 \AfterEndEnvironment{multline}{\LWR@amsmathenv@after{multline}}
```

multline∗ (*env.*)

```
85 \BeforeBeginEnvironment{multline*}{\LWR@amsmathenv@before*{multline*}}
86
87 \AfterEndEnvironment{multline*}{\LWR@amsmathenv@after*{multline*}}
88
```

gather (*env.*)

```
89 \BeforeBeginEnvironment{gather}{\LWR@amsmathenv@before{gather}}
90
91 \AfterEndEnvironment{gather}{\LWR@amsmathenv@after{gather}}
```

gather∗ (*env.*)

```
92 \BeforeBeginEnvironment{gather*}{\LWR@amsmathenv@before*{gather*}}
93
94 \AfterEndEnvironment{gather*}{\LWR@amsmathenv@after*{gather*}}
```

align (*env.*)

```
95 \BeforeBeginEnvironment{align}{\LWR@amsmathenv@before{align}}
96
97 \AfterEndEnvironment{align}{\LWR@amsmathenv@after{align}}
```

align∗ (*env.*)

```
 98 \BeforeBeginEnvironment{align*}{\LWR@amsmathenv@before*{align*}}
 99
100 \AfterEndEnvironment{align*}{\LWR@amsmathenv@after*{align*}}
```

flalign (*env.*)

```
101 \BeforeBeginEnvironment{flalign}{\LWR@amsmathenv@before{flalign}}
102
103 \AfterEndEnvironment{flalign}{\LWR@amsmathenv@after{flalign}}
```

flalign⋆ (*env.*)

```
104 \BeforeBeginEnvironment{flalign*}{\LWR@amsmathenv@before*{flalign*}}
105
106 \AfterEndEnvironment{flalign*}{\LWR@amsmathenv@after*{flalign*}}
```

alignat (*env.*)

```
107 \BeforeBeginEnvironment{alignat}{\LWR@amsmathenv@before{alignat}}
108
109 \AfterEndEnvironment{alignat}{\LWR@amsmathenv@after{alignat}}
```

alignat⋆ (*env.*)

```
110 \BeforeBeginEnvironment{alignat*}{\LWR@amsmathenv@before*{alignat*}}
111
112 \AfterEndEnvironment{alignat*}{\LWR@amsmathenv@after*{alignat*}}
```

```
113 \AtBeginEnvironment{subequations}{
114     \renewcommand*{\theMathJaxsubequations}{1}
115     \renewcommand*{\theMathJaxsection}{\theparentequation}
116     \renewcommand*{\theMathJaxequation}{\arabic{equation}}
117 }
```

For MATHJAX:

```
118 \begin{warpMathJax}
119 \CustomizeMathJax{\newcommand{\intertext}[1]{\text{#1}\notag \\}}
120 \CustomizeMathJax{\let\Hat\hat}
121 \CustomizeMathJax{\let\Check\check}
122 \CustomizeMathJax{\let\Tilde\tilde}
123 \CustomizeMathJax{\let\Acute\acute}
124 \CustomizeMathJax{\let\Grave\grave}
125 \CustomizeMathJax{\let\Dot\dot}
126 \CustomizeMathJax{\let\Ddot\ddot}
127 \CustomizeMathJax{\let\Breve\breve}
128 \CustomizeMathJax{\let\Bar\bar}
129 \CustomizeMathJax{\let\Vec\vec}
130 \end{warpMathJax}
```

File 22   `lwarp-amsthm.sty`

§ 131   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 \def\@xnthm#1#2{%
29   \csedef{LWR@thmstyle#2}{\LWR@newtheoremstyle}% lwarp
30   \let\@tempa\relax
31   \@xp\@ifdefinable\csname #2\endcsname{%
32     \global\@xp\let\csname end#2\endcsname\@endtheorem
33     \ifx *#1% unnumbered, need to get one more mandatory arg
34       \edef\@tempa##1{%
35         \gdef\@xp\@nx\csname#2\endcsname{%
36           \@nx\@thm{\@xp\@nx\csname th@\the\thm@style\endcsname}%
37             {}{##1}}}%
38     \else % numbered theorem, need to check for optional arg
39       \def\@tempa{\@oparg{\@ynthm{#2}}[]}%
40     \fi
41     \AtBeginEnvironment{#2}{%                                lwarp
42       \edef\LWR@thisthmstyle{\@nameuse{LWR@thmstyle#2}}%  lwarp
43     }%                                                      lwarp
44   }%
45   \@tempa%
46 }
```

Patched to enclose with css:

```
47 \newcommand{\LWR@haveamsthmname}{
48     \renewcommand{\thmname}[1]{%
49         \InlineClass{amsthmname\LWR@thisthmstyle}{##1}%
50     }
51 }
52
53 \newcommand{\LWR@haveamsthmnumber}{
54     \renewcommand{\thmnumber}[1]{%
55         \InlineClass{amsthmnumber\LWR@thisthmstyle}{##1}%
56     }
57 }
58
59 \newcommand{\LWR@haveamsthmnote}{
60     \renewcommand{\thmnote}[1]{%
61         \InlineClass{amsthmnote\LWR@thisthmstyle}{##1}%
62     }
63 }
64
65 \LWR@haveamsthmname
66 \LWR@haveamsthmnumber
67 \LWR@haveamsthmnote
```

Patched for css:

```
68 \def\@begintheorem#1#2[#3]{%


69     \GetTitleString{#3}%                                    lwarp
70     \let\@currentlabelname\GetTitleStringResult%            lwarp
71     \item[%


72     \LWR@newautopagelabel{page}\LWR@orignewline%


73 %   \deferred@thm@head{
74 %     \the\thm@headfont \thm@indent
```

```
75      \@ifempty{#1}{\let\thmname\@gobble}{\LWR@haveamsthmname}%          lwarp
76      \@ifempty{#2}{\let\thmnumber\@gobble}{\LWR@haveamsthmnumber}%      lwarp
77      \@ifempty{#3}{\let\thmnote\@gobble}{\LWR@haveamsthmnote}%          lwarp
78      \thm@swap\swappedhead\thmhead{#1}{#2}{#3}%
79      \the\thm@headpunct % space
80      \thmheadnl % possibly a newline.
81      \hskip\thm@headsep
82 %    }%
83      ]%
84   \ignorespaces}
```

Patched for css:

```
85 \def\@thm#1#2#3{%
86   \ifhmode\unskip\unskip\par\fi
87   \normalfont
88   \LWR@forcenewpage%                                    lwarp


89   \LWR@printpendingfootnotes%                           lwarp


90   \BlockClass{amsthmbody\LWR@thisthmstyle}%    lwarp
91   \trivlist
92   \let\thmheadnl\relax
93   \let\thm@swap\@gobble
94   \thm@notefont{\fontseries\mddefault\upshape}%
95   \thm@headpunct{.}% add period after heading
96   \thm@headsep 5\p@ plus\p@ minus\p@\relax
97   \thm@space@setup
98   #1% style overrides
99   \@topsep \thm@preskip                % used by thm head
100  \@topsepadd \thm@postskip            % used by \@endparenv
101  \def\@tempa{#2}\ifx\@empty\@tempa
102    \def\@tempa{\@oparg{\@begintheorem{#3}{}}[]}%
103  \else
104    \refstepcounter{#2}%
105    \def\@tempa{\@oparg{\@begintheorem{#3}{\csname the#2\endcsname}}[]}%
106  \fi
107  \@tempa%
108 }
```

cleveref patches \@thm to do \cref@thmoptarg if an optional argument is given. lwarp then patches \cref@thmoptarg \AtBeginDocument.

```
109 \AtBeginDocument{%
110 \def\cref@thmoptarg[#1]#2#3#4{%
111    \ifhmode\unskip\unskip\par\fi%
112    \normalfont%
113    \LWR@forcenewpage%                                       lwarp


114    \LWR@printpendingfootnotes%                              lwarp


115    \BlockClass{amsthmbody\LWR@thisthmstyle}%    lwarp
116    \trivlist%
117    \let\thmheadnl\relax%
118    \let\thm@swap\@gobble%
119    \thm@notefont{\fontseries\mddefault\upshape}%
120    \thm@headpunct{.}% add period after heading
121    \thm@headsep 5\p@ plus\p@ minus\p@\relax%
```

```
122     \thm@space@setup%
123     #2% style overrides
124     \@topsep \thm@preskip                % used by thm head
125     \@topsepadd \thm@postskip            % used by \@endparenv
126     \def\@tempa{#3}\ifx\@empty\@tempa%
127         \def\@tempa{\@oparg{\@begintheorem{#4}{}}[]}%
128     \else%
129         \refstepcounter[#1]{#3}%  <<< cleveref modification
130         \def\@tempa{\@oparg{\@begintheorem{#4}{\csname the#3\endcsname}}[]}%
131     \fi%
132     \@tempa
133 }%
134 }% AtBeginDocument
135
136 \def\@endtheorem{%
137     \endtrivlist%


138   \LWR@printpendingfootnotes%                          lwarp


139     \endBlockClass%
140     \@endpefalse%
141 }
```

Proof QED symbol:

```
142 \AtBeginDocument{
143 \@ifundefined{LWR@orig@openbox}{
144 \LetLtxMacro\LWR@orig@openbox\openbox
145 \LetLtxMacro\LWR@orig@blacksquare\blacksquare
146 \LetLtxMacro\LWR@orig@Box\Box
147
148 \def\openbox{\text{\HTMLunicode{25A1}}}% UTF-8 white box
149 \def\blacksquare{\text{\HTMLunicode{220E}}}% UTF-8 end-of-proof
150 \def\Box{\text{\HTMLunicode{25A1}}}% UTF-8 white box
151
152 \appto\LWR@restoreorigformatting{%
153     \LetLtxMacro\openbox\LWR@orig@openbox%
154     \LetLtxMacro\blacksquare\LWR@orig@blacksquare%
155     \LetLtxMacro\Box\LWR@orig@Box%
156 }% appto
157 }{}% @ifundefined
158 }% AtBeginDocument
```

Patched to add a <span>:

```
159 \DeclareRobustCommand{\qed}{%
160   \ifmmode \mathqed
161   \else
162 %     \leavevmode\unskip\penalty9999 \hbox{}\nobreak\hfill
163 %     \quad\hbox{\qedsymbol}%
164         \InlineClass{theoremendmark}{\qedsymbol}%   lwarp
165   \fi
166 }
```

Patched for css:

```
167 \renewenvironment{proof}[1][\proofname]{\par
168   \LWR@forcenewpage% lwarp
```

```
169    \LWR@printpendingfootnotes%                          lwarp
```

```
170      \BlockClass{amsthmproof}% lwarp
171      \LWR@newautopagelabel{page}%
172   \pushQED{\qed}%
173   \normalfont \topsep6\p@\@plus6\p@\relax
174   \trivlist
175   \item[
176        \InlineClass{amsthmproofname}{#1\@addpunct{.}}]\ignorespaces% changes
177 }{%
178   \popQED\endtrivlist%
```

```
179    \LWR@printpendingfootnotes%                          lwarp
```

```
180    \endBlockClass% lwarp
181    \@endpefalse
182 }
```

File 23   `lwarp-anonchap.sty`

§ 132   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 packages 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`

§ 133   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**

§ 134   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-ar.sty**

§ 135   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 27 **`lwarp-arabicfront.sty`**

§ 136   Package **arabicfront**

arabicfront (*Pkg*)   arabicfront is ignored.

for HTML output: 1 `\LWR@ProvidesPackageDrop{arabicfront}[2006/09/03]`

File 28   **`lwarp-array.sty`**

§ 137   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 29 **`lwarp-arydshln.sty`**

§ 138 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.

for HTML output: 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 30 **`lwarp-asymptote.sty`**

§ 139   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 \xpatchcmd{\asyinclude}
8     {\begingroup}
9     {\begin{lateximage}[-asymptote-~\PackageDiagramAltText]}
10    {}
11    {\LWR@patcherror{asymptote}{asyinclude-begingroup}}
12
13 \xpatchcmd{\asyinclude}
14    {\endgroup}
15    {\end{lateximage}}
16    {}
17    {\LWR@patcherror{asymptote}{asyinclude-endgroup}}
```

File 31   **lwarp-atbegshi.sty**

§ 140   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 32   **`lwarp-attachfile.sty`**

§ 141   Package   **attachfile**

(*Emulates or patches code by* SCOTT PAKIN.)

attachfile (*Pkg*)   attachfile is patched for use by lwarp.

⚠   Metadata is ignored for now.

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{#1}{\LWR@attachfile@appearance}%
33 }
```

File 33 **`lwarp-attachfile2.sty`**

§ 142   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{#1}{\LWR@attachfile@appearance}%
45 }
```

Modified for text color:

```
46 \DeclareRobustCommand{\notextattachfile}[2][]{%
47   \begingroup
48     \atfi@setup{#1}%
49     \ifatfi@print
50       \leavevmode
51       \begingroup
52         \HyColor@UseColor\atfi@color@tex
53         \LWR@textcurrentcolor{#2}%          lwarp
54 % \strut
55       \endgroup
56 %    \else
57 %      \sbox\ltx@zero{#2\strut}%
58 %      \makebox[\wd0]{}%
59     \fi
60   \endgroup
61 }
```

Modified to draw the icon:

```
62 \DeclareRobustCommand{\noattachfile}[1][]{%
63   \begingroup
64     \atfi@setup{#1}%
65     \atfi@set@appearance@icon
66     \ifatfi@print
67       \LWR@attachfile@appearance%      lwarp
68 %       \expandafter
69 %       \atfi@refxform\csname atfi@appobj@\atfi@icon@icon\endcsname
70 %     \else
71 %       \makebox[\atfi@appearancewidth]{}%
72     \fi
73   \endgroup
74 }
```

File 34   **`lwarp-authblk.sty`**

§ 143   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 69.8.

(*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 35 **`lwarp-autobreak.sty`**

§ 144 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 36 **`lwarp-autonum.sty`**

§ 145 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 37 **`lwarp-awesomebox.sty`**

§ 146   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 38   **`lwarp-axessibility.sty`**

§ 147   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
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 39 **`lwarp-axodraw2.sty`**

§ 148   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 40   **`lwarp-backnaur.sty`**

§ 149   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 41   `lwarp-backref.sty`

§ 150   Package   # backref

(*Emulates or patches code by* DAVID CARLISLE AND SEBASTIAN RAHTZ.)

backref (*Pkg*)   backref is patched for use by lwarp.

⚠ loading   Note that backref must be explicitly loaded, and is not automatically loaded by hyperref when generating HTML output.

for HTML output: `1 \LWR@ProvidesPackagePass{backref}[2016/05/21]`

Force the hyperref option:

```
2 \def\backref{}
3
4 \long\def\hyper@section@backref#1#2#3{%
5    \LWR@refwithsection{#3}%
6 }
7
8 \let\backrefxxx\hyper@section@backref
```

File 42   `lwarp-balance.sty`

§ 151   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 43   `lwarp-bbding.sty`

§ 152   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@FourClowerOpen}{\LWR@bbdingsymbol{126}         {273F}}
91 \newcommand{\LWR@HTML@FourClowerSolid}{\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{FourClowerOpen}
217 \LWR@formatted{FourClowerSolid}
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 44 **`lwarp-beamerarticle.sty`**

§ 153   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 45 **`lwarp-biblatex.sty`**

§ 154   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{
8
9 \let\blx@anchors\@empty
10 \protected\def\blx@anchor{%
11     \xifinlist{\the\c@refsection @\abx@field@entrykey}{\blx@anchors}
12         {}
13         {\listxadd\blx@anchors{\the\c@refsection @\abx@field@entrykey}%
14         \hypertarget{cite.\the\c@refsection @\abx@field@entrykey}{}}}
15
16 \protected\def\blx@imc@bibhyperref{%
17     \@ifnextchar[%
18         {\blx@bibhyperref}
19         {\blx@bibhyperref[\abx@field@entrykey]}}%
20
21 \long\def\blx@bibhyperref[#1]#2{%
22 %         \blx@sfsave
23         \hyperlink{cite.\the\c@refsection @#1}{%
24 %             \blx@sfrest
25         #2%
26 %         \blx@sfsave
27     }%
28 % \blx@sfrest%
29 }%% \def\blx@nohyperref[#1]#2{#2}%
30
31 \protected\long\def\blx@imc@bibhyperlink#1#2{%
32 %         \blx@sfsave
33         \hyperlink{cite.\the\c@refsection:#1}{%
34 %         \blx@sfrest
35         #2%
36 %         \blx@sfsave
37     }%
38 %         \blx@sfrest%
39 }%
40
41 \protected\long\def\blx@imc@bibhypertarget#1#2{%
42 %         \blx@sfsave%
43         \hypertarget{cite.\the\c@refsection:#1}{%
```

```
44 %          \blx@sfrest
45            #2%
46 %          \blx@sfsave%
47          }%
48 %          \blx@sfrest%
49 }
50
51 \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.

```
52 \xpretocmd{\blx@citecmdinit}
53      {\LWR@newautopagelabel{page}}%
54      {}
55      {\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.

```
56 \xpatchcmd{\blx@addbackref@i}
57      {\thepage}
58      {\theLWR@previousautopagelabel}% ref to the most recent object
59      {}
60      {\LWR@patcherror{biblatex}{blx@addbackref@i A}}
61
62 \xpatchcmd{\blx@addbackref@i}
63      {\c@page}
64      {\c@LWR@previousautopagelabel}% ref to the most recent object
65      {}
66      {\LWR@patcherror{biblatex}{blx@addbackref@i B}}
```

The following patches are for back page references.

```
67 \DeclareListFormat{pageref}{%
68   \ifnumless{\abx@pagerefstyle}{0}
69      {\usebibmacro{list:plain}%
70       \ifhyperref
71        {%
72 %              \hyperlink{page.#1}{#1}%
73             \LWR@refwithsection{\BaseJobname-autopage-#1}%  lwarp
74        }
75        {#1}}
76      {\ifnumequal{\value{listcount}}{1}
77         {\usebibmacro{pageref:init}}
78         {}%
79       \usebibmacro{pageref:comp}{#1}%
80       \ifnumequal{\value{listcount}}{\value{liststop}}
81         {\usebibmacro{pageref:dump}}
82         {}}}
83
84 \renewbibmacro*{pageref:comp}[1]{%
85   \numdef\abx@range@prev{\abx@range@prev+1}%
86   \ifinteger{#1}
87      {\def\abx@range@num{#1}%
88       \def\abx@range@this{1}%
89       \ifnumequal{\abx@range@this}{\abx@range@last}
90         {}
91         {\def\abx@range@prev{-1}}}
```

```
 92      {\ifrmnum{#1}
 93        {\numdef\abx@range@num{\rmntonum{#1}}%
 94         \def\abx@range@this{2}%
 95         \ifnumequal{\abx@range@this}{\abx@range@last}
 96           {}
 97           {\def\abx@range@prev{-1}}}
 98        {\undef\abx@range@num
 99         \def\abx@range@this{0}%
100         \def\abx@range@prev{-1}}}%
101   \ifdef\abx@range@num
102     {\ifnumequal{\abx@range@num}{\abx@range@prev}
103        {\def\abx@range@hold{#1}%
104         \numdef\abx@range@diff{\abx@range@diff+1}}
105        {\usebibmacro{pageref:dump}%
106         \ifnumgreater{\abx@range@last}{-1}
107           {\printdelim{multilistdelim}}
108           {}%
109         \ifhyperref
110 %          {\hyperlink{page.#1}{#1}}
111           {\LWR@refwithsection{\BaseJobname-autopage-#1}}%  lwarp
112           {#1}}%
113      \edef\abx@range@prev{\abx@range@num}}
114     {\usebibmacro{pageref:dump}%
115      \ifnumgreater{\abx@range@last}{-1}
116        {\printdelim{multilistdelim}}
117        {}%
118      \ifhyperref
119 %        {\hyperlink{page.#1}{#1}}
120        {\LWR@refwithsection{\BaseJobname-autopage-#1}}%  lwarp
121        {#1}%
122      \def\abx@range@prev{-1}}%
123   \edef\abx@range@last{\abx@range@this}}
124
125 \renewbibmacro*{pageref:dump}{%
126   \ifnumgreater{\abx@range@diff}{0}
127     {\ifcase\abx@pagerefstyle\relax % two
128        \bibrangedash
129        \ifhyperref
130 %          {\hyperlink{page.\abx@range@hold}{\abx@range@hold}}
131          {\LWR@refwithsection{\BaseJobname-autopage-\abx@range@hold}}%  lwarp
132          {\abx@range@hold}%
133     \or % three
134        \ifnumless{\abx@range@diff}{2}
135          {\printdelim{multilistdelim}}
136          {\bibrangedash}%
137        \ifhyperref
138 %          {\hyperlink{page.\abx@range@hold}{\abx@range@hold}}
139          {\LWR@refwithsection{\BaseJobname-autopage-\abx@range@hold}}%  lwarp
140          {\abx@range@hold}%
141     \or % two+
142        \ifnumless{\abx@range@diff}{2}
143          {\sqspace
144           \ifhyperref
145 %            {\hyperlink{page.\abx@range@hold}{\bibstring{sequens}}}
146          {\LWR@refwithsection{\BaseJobname-autopage-\abx@range@hold}}%  lwarp
147            {\bibstring{sequens}}}
148          {\bibrangedash
149           \ifhyperref
150 %            {\hyperlink{page.\abx@range@hold}{\abx@range@hold}}
151          {\LWR@refwithsection{\BaseJobname-autopage-\abx@range@hold}}%  lwarp
```

```
152           {\abx@range@hold}}%
153     \or % three+
154       \ifnumless{\abx@range@diff}{2}
155         {\sqspace
156          \ifhyperref
157 %            {\hyperlink{page.\abx@range@hold}{\bibstring{sequens}}}
158         {\LWR@refwithsection{\BaseJobname-autopage-\abx@range@hold}}% lwarp
159            {\bibstring{sequens}}}
160         {\ifnumless{\abx@range@diff}{3}
161           {\sqspace
162            \ifhyperref
163 %              {\hyperlink{page.\abx@range@hold}{\bibstring{sequentes}}}
164           {\LWR@refwithsection{\BaseJobname-autopage-\abx@range@hold}}% lwarp
165              {\bibstring{sequentes}}}
166           {\bibrangedash
167            \ifhyperref
168 %              {\hyperlink{page.\abx@range@hold}{\abx@range@hold}}
169           {\LWR@refwithsection{\BaseJobname-autopage-\abx@range@hold}}% lwarp
170              {\abx@range@hold}}}%
171     \else % all+
172       \ifnumless{\abx@range@diff}{2}
173         {\sqspace
174          \ifhyperref
175 %            {\hyperlink{page.\abx@range@hold}{\bibstring{sequens}}}
176         {\LWR@refwithsection{\BaseJobname-autopage-\abx@range@hold}}% lwarp
177            {\bibstring{sequens}}}
178         {\sqspace
179          \ifhyperref
180 %            {\hyperlink{page.\abx@range@hold}{\bibstring{sequentes}}}
181         {\LWR@refwithsection{\BaseJobname-autopage-\abx@range@hold}}% lwarp
182            {\bibstring{sequentes}}}%
183     \fi
184     \def\abx@range@diff{0}}
185   {}}
186
187 }% \AfterPreamble
```

File 46   `lwarp-bibunits.sty`

§ 155   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 47   `lwarp-bigdelim.sty`

§ 156   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 48   `lwarp-bigfoot.sty`

§ 157   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 49   `lwarp-bigstrut.sty`

§ 158   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 50   **`lwarp-bitpattern.sty`**

§ 159   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 \xpatchcmd\{\bitpattern}
3     {\begingroup}
4     {\begin{lateximage}[-bitpattern-~\PackageDiagramAltText]}
5     {}
6     {\LWR@patcherror{bitpattern}{bitpattern}}
7
8 \xpatchcmd\{\bp@Done}
9     {\endgroup}
10    {\end{lateximage}}
11    {}
12    {\LWR@patcherror{bitpattern}{bp@Done}}
```

File 51   **`lwarp-blowup.sty`**

§ 160   Package   # blowup

blowup (*Pkg*)   blowup is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{blowup}[2018/01/02]`

`2 \newcommand*\blowUp[1]{}`

File 52   **`lwarp-bm.sty`**

§ 161   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 53   **`lwarp-booklet.sty`**

§ 162   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 54   **`lwarp-bookmark.sty`**

§ 163   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 55 **lwarp-booktabs.sty**

§ 164 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 56   `lwarp-bophook.sty`

§ 165   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 57   `lwarp-bounddvi.sty`

§ 166   Package   # bounddvi

bounddvi (*Pkg*)   bounddvi is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{bounddvi}[2016/12/28]

File 58   `lwarp-boxedminipage.sty`

§ 167   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 59   `lwarp-boxedminipage2e.sty`

§ 168   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 60   `lwarp-braket.sty`

§ 169   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 61   `lwarp-breakurl.sty`

§ 170   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}%
7     \LWR@subhyperreftext{#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 62   `lwarp-breqn.sty`

§ 171   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.

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{lateximage}[-breqn dmath- \MathImageAltText]
12 }
13
14 \AfterEndEnvironment{dmath}{
15    \end{lateximage}\end{BlockClass}
16 }
17
18 \BeforeBeginEnvironment{dmath*}{
19    \begin{BlockClass}{displaymath}
20    \LWR@newautoidanchor%
21    \booltrue{LWR@indisplaymathimage}%
22    \begin{lateximage}[-breqn dmath*- \MathImageAltText]
23 }
24
25 \AfterEndEnvironment{dmath*}{
26    \end{lateximage}\end{BlockClass}
27 }
28
29 \BeforeBeginEnvironment{dseries}{
30    \begin{BlockClass}{displaymathnumbered}
31    \LWR@newautoidanchor%
32    \booltrue{LWR@indisplaymathimage}%
33    \begin{lateximage}[-breqn dseries- \MathImageAltText]
34 }
35
36 \AfterEndEnvironment{dseries}{
37    \end{lateximage}\end{BlockClass}
38 }
39
40 \BeforeBeginEnvironment{dseries*}{
41    \begin{BlockClass}{displaymath}
42    \LWR@newautoidanchor%
43    \booltrue{LWR@indisplaymathimage}%
44    \begin{lateximage}[-breqn dseries*- \MathImageAltText]
45 }
46
47 \AfterEndEnvironment{dseries*}{
48    \end{lateximage}\end{BlockClass}
49 }
50
51 \BeforeBeginEnvironment{dgroup}{
52    \begin{BlockClass}{displaymath}
53    \LWR@newautoidanchor%
```

```
54     \booltrue{LWR@indisplaymathimage}%
55     \begin{lateximage}[-breqn dgroup- \MathImageAltText]
56 }
57
58 \AfterEndEnvironment{dgroup}{
59     \end{lateximage}\end{BlockClass}
60 }
61
62 \BeforeBeginEnvironment{dgroup*}{
63     \begin{BlockClass}{displaymath}
64     \LWR@newautoidanchor%
65     \booltrue{LWR@indisplaymathimage}%
66     \begin{lateximage}[-breqn dgroup*- \MathImageAltText]
67 }
68
69 \AfterEndEnvironment{dgroup*}{
70     \end{lateximage}\end{BlockClass}
71 }
```

File 63   **`lwarp-bsheaders.sty`**

§ 172   Package   # bsheaders

bsheaders (*Pkg*)   bsheaders is ignored.

for HTML output: 1 `\LWR@ProvidesPackageDrop{bsheaders}[1997/10/06]`

File 64   **`lwarp-bussproofs.sty`**

§ 173   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         {\LWR@doequation{\BODY}{prooftree}}%
7         [\LWR@doendequation{prooftree}]
8     \LWR@formattedenv{prooftree}
9 }{% SVG HTML
10     \BeforeBeginEnvironment{prooftree}{%
11         \begin{lateximage}[-bussproofs-~\PackageDiagramAltText]%
12     }
13     \AfterEndEnvironment{prooftree}{\end{lateximage}}
14 }
```

File 65 **`lwarp-bxpapersize.sty`**

§174 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 66 **`lwarp-bytefield.sty`**

§175 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 67 **`lwarp-cancel.sty`**

§176 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 68 **`lwarp-canoniclayout.sty`**

§ 177 Package **canoniclayout**

canoniclayout (*Pkg*) canoniclayout is ignored.

for HTML output: ₁ \LWR@ProvidesPackageDrop{canoniclayout}[2011/11/05]

```
2 \newcommand*{\currentfontletters}{}
3 \newcommand*{\charactersperpage}{}
```

File 69 **`lwarp-caption.sty`**

§ 178 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}[2020/10/26]
```

```
10 \long\def\caption@iibox@#1#2#3#4{%
```

```
11 %    \setbox\@tempboxa\hbox{#4}%
12   \caption@iiibox{#1}{#2}{#3}%
13 %        [\wd\@tempboxa]%
14        []%                              lwarp
15        [\captionbox@hj@default]%
16 %        {\unhbox\@tempboxa}%
17        {{#4}}%                          lwarp
18 }


19 \long\def\caption@iiiibox#1#2#3#4#5[#6][#7]#8{%
20   \begingroup
21   #1*% set \caption@position
22   \caption@iftop{%
23     \endgroup

24     \minipagefullwidth%                 lwarp
25     \parbox[t]{\linewidth}{%
26       #1\relax
27       \caption@setposition t%
28 %        #2%
29            {\caption#4{#5}}%
30 %        \captionbox@hrule
31 %        \csname caption@hj@#7\endcsname
32       #8%
33     }%
34   }{%
35     \endgroup

36 %      \parbox[b]{#6}{%
37     \minipagefullwidth%                 lwarp
38     \parbox[b]{\linewidth}{%            lwarp
39       #1\relax
40       \caption@setposition b%
41 %        \csname caption@hj@#7\endcsname
42       #8%
43 %        \captionbox@hrule
44 %        #3
45            {\caption#4{#5}}%
46     }%
47   }%
48 }
```

\caption@makecaption

```
49 \long\def\caption@makecaption#1#2{%
50 %    \caption@make@above
51   \caption@@make{#1}{#2}%
52 %    \caption@make@below
53 }
54
55 \AtBeginDocument{
56     \let\@makecaption\caption@makecaption
57 }
```

Appended to look ahead to the next token for \centering, etc:

```
58 \AtBeginDocument{
59 \xapptocmd{\@xfloat}
60    {\LWR@futurenonspacelet\LWR@mynexttoken\LWR@floatalignment}
```

```
61        {}
62        {\LWR@patcherror{caption}{@xfloat}}
63
64  \xapptocmd{\@xdblfloat}
65        {\LWR@futurenonspacelet\LWR@mynexttoken\LWR@floatalignment}
66        {}
67        {\LWR@patcherror{caption}{@xdblfloat}}
68  }


69  \long\def\caption@@@text#1#2#3[#4]#5{%
70        \begin{BlockClass}{figurecaption}%          lwarp
71      \begingroup
72      #3{\csname c@#1\endcsname #4\relax}%
73      #2{\caption@fnum{#1}}{#5}%
74      \endgroup%
75        \end{BlockClass}%                           lwarp
76  }
```

Updates for late patches for scrextend:

```
77  \caption@AtBeginDocument{
78  \IfPackageLoadedTF{lwarp-scrextend}{
79        \LetLtxMacro\captionbelow\caption
80        \LetLtxMacro\captionabove\caption
81        \LetLtxMacro\captionofbelow\captionof
82        \LetLtxMacro\captionofabove\captionof
83  }{}
84  }
```

File 70 **`lwarp-caption3.sty`**

§ 179   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}[2020/10/21]

\caption@@@make   {⟨*caption label*⟩} {⟨*caption text*⟩}

```
2  \IfPackageAtLeastTF{caption3}{2020/08/23}{
3  \renewcommand\caption@@@make[2]{%
4  \LWR@traceinfo{caption@@@make}%

5      \LWR@stoppars%                                 lwarp

6  %    \sbox\@tempboxa{#1}%
7  %    \ifdim\wd\@tempboxa=\z@
8  %      \caption@set{labelseparator}{none}%
9  %    \fi
10  \caption@ifempty{#2}{%
11      \caption@set{labelseparator}{none}%
12      \caption@set{textformat}{simple}%
13  }%
14  \caption@labelseparator % defines \caption@iflabelfont,
```

```
15 %        \caption@labelsep and \caption@labelsep@name
16 %        (the latter is needed by \caption@fmt)
17 %
18 %    \@setpar{\@@par\caption@@par}\caption@@par
19   \caption@applyfont
```

\caption@fmt with plain format is defined as {#1#2#3\par}:

```
20 %   \caption@fmt
21     {\ifcaption@star\else
22        \begingroup
23          \captionlabelfont

24          \LWR@isolate{#1}%                          lwarp
25        \endgroup
26     \fi}%
27     {\ifcaption@star\else
28        \begingroup
29          \caption@iflabelfont\captionlabelfont
30          \relax\caption@labelsep
31        \endgroup
32     \fi}%
33     {{\captiontextfont

34        \let\\\newline%                             lwarp
35 %
36      \caption@textstart

37 %      \caption@ifstrut
38 %        {\vrule\@height\ht\strutbox\@width\z@}%
39 %        {}%
40 %      \nobreak\hskip\z@skip % enable hyphenation

41      \LWR@isolate{\caption@textformat{#2}}%     lwarp

42 %      \caption@ifstrut
43 %        {\ifhmode\@finalstrut\strutbox\fi}%
44 %        {}%
45      \caption@textend}}%

46    \LWR@startpars%                               lwarp
47 \LWR@traceinfo{caption@@@make done}%
48 }
49 }% later than 2020/08/23
50 {% earlier than 2020/08/23
51 \renewcommand\caption@@@make[2]{%
52 \LWR@traceinfo{caption@@@make}%
53    \LWR@stoppars%                                lwarp
54 %  \sbox\@tempboxa{#1}%
55 %  \ifdim\wd\@tempboxa=\z@
56 %    \let\caption@lsep\relax
57 %  \fi
58   \caption@ifempty{#2}{%
59   \let\caption@lsep\@empty
60   \let\caption@tfmt\@firstofone
61   }%
62 %  \@setpar{\@@par\caption@@par}\caption@@par
63   \caption@applyfont
```

\caption@fmt with plain format is defined as {#1#2#3\par}:

```
64 %   \caption@fmt
```

```
65      {\ifcaption@star\else
66        \begingroup
67          \captionlabelfont
68            \LWR@isolate{#1}%                              lwarp
69          \endgroup
70      \fi}%
71      {\ifcaption@star\else
72        \begingroup
73          \caption@iflf\captionlabelfont
74          \relax
75          \caption@lsep
76        \endgroup
77      \fi}%
78      {{%
79          \captiontextfont
80          \let\\\newline%                              lwarp
81 %        \caption@ifstrut
82 %          {\vrule\@height\ht\strutbox\@width\z@}%
83 %          {}%
84 %        \nobreak\hskip\z@skip % enable hyphenation
85          \LWR@isolate{\caption@tfmt{#2}}%         lwarp
86 %        \caption@ifstrut
87 %          {\ifhmode\@finalstrut\strutbox\fi}%
88 %          {}%
89        }}%
90      \LWR@startpars%                                  lwarp
91 \LWR@traceinfo{caption@@@make done}%
92 }
93 }% earlier than 2020/08/23
```

\caption@@make@   {⟨⟩} {⟨⟩}

```
94 \renewcommand{\caption@@make@}[2]{%
95   \caption@stepthecounter%
96   \caption@beginhook%
97 %   \caption@box\hsize{%
98 %   \caption@singlelinecheck{\caption@slc{#1}{#2}\caption@singleline\caption@multiline}{\caption@mu
99 %     \caption@calcmargin
100 %     \caption@make@leftmargin
101 %     \caption@make@parbox{%
102 %       \caption@make@indention
103        \caption@@@make{#1}{#2}%
104 %     }
105 %     \caption@make@rightmargin
106 %   }%
107   \caption@endhook%
108 }
```

```
109 \DeclareCaptionBox{none}{#2}
110 \DeclareCaptionBox{parbox}{%
111     #2%
112 }
```

```
113 \DeclareCaptionBox{colorbox}{%
114     #2%
115 }
```

File 71 **`lwarp-cases.sty`**

§ 180   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 72   **`lwarp-ccicons.sty`**

§ 181   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 73   `lwarp-centerlastline.sty`

§ 182   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 74   `lwarp-centernot.sty`

§ 183   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 75   `lwarp-changebar.sty`

§ 184   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 76 **`lwarp-changelayout.sty`**

§ 185   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 77   **`lwarp-changepage.sty`**

§ 186   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 78   **`lwarp-changes.sty`**

§ 187   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 \renewcommand{\listofchanges}[1][\@empty]{%
12 \setkeys{Changes@loc}{#1}%
13 \ifbool{Changes@optiondraft}%
14 {%
15 \IfIsInList{\Changes@loc@style}{list|summary|compactsummary}%
16 {}%
17 {%
18 \PackageWarning{changes}{Wrong style for list of changes: '\Changes@loc@style', using 'list' instead.}%
19 \def\Changes@loc@style{}%
20 }%
21 \IfIsEmpty{\Changes@loc@style}%
22 {\def\Changes@loc@style{list}}%
23 {}%
24 \IfStrEq{\Changes@loc@show}{all}%
25 {\def\Changes@loc@show{added|deleted|replaced|highlight|comment}}%
26 {}%
27 \IfIsInList{\Changes@loc@show}{added|deleted|replaced|highlight|comment}%
```

```
28 {}%
29 {%
30 \PackageWarning{changes}{Wrong show-value for list of changes: '\Changes@loc@show', using 'all' instead
31 \def\Changes@loc@show{}%
32 }%
33 \IfIsEmpty{\Changes@loc@show}%
34 {\def\Changes@loc@show{added|deleted|replaced|highlight|comment}}%
35 {}%
36 \IfIsEmpty{\Changes@loc@title}%
37 {%
38 \IfStrEq{\Changes@loc@style}{list}%
39 {\def\Changes@heading{\listofchangesname}}{}%
40 \IfStrEq{\Changes@loc@style}{summary}%
41 {\def\Changes@heading{\summaryofchangesname}}{}%
42 \IfStrEq{\Changes@loc@style}{compactsummary}%
43 {\def\Changes@heading{\compactsummaryofchangesname}}{}%
44 }%
45 {\def\Changes@heading{\Changes@loc@title}}%
46 \section*{\Changes@heading}
47 \IfIsInList{\Changes@loc@style}{list}%
48 {%
49 \IfFileExists{\jobname.\Changes@locextension}%
50 {%
51 \newread\Changes@InFile%
52 \openin\Changes@InFile=\jobname.\Changes@locextension%
53 \loop\unless\ifeof\Changes@InFile%
54 \read\Changes@InFile to \Changes@Line%
55 \ifeof\Changes@InFile\else%
56 \Changes@Line%
57 \fi
58 \repeat
59 \closein\Changes@InFile%
60 }{%
61 \emph{\changesnoloc}%
62 \PackageWarning{changes}{LaTeX rerun needed for list of changes}%
63 }%
64 }{}%
65 \IfIsInList{\Changes@loc@style}{summary|compactsummary}%
66 {%
67 \IfFileExists{\jobname.\Changes@socextension}%
68 {%
69 \newread\Changes@InFile%
70 \openin\Changes@InFile = \jobname.\Changes@socextension%
71 \loop\unless\ifeof\Changes@InFile%
72 \read\Changes@InFile to \Changes@Line%
73 \ifeof\Changes@InFile\else%
74 \expandafter\changes@chopline\Changes@Line\\%
75 \textbf{%
76 \IfIsColored%
77 {\color{\Changes@Incolor}}%
78 {}%
79 \IfIsAnonymous{\Changes@Inid}%
80 {%
81     \LWR@textcurrentcolor{%      lwarp
82         \changesauthorname: \changesanonymousname%
83     }%  lwarp
84 }%
85 {%
86     \LWR@textcurrentcolor{%      lwarp
87 \changesauthorname: \Changes@Inid%
```

```
88       }%  lwarp
89 \IfIsEmpty{\Changes@Inname}%
90 {}%
91 { %
92      \LWR@textcurrentcolor{%       lwarp
93 (\Changes@Inname)%
94      }%  lwarp
95 }%
96 }%
97 }\\%
98 \numdef{\Changes@InSum}{0}%
99 \renewcommand*{\do}[1]{%
100 \numdef{\Changes@InSum}{\Changes@InSum + \csuse{Changes@In########1}}%
101 }%
102 \expandafter\dopsvlist\expandafter{\Changes@loc@show}%
103 \ifnumcomp{\Changes@InSum}{=}{0}%
104 {%
105 % \parbox{\Changes@summary@width}{% lwarp
106      \changesnochanges%
107 % }%     lwarp
108 % \\[1ex]%     lwarp
109      \par%  \lwarp
110 }%
111 {%
112 \numdef{\Changes@InCount}{0}%
113 \renewcommand*{\do}[1]{%
114 \numdef{\Changes@InCount}{\Changes@InCount + \csuse{Changes@In########1}}%
115 \ifboolexpr{%
116 not test {\IfStrEq{\Changes@loc@style}{compactsummary}} or%
117 test {\ifnumgreater{\csuse{Changes@In########1}}{0}}%
118 }%
119 {%
120 % \parbox{\Changes@summary@width}{%  lwarp
121 \csuse{changes########1name}~%
122 % \let\cleaders\leaders\dotfill~%    lwarp
123 \dotfill~%    \lwarp
124 \csuse{Changes@In########1}%
125 % }%     lwarp
126 % \ifnumless{\Changes@InCount}{\Changes@InSum}%     lwarp
127 {\\}%
128 % {\\[1ex]}%    lwarp
129 }%
130 {}%
131 }%
132 \expandafter\dopsvlist\expandafter{\Changes@loc@show}%
133      \par%   lwarp
134 }%
135 \fi%
136 \repeat
137 \closein\Changes@InFile%
138 }{%
139 \emph{\changesnosoc}%
140 \PackageWarning{changes}{LaTeX rerun needed for summary of changes}%
141 }%
142 }{}%
143 }{}%
144 }
145
146
147 \renewcommand{\Changes@Markup@comment}[3]{%
```

```
148 \IfStrEq{\Changes@optioncommentmarkup}{todo}%
149 {%
150 \IfIsColored%
151 {\colorlet{Changes@todocolor}{authorcolor}}%
152 {\colorlet{Changes@todocolor}{black}}%
153 \todo[color=Changes@todocolor!10, bordercolor=Changes@todocolor, linecolor=Changes@todocolor!70, noli
154 }{}%
155 \IfStrEq{\Changes@optioncommentmarkup}{margin}%
156 {%
157 \marginpar{%
158 \IfIsColored%
159 {\leavevmode\color{authorcolor}}%
160 {}%
161     \LWR@textcurrentcolor{%        lwarp
162 \textbf{[\IfIsAnonymous{#2}{}{#3~}\arabic{Changes@commentCount#2}]:} #1%
163     }%  lwarp
164 }%
165 }{}%
166 \IfStrEq{\Changes@optioncommentmarkup}{footnote}%
167 {%
168 \footnote{%
169     \LWR@textcurrentcolor{%        lwarp
170 \textbf{[\IfIsAnonymous{#2}{}{#3~}\arabic{Changes@commentCount#2}]:} #1%
171     }%  lwarp
172 }%
173 }{}%
174 \IfStrEq{\Changes@optioncommentmarkup}{uwave}%
175 {%
176 {%
177 \IfIsColored%
178 {\color{authorcolor}}%
179 {}%
180 \allowbreak%
181 \uwave{%
182 \textbf{[\IfIsAnonymous{#2}{}{#3~}\arabic{Changes@commentCount#2}]:} #1%
183 }%
184 }%
185 }{}%
186 }
187
188 \renewrobustcmd\Changes@output}[7]{%
189 \ifbool{Changes@optiondraft}%
190 {%
191 \Changes@check@author{#2}%
192 \Changes@set@color{#2}%
193 {%
194 \IfIsInList{#1}{added|deleted|replaced|highlight}%
195 {%
196 \IfIsEmpty{#5}%
197 {%
198 \IfIsAuthorEmptyAtPosition{#2}{left}%
199 {}%
200 {{%
201 \IfIsColored%
202 {\color{authorcolor}}%
203 {}%
204     \LWR@textcurrentcolor{%        lwarp
205 \Changes@Markup@author{\Changes@output@author@position{#2}{left}}%
206     }%  lwarp
207 }}%
```

```
208 }{}%
209 {%
210 \IfStrEq{#1}{highlight}%
211 {}{%
212 \IfIsColored%
213 {\color{authorcolor}}%
214 {}%
215 }%
216     \LWR@textcurrentcolor{%        lwarp
217 \IfStrEq{#1}{added}{\Changes@Markup@added{#3}}{}%
218 \IfStrEq{#1}{deleted}{\Changes@Markup@deleted{#4}}{}%
219 \IfStrEq{#1}{replaced}{{\Changes@Markup@added{#3}}\allowbreak\Changes@Markup@deleted{#4}}{}%
220 \IfStrEq{#1}{highlight}{\Changes@Markup@highlight{#3}}{}%
221     }%  lwarp
222 }%
223 \IfIsEmpty{#5}%
224 {%
225 \IfIsAuthorEmptyAtPosition{#2}{right}%
226 {}%
227 {{%
228 \IfIsColored%
229 {\color{authorcolor}}%
230 {}%
231     \LWR@textcurrentcolor{%        lwarp
232 \Changes@Markup@author{\Changes@output@author@position{#2}{right}}%
233     }%  lwarp
234 }}%
235 }{}%
236 \stepcounter{Changes@#1Count#2}%
237 }{}%
238 \IfIsEmpty{#5}%
239 {}%
240 {%
241 \stepcounter{Changes@commentCount#2}%
242 \Changes@set@commentcount{#2}%
243 \Changes@Markup@comment%
244 {#5}%
245 {#2}%
246 {\Changes@output@author{#2}}%
247 }%
248 }%
249 \IfIsEmpty{#2}%
250 {\def\Changes@locid{}}%
251 {\def\Changes@locid{~(#2)}}%
252 \addtocontents{\Changes@locextension}{\protect\ChangesListline{#1}{#6\Changes@locid}{#7}{\thepage}}%
253 }%
254 {%
255 \IfIsEmpty{#3}%
256 {\@bsphack\@esphack}%
257 {#3}%
258 }%
259 }
```

---

§ 188   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 80 **`lwarp-chapterbib.sty`**

§ 189 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 81 **`lwarp-chemfig.sty`**

§ 190 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      \xpretocmd{\CF_chemfiga}
17          {\begin{lateximage}[-chemfig-~\PackageDiagramAltText]}
18          {}{\LWR@patcherror{chemfig}{CF_chemfiga}}
19
20      \xpatchcmd{\CF_chemfigb}
21          {\let\CF_flipstate\CF_zero}
22          {\end{lateximage}\let\CF_flipstate\CF_zero}
23          {}{\LWR@patcherror{chemfig}{CF_chemfigb}}
24
25      \GlobalLetLtxMacro\LWR@chemfig@origCF_lewisc\CF_lewisc
26      \gdef\CF_lewisc#1,#2\_nil{%
27      \begin{lateximage}[-chemfig-~\PackageDiagramAltText]%
28      \LWR@chemfig@origCF_lewisc#1,#2\_nil
29      \end{lateximage}
30      }
31
32      \gpreto{\schemestart}{%
33          \begin{lateximage}[-chemfig-~\PackageDiagramAltText]%
34      }
35      \gappto{\CF_schemestop}{\end{lateximage}}
36
37 }% 2019/04/18 or newer
38 {% older than 2019/04/18
39
40      \LetLtxMacro\LWR@chemfig@origchemfig\chemfig
41
42      \DeclareDocumentCommand\chemfig{s O{} O{} m}{%
43          \begin{lateximage}[-chemfig-~\PackageDiagramAltText]%
44          \IfBooleanTF{#1}{%
45              \LWR@chemfig@origchemfig*[#2][#3]{#4}%
46          }{%
47              \LWR@chemfig@origchemfig[#2][#3]{#4}%
48          }
49          \end{lateximage}%
50      }
51
52      \LetLtxMacro\LWR@chemfig@origCF@lewis@b\CF@lewis@b
53
54      \def\CF@lewis@b#1#2{%
55      \begin{lateximage}[-chemfig-~\PackageDiagramAltText]%
56      \LWR@chemfig@origCF@lewis@b{#1}{#2}%
57      \end{lateximage}%
58      }
59
60      \preto{\schemestart}{%
61          \begin{lateximage}[-chemfig-~\PackageDiagramAltText]%
62      }
63      \appto{\CF@schemestop}{\end{lateximage}}
64
65 }% older than 2019/04/18
66
67 \catcode`\_=8%
68
69
70
71 \LetLtxMacro\LWR@chemfig@origchemleft\chemleft
72
73 \def\chemleft#1#2\chemright#3{%
74 \begin{lateximage}[-chemfig-~\PackageDiagramAltText]%
75 \LWR@chemfig@origchemleft#1#2\chemright#3%
```

```
76 \end{lateximage}%
77 }
78
79 \LetLtxMacro\LWR@chemfig@origchemup\chemup
80
81 \def\chemup#1#2\chemdown#3{%
82 \begin{lateximage}[-chemfig-~\PackageDiagramAltText]%
83 \LWR@chemfig@origchemup#1#2\chemdown#3%
84 \end{lateximage}%
85 }
```

File 82   **`lwarp-chemformula.sty`**

§ 191   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 MATHJAX   chemformula works best without MATHJAX. If MATHJAX is used, \displaymathother 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 \RenewDocumentCommand \ch { O{}m }
4   {%
```

To work inside align with \displaymathother, a simple version must be used to work with chemformula's adaptation to align.

```
5     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%  lwarp
6     {
7         \chemformula_ch:nn {#1} {#2}%   original
8     }
```

If used as the outer level, must temporarily ensure MATHJAX is disabled:

```
9     {
10        \begingroup%
11        \boolfalse{mathjax}%
```

An inline image is used, adjusted for the baseline:

```
12        \LWR@subsingledollar*{%  lwarp
```

```
13            \textbackslash{}%
14            ch%
15            \{%
16                \LWR@HTMLsanitizedetokenized{\detokenize{#2}}%
17            \}% alt text
18        }{%
19            \protect\LWR@HTMLsanitizedetokenized{%
20                \detokenize\expandafter{#1}%
21            }% add'l hashing
22        }%
23        {%
24            \chemformula_ch:nn {#1} {#2}%    original
25        }%
26        \endgroup%
27    }
28  }
```

\chcpd   Similar to \ch.

```
29 \IfPackageAtLeastTF{chemformula}{2019/10/13}{
30 \cs_gset_protected:Npn \chemformula_chcpd:nn #1#2
31  {
32    \begingroup%
33    \boolfalse{mathjax}%
34    \LWR@subsingledollar*{% lwarp
35        \textbackslash{}%
36        chcpd%
37        \{%
38            \LWR@HTMLsanitizedetokenized{\detokenize{#2}}%
39        \}%
40    }{%
41        \protect\LWR@HTMLsanitizedetokenized{\detokenize\expandafter{#1}}%
42    }{% original
43    \group_begin:
44      \tl_if_blank:nF {#2}
45        {
46          \keys_set:nn {chemformula} {#1}
47          \__chemformula_save_catcodes:
48          \__chemformula_sanitize:Nn
49            \l__chemformula_chemformula_tmpa_tl
50            {#2}
51          \__chemformula_input_compound_no_check:NV
52            \l__chemformula_compound_tl
53            \l__chemformula_chemformula_tmpa_tl
54          \__chemformula_prepare_output:NV
55            \l__chemformula_compound_tl
56            \l__chemformula_catcodes_tl
57          \chemformula_write:V \l__chemformula_compound_tl
58        }
59    \group_end:
60    }
61    \endgroup
62  }
63 }% later than 2019/10/13
64 {% earlier than 2019/10/13
65 % \changes{v0.903}{2021/12/18}{\pkg{chemformula}: Improved alt tag sanitization.}
66 \cs_gset_protected:Npn \chemformula_chcpd:nn #1#2
67  {
68    \begingroup%
69    \boolfalse{mathjax}%
```

```
70        \LWR@subsingledollar*{% lwarp
71            \textbackslash{}%
72            chcpd%
73            \{%
74                \LWR@HTMLsanitizedetokenized{\detokenize{#2}}%
75            \}%
76        }{%
77            \protect\LWR@HTMLsanitizedetokenized{\detokenize\expandafter{#1}}%
78        }{% original
79        \group_begin:
80          \tl_if_blank:nF {#2}
81            {
82              \keys_set:nn {chemformula} {#1}
83              \__chemformula_save_catcodes:
84              \__chemformula_sanitize:Nn
85                \l__chemformula_chemformula_tmpa_tl
86                {#2}
87              \__chemformula_input_compound_no_check:NV
88                \l__chemformula_compound_tl
89                \l__chemformula_chemformula_tmpa_tl
90              \__chemformula_prepare_output:N \l__chemformula_compound_tl
91              \chemformula_write:V \l__chemformula_compound_tl
92            }
93        \group_end:
94        }
95        \endgroup
96    }
97 }% earlier than 2019/10/13
```

\charrow   If standalone, appears in a regular lateximage.

```
98 \RenewDocumentCommand \charrow { mO{}O{} }
99 {
100      \begin{lateximage}[-chemformula- charrow]
101      \group_begin:
102        \__chemformula_draw_arrow:nnn {#1} {#2} {#3}
103      \group_end:
104      \end{lateximage}
105 }
```

\chname   If standalone, appears in a regular lateximage, hashed according to contents.

```
106 \RenewDocumentCommand \chname { R(){}R(){} }
107   {
108      \begin{lateximage}*[%
109          \textbackslash{}%
110          chname%
111          (\LWR@HTMLsanitizedetokenized{\detokenize{#1}})%
112          (\LWR@HTMLsanitizedetokenized{\detokenize{#2}})%
113      ]*%
114          \chemformula_chwritebelow:nn {#1} {#2}
115      \end{lateximage}
116   }
```

\chlewis   Placed inline, hashed according to contents and options.

```
117 \RenewDocumentCommand \chlewis { O{}mm }
118   {
119      \begingroup%
```

```
120        \boolfalse{mathjax}%
121        \LWR@subsingledollar*{\textbackslash{}chlewis\{#2\}\{#3\}}%
122        {
123            \protect\LWR@HTMLsanitizedetokenized{\detokenize\expandafter{#1}}%
124        }{
125            \chemformula_lewis:nnn {#1} {#2} {#3}
126        }
127        \endgroup%
128    }
```

lwarp redefines the $ character, so special handling is required to escape math expressions inside \ch.

This boolean tracks a new kind of escaped math:

```
129 \bool_new:N        \l__chemformula_first_last_LWRdollar_bool
```

\chemformula_input_escape_math

Adds additional escaping for the new dollar definition:

```
130 \cs_gset_protected:Npn \__chemformula_input_escape_math:n #1
131   {
132     \__chemformula_first_last_math:n {#1}
133     \bool_if:NT \l__chemformula_first_last_dollar_bool
134       {
135         \bool_set_true:N \l__chemformula_first_last_math_bool
136         \__chemformula_read_escape_dollar:w #1 \q_nil
137       }
138     \bool_if:NT \l__chemformula_first_last_mathbraces_bool
139       {
140         \bool_set_true:N \l__chemformula_first_last_math_bool
141         \__chemformula_read_escape_mathbraces:w #1 \q_nil
142       }
```

Added by lwarp:

```
143      \bool_if:NT \l__chemformula_first_last_LWRdollar_bool%        lwarp
144        {
145          \bool_set_true:N \l__chemformula_first_last_math_bool%  lwarp
146          \__chemformula_read_escape_LWRdollar:w #1 \q_nil%        lwarp
147        }
148  }
```

\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.

```
149 \begingroup
150 \catcode`\$=\active
151
152 \cs_new_protected:Npn \__chemformula_read_escape_LWRdollar:w $#1$ \q_nil
153 {
154     \__chemformula_read_escape_math:n {#1}
155 }
156
```

157 \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 $.

```
158 \cs_new_protected:Npn \__chemformula_bool_cs_set_if_first_last:NnNN #1#2#3#4
159   {
160     \int_zero:N \l__chemformula_tmpa_int
161     \int_zero:N \l__chemformula_tmpb_int
162     \int_set:Nn \l__chemformula_tmpa_int { \tl_count:n {#2} }
163     \tl_map_inline:nn {#2}
164       {
165         \int_incr:N \l__chemformula_tmpb_int
166         \int_compare:nT { \l__chemformula_tmpb_int = 1 }
167           {
```

At the start, the cs_ version compares control sequences:

```
168             \ifdefstrequal{##1}{#3}% lwarp
169               {
170                   \bool_set_true:N #1
171               }% lwarp
172               {}
173           }
```

At the end, compare more control sequences:

```
174         \int_compare:nT { \l__chemformula_tmpb_int = \l__chemformula_tmpa_int }
175           {
176             \ifdefstrequal{##1}{#4}
177               {}
178               {
179                   \bool_set_false:N #1
180               }
181           }
182       }
183   }
```

\chemformula_first_last_math

Modified to check for the new meaning of $ at first/last:

```
184 \cs_gset_protected:Npn \__chemformula_first_last_math:n #1
185   {
186     \bool_set_false:N \l__chemformula_first_last_math_bool
187     \bool_set_false:N \l__chemformula_first_last_dollar_bool
188     \bool_set_false:N \l__chemformula_first_last_LWRdollar_bool%    lwarp
189     \bool_set_false:N \l__chemformula_first_last_mathbraces_bool
190     \__chemformula_bool_set_if_first_last:Nnnn
191       \l__chemformula_first_last_dollar_bool
192       {#1}
193       { $ } { $ }
194     \bool_if:NF \l__chemformula_first_last_dollar_bool
195       {
196         \__chemformula_bool_set_if_first_last:Nnnn
197           \l__chemformula_first_last_mathbraces_bool
198           {#1}
```

```
199        { \( } { \) }
```

Added by lwarp:

```
200        \bool_if:NF \l__chemformula_first_last_mathbraces_bool%    lwarp
201          {
202              \__chemformula_bool_cs_set_if_first_last:NnNN
203              \l__chemformula_first_last_LWRdollar_bool
204              {#1}
205              { \LWR@newsingledollar } { \LWR@newsingledollar }
206          }% lwarp
207      }
208  }

209 \ExplSyntaxOff
```

File 83 **`lwarp-chemgreek.sty`**

§ 192   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.

⚠ XƎLATEX, LuaLATEX   If using XƎLATEX or LuaLATEX, 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 84 **`lwarp-chemmacros.sty`**

§ 193   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.

§ 193.1 **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}
```

§ 193.2 **Code**

```
2 \ExplSyntaxOn
```

§ 193.3 **Loading packages**

Also accept the lwarp version:

```
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  }
```

§ 193.4 **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.

§ 193.5 **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 90.

```
20 \DeclareDocumentEnvironment{redoxreaction}{m m}
21 {\begin{lateximage}[-chemmacros- redoxreaction]}
22 {\end{lateximage}}
```

§ 193.6 **Acid-base**

```
23 \AtBeginDocument{
24 \chemmacros_module_if_loaded:nTF{{acid-base}}{
25 \PackageInfo{lwarp}{Patching~chemmacros~module~acid-base}
26
27 \cs_gset_protected:Npn \chemmacros_p:n #1
28   {
29     \begingroup
30     \boolfalse{mathjax}
31     \LWR@subsingledollar*{
32         \textbackslash{}%
33         p%
34         \{%
35             \LWR@HTMLsanitizedetokenized{\detokenize{#1}}%
36         \}
37     }{
38         chemmacrosp%
39         \protect\LWR@HTMLsanitizedetokenized{\detokenize\expandafter{#1}}%
40     }{
41     \group_begin:
42       \mbox
43         {
44           \chemmacros_p_style:n {p}
45           \ensuremath {#1}
46         }
47     \group_end:
48     }
49     \endgroup
50   }
51
52 \RenewDocumentCommand \pH  {} {
```

```
53      \begingroup
54      \boolfalse{mathjax}
55      \LWR@subsingledollar*{\textbackslash{}pH}{chemmacros}{
56          \chemmacros_p:n { \chemmacros_formula:n {H} }
57      }
58      \endgroup
59 }
60
61 \RenewDocumentCommand \pOH {} {
62      \begingroup
63      \boolfalse{mathjax}
64      \LWR@subsingledollar*{\textbackslash{}pOH}{chemmacros}{
65          \chemmacros_p:n { \chemmacros_formula:n {OH} }
66      }
67      \endgroup
68 }
69
70 \RenewDocumentCommand \pKa {O{}}
71  {
72      \begingroup
73      \boolfalse{mathjax}
74      \LWR@subsingledollar*{\textbackslash{}pKa{[}#1{]}}{chemmacros #1}{
75          \chemmacros_p:n
76          {
77              \Ka \ifblank {#1} {}
78              { {} \c_math_subscript_token { \chemmacros_bold:n {#1} } }
79          }
80      }
81      \endgroup
82  }
83
84 \RenewDocumentCommand \pKb {O{}}
85  {
86      \begingroup
87      \boolfalse{mathjax}
88      \LWR@subsingledollar*{\textbackslash{}pKb{[}#1{]}}{chemmacros #1}{
89          \chemmacros_p:n
90          {
91              \Kb \ifblank {#1} {}
92              { {} \c_math_subscript_token { \chemmacros_bold:n {#1} } }
93          }
94      }
95      \endgroup
96  }
97
98 \LetLtxMacro\LWR@chemmacros@origKa\Ka
99 \renewcommand*{\Ka}{%
100      \begingroup
101      \boolfalse{mathjax}
102      \LWR@subsingledollar*{\textbackslash{}Ka}{chemmacros}{%
103          \LWR@chemmacros@origKa%
104      }%
105      \endgroup
106 }
107
108 \LetLtxMacro\LWR@chemmacros@origKb\Kb
109 \renewcommand*{\Kb}{%
110      \begingroup
111      \boolfalse{mathjax}
112      \LWR@subsingledollar*{\textbackslash{}Kb}{chemmacros}{%
```

```
113        \LWR@chemmacros@origKb%
114      }%
115      \endgroup
116 }
117
118 \LetLtxMacro\LWR@chemmacros@origKw\Kw
119 \renewcommand*{\Kw}{%
120      \begingroup
121      \boolfalse{mathjax}
122      \LWR@subsingledollar*{\textbackslash{}Kw}{chemmacros}{
123          \LWR@chemmacros@origKw
124      }
125      \endgroup
126 }
127
128 }{}% module loaded
129 }% AtBeginDocument
```

§ 193.7 Charges

```
130 \AtBeginDocument{
131 \chemmacros_module_if_loaded:nTF{{charges}}{
132 \PackageInfo{lwarp}{Patching~chemmacros~module~charges}
133
134 \cs_gset_protected:Npn \fplus  {
135      \begingroup
136      \boolfalse{mathjax}
137      \LWR@subsingledollar*{\textbackslash{}fplus}{chemmacros}
138      { \LWR@origensuredmath{\chemformula_fplus:}  }
139      \endgroup
140 }
141 \cs_gset_protected:Npn \fminus {
142      \begingroup
143      \boolfalse{mathjax}
144      \LWR@subsingledollar*{\textbackslash{}fminus}{chemmacros}
145      { \LWR@origensuredmath{\chemformula_fminus:} }
146      \endgroup
147 }
148
149 }{}% Module loaded.
150 }% AtBeginDocument
```

§ 193.8 Nomenclature

```
151 \AtBeginDocument{
152 \chemmacros_module_if_loaded:nTF{{nomenclature}}{
153 \PackageInfo{lwarp}{Patching~chemmacros~module~nomenclature}
154
155 \cs_gset_protected:Npn \chemmacros_charge:n #1
156   {
157      \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}
158      {\chemmacros_formula:n { {}^{#1} }}
159      {
160          \ifmmode
161              {\chemmacros_formula:n { {}^{#1} }}
162          \else
163              { \textsuperscript{\ensuremath{#1}} }
164          \fi
165      }
166   }
```

```
167
168 \hook_gput_code:nnn {begindocument/end} {chemmacros}
169 {
170 \protected\def\LWR@HTML@chemprime { \HTMLunicode{2032} }
171 \LWR@formatted{chemprime}
172 }

173 \cs_gset_protected:Npn \__chemmacros_cip:n #1
174   {
175     \tl_set:Nn \l__chemmacros_tmpa_tl {#1}
176     \int_step_inline:nnnn {0} {1} {9}
177       {
178         \tl_replace_all:Nnn \l__chemmacros_tmpa_tl
179           {##1}
180           { { \l__chemmacros_cip_number_tl ##1} }
181       }
182     {
183       \l__chemmacros_cip_inner_tl
184       \LWR@textcurrentcolor{\LWR@textcurrentfont{% lwarp
185         \l__chemmacros_tmpa_tl
186       }}% lwarp
187     }
188   }

189 \RenewDocumentCommand \Sconf { O{S} } {
190 \begin{lateximage}[\textbackslash{}Sconf{[}#1{]}]*
191   \chemmacros_sconf:n {#1}
192 \end{lateximage}
193 }
194
195 \RenewDocumentCommand \Rconf { O{R} } {
196 \begin{lateximage}[\textbackslash{}Rconf{[}#1{]}]*
197   \chemmacros_rconf:n {#1}
198 \end{lateximage}
199 }

200 \cs_gset_protected:Npn \chemmacros_hapto:n #1
201   {
202     \begingroup
203     \boolfalse{mathjax}
204     \LWR@subsingledollar*{\textbackslash{}hapto\{#1\}}{chemmacros}{
205       \chemmacros_coordination_symbol:nnnn
206       { \l__chemmacros_coord_use_hyphen_bool }
207       {
208         { \c_true_bool }
209       }
210       { \chemeta }
211       {#1}
212     }
213     \endgroup
214   }
215
216 \cs_gset_protected:Npn \chemmacros_dento:n #1
217   {
218     \begingroup
219     \boolfalse{mathjax}
220     \LWR@subsingledollar*{\textbackslash{}dento\{#1\}}{chemmacros}{
221       \chemmacros_coordination_symbol:nnnn
222       { \l__chemmacros_coord_use_hyphen_bool }
223       {
```

```
224           { \c_true_bool }
225         }
226       { \chemkappa }
227       {#1}
228     }
229     \endgroup
230   }
231
232 \cs_gset_protected:Npn \chemmacros_bridge:n #1
233   {
234     \begingroup
235     \boolfalse{mathjax}
236     \LWR@subsingledollar*{\textbackslash{}bridge\{#1\}}{chemmacros}{
237       \chemmacros_coordination_symbol:nnnn
238       { \l__chemmacros_coord_use_hyphen_bool }
239       { \l__chemmacros_bridge_super_bool }
240       { \chemmu }
241       {#1}
242     }
243     \endgroup
244   }
245 }{}% Module loaded.
246 }% AtBeginDocument
```

§ 193.9 **Particles**

```
247 \AtBeginDocument{
248 \chemmacros_module_if_loaded:nTF{{particles}}{
249 \PackageInfo{lwarp}{Patching~chemmacros~module~particles}
250
251 \cs_gset_protected:Npn \chemmacros_declare_nucleophile:Nn #1#2
252   {
253     \cs_set_protected:cpn {__chemmacros_ \chemmacros_remove_backslash:N #1:}
254       {
255         \bool_if:NTF \l__chemmacros_nucleophile_elpair_bool
256           {
257             \chemmacros_elpair:n { #2 }
258               { \skip_horizontal:N \l__chemmacros_nucleophile_dim }
259             \chemmacros_formula:n { {}^{-} }
260           }
261           { \chemmacros_formula:n { #2^{-} } } }
262     }
263     \DeclareDocumentCommand #1 {o}
264       {%
265         \begin{lateximage}%
266         \group_begin:%
267           \IfNoValueF {##1}%
268             { \chemmacros_set_keys:nn {particles} {##1} }%
269           \use:c {__chemmacros_ \chemmacros_remove_backslash:N #1:}%
270         \group_end:%
271         \end{lateximage}%
272       }
273   }
274
275 \RenewChemNucleophile \Nuc {Nu}
276 \RenewChemNucleophile \ba  {ba}
277
278 }{}% Module loaded.
279 }% AtBeginDocument
```

§ 193.10 **Phases**

```
280 \AtBeginDocument{
281 \chemmacros_module_if_loaded:nTF{{phases}}{
282 \PackageInfo{lwarp}{Patching~chemmacros~module~phases}
283
284 \cs_undefine:N \chemmacros_phase:n
285 \cs_new_protected:Npn \chemmacros_phase:n #1
286   {

287     \mode_leave_vertical:

288     \bool_if:NTF \l__chemmacros_phases_sub_bool
289       {
290         \ifnumequal{\value{LWR@lateximagedepth}}{0}
291         {
292             \textsubscript{ (#1) }
293         }
294         {
295             \chemformula_subscript:n { (#1) }
296         }
297       }
298       {
299         \skip_horizontal:N \l__chemmacros_phases_space_dim
300         \chemmacros_text:n { (#1) }
301       }
302   }
303
304 }{}% Module loaded.
305 }% AtBeginDocument
```

§ 193.11 **Mechanisms**

```
306 \AtBeginDocument{
307 \chemmacros_module_if_loaded:nTF{{mechanisms}}{
308 \PackageInfo{lwarp}{Patching~chemmacros~module~mechanisms}
309
310 \chemmacros_define_keys:nn {textmechanisms}
311   {
312     type       .choice: ,
313     type /     .code:n    =
314       {
315         \__chemmacros_set_mechanisms:nnn { S }
316         {
317             \textsubscript{N}
318         }
319         { }
320       } ,
321     type / 1  .code:n     =
322       {
323         \__chemmacros_set_mechanisms:nnn { S }
324         {
325             \textsubscript{N}
326           1
327         }
328         { }
329       } ,
330     type / 2  .code:n     =
331       {
332         \__chemmacros_set_mechanisms:nnn { S }
```

```
333        {
334            \textsubscript{N}
335          2
336        }
337        { }
338      } ,
339    type / se .code:n    =
340      {
341        \__chemmacros_set_mechanisms:nnn { S }
342        {
343            \textsubscript{E}
344        }
345        { }
346      } ,
347    type / 1e .code:n    =
348      {
349        \__chemmacros_set_mechanisms:nnn { S }
350        {
351            \textsubscript{E}
352          1
353        }
354        { }
355      } ,
356    type / 2e .code:n    =
357      {
358        \__chemmacros_set_mechanisms:nnn { S }
359        {
360            \textsubscript{E}
361          2
362        }
363        { }
364      } ,
365    type / ar .code:n    =
366      {
367        \__chemmacros_set_mechanisms:nnn { S }
368        {
369            \textsubscript{E}
370        }
371        { Ar - }
372      } ,
373    type / e  .code:n    =
374      { \__chemmacros_set_mechanisms:nnn { E } { } { } } ,
375    type / e1 .code:n    =
376      { \__chemmacros_set_mechanisms:nnn { E } { 1 } { } } ,
377    type / e2 .code:n    =
378      { \__chemmacros_set_mechanisms:nnn { E } { 2 } { } } ,
379    type / cb .code:n    =
380      {
381        \__chemmacros_set_mechanisms:nnn { E }
382        {
383          1
384            \textsubscript{cb}
385        }
386        { }
387      } ,
388    type      .default:n =
389  }
390
391 \cs_gset_protected:Npn \chemmacros_mechanisms:n #1
392  {
```

```
393    \tl_if_blank:nTF {#1}
394      { \chemmacros_set_keys:nn {textmechanisms} { type } }
395      { \chemmacros_set_keys:nn {textmechanisms} { type = #1 } }
396    \mbox
397      {
398        \tl_use:N \l__chemmacros_mechanisms_ar_tl
399        \tl_use:N \l__chemmacros_mechanisms_type_tl
400        \tl_use:N \l__chemmacros_mechanisms_mol_tl
401      }
402  }
403
404 \appto\LWR@restoreorigformatting{%
405 \cs_set_protected:Npn \chemmacros_mechanisms:n #1%
406  {%
407    \tl_if_blank:nTF {#1}%
408      { \chemmacros_set_keys:nn {mechanisms} { type } }%
409      { \chemmacros_set_keys:nn {mechanisms} { type = #1 } }%
410    \mbox%
411      {%
412        \tl_use:N \l__chemmacros_mechanisms_ar_tl%
413        \tl_use:N \l__chemmacros_mechanisms_type_tl%
414        \tl_use:N \l__chemmacros_mechanisms_mol_tl%
415      }%
416  }%
417 }
418
419 }{}% Module loaded.
420 }% AtBeginDocument
```

§ 193.12 **Newman**

There are so many options that it is hard to hash these images for reuse.

```
421 \AtBeginDocument{
422 \chemmacros_module_if_loaded:nTF{{newman}}{
423 \PackageInfo{lwarp}{Patching~chemmacros~module~newman}
424
425 \RenewDocumentCommand \newman {od()m}%
426  {
427    \IfValueTF{#2}
428    {\begin{lateximage}[\textbackslash{}newman(#2)\{#3\}]*}
429    {\begin{lateximage}[\textbackslash{}newman\{#3\}]*}
430    \group_begin:
431      \IfNoValueF  {#1} { \chemmacros_set_keys:nn {newman} {#1} }
432      \IfNoValueTF {#2}
433        { \chemmacros_newman:nn {   } {#3} }
434        { \chemmacros_newman:nn {#2} {#3} }
435    \group_end:
436    \end{lateximage}
437  }%
438
439 }{}% Module loaded.
440 }% AtBeginDocument
```

§ 193.13 **Orbital**

```
441 \AtBeginDocument{
442 \chemmacros_module_if_loaded:nTF{{orbital}}{
443 \PackageInfo{lwarp}{Patching~chemmacros~module~orbital}
444
445 \RenewDocumentCommand \orbital {om}
446  {
447    \IfValueTF{#1}
448    {
449        \begin{lateximage}[%
450            \textbackslash{}%
451            orbital{[}%
452            \LWR@HTMLsanitizedetokenized{\detokenize{#1}}%
453            {]}\{#2\}%
454        ]*[][margin-left: 1em ; margin-right: 1em]
455    }
456    {
457        \begin{lateximage}[%
458            \textbackslash{}orbital\{#2\}%
459        ]*[][margin-left: 1em ; margin-right: 1em]
460    }
461    \group_begin:
462      \chemmacros_set_keys:nn {orbital/type} {#2}
463      \IfNoValueTF {#1}
464        { \chemmacros_orbital:n {   } }
465        { \chemmacros_orbital:n {#1} }
466    \group_end:
467    \end{lateximage}
468  }
469
470 }{}% Module loaded.
471 }% AtBeginDocument
```

§ 193.14 **Reactions**

\chemmacros_declare_reaction_env   {⟨*chem*⟩} {⟨*math*⟩} {⟨*args number*⟩} {⟨*argument list ({#2}{#3}...)*⟩}

```
472 \AtBeginDocument{
473 \chemmacros_module_if_loaded:nTF{{reactions}}{
474 \PackageInfo{lwarp}{Patching~chemmacros~module~reactions}
475
476 % #1: chem
477 % #2: math
478 % #3: args number
479 % #4: argument list ({#2}{#3}...)
480 \cs_gset_protected:Npn \__chemmacros_declare_reaction_env:nnnn #1#2#3#4
481  {
482    \exp_args:Nnx \DeclareDocumentEnvironment {#1}
483      { \int_compare:nT { #3+0 = 0 } {!} O{} \prg_replicate:nn {#3+0} {m} }
484      {
485        \boolfalse{mathjax}%                                lwarp
486        \ifdefvoid{\LWR@ThisAltText}{%                      lwarp
487            \ThisAltText{-chemmacros-~reaction}%            lwarp
488        }{}%                                                lwarp
489        \chemmacros_add_reaction_description:n {##1}
490        \__chemmacros_begin_reaction:
491        \__chemmacros_reaction_read:nnw {#2} {#4}
492      }
493      {
```

```
494        \__chemmacros_end_reaction:
495        \gdef\LWR@ThisAltText{}%                              lwarp
496        \ignorespacesafterend
497      }
498  }
499
500 \cs_generate_variant:Nn \chemmacros_declare_reaction_env:nnnn {nnnV}
501
502 \RenewChemReaction {reaction}   {equation}
503 \RenewChemReaction {reaction*}  {equation*}
504 \RenewChemReaction {reactions}  {align}
505 \RenewChemReaction {reactions*} {align*}
506
507 }{}% Module loaded.
508 }% AtBeginDocument
```

§ 193.15 **Reactants**

Recompiled for tabular ampersand processing, with the only change being \StartDefiningTabulars. \xpatchcmd does not work here.

```
509 \StartDefiningTabulars%      lwarp
510
511 % #1: star: include ID in table
512 \RenewDocumentCommand \printreactants {s}
513  {
514    \group_begin:
515      \chemmacros_set_keys:nn {reactants} { switch = false }
516      \int_step_variable:nNn
517        { \seq_count:N \g_chemnum_initiated_compounds_seq }
518        \l__chemmacros_reactants_tmpa_tl
519        {
520          \seq_put_right:Nx
521            \l__chemmacros_reactants_tmpa_seq
522            {
523              \chemnum_cmpd:nnne { \c_false_bool } { \c_true_bool } {}
524                {
525                  \seq_item:NV
526                    \g_chemnum_initiated_compounds_seq
527                    \l__chemmacros_reactants_tmpa_tl
528                }
529              &
530              \bool_if:nT {#1}
531                {
532                  \seq_item:NV
533                    \g_chemnum_initiated_compounds_seq
534                    \l__chemmacros_reactants_tmpa_tl
535                  &
536                }
537              % TODO: expl3-command ??
538              \solvent
539                {
540                  \seq_item:NV
541                    \g_chemnum_initiated_compounds_seq
542                    \l__chemmacros_reactants_tmpa_tl
543                }
544              \tabularnewline
545            }
546          \tl_set:Nx
547            \l__chemmacros_reactants_tmpb_tl
```

```
548                    {
549                      \seq_item:NV
550                        \g_chemnum_initiated_compounds_seq
551                        \l__chemmacros_reactants_tmpa_tl
552                    }
553              \chemmacros_reactants_list_subreactant:Vn
554                \l__chemmacros_reactants_tmpb_tl
555                {#1}
556          }
557      % TODO: longtable ?
558      %        table customizable?
559      % first draft of two styles
560      \par
561      \noindent
562      \bool_if:NTF \l__chemmacros_reactants_printreactants_style_bool
563        {
564          \str_case:Vn \l__chemmacros_reactants_printreactants_style_str
565            {
566              {xltabular}
567              {
568                \chemmacros_if_package_loaded:nTF {xltabular}
569                  {
570                    \bool_if:nTF {#1}
571                      {
572                        \begin {xltabular}
573                          { \textwidth }
574                          { @{}ll>{\raggedright\arraybackslash}X@{} }
575                      }
576                      {
577                        \begin {xltabular}
578                          { \textwidth }
579                          { @{}l>{\raggedright\arraybackslash}X@{} }
580                      }
581                    \seq_use:Nn \l__chemmacros_reactants_tmpa_seq { }
582                    \end{xltabular}
583                  }
584                  {
585                    \msg_expandable_error:nnnn
586                      {chemmacros}
587                      {package-not-loaded}
588                      { \printreactants }
589                      {xltabular}
590                  }
591              }
592              {longtable}
593              {
594                \chemmacros_if_package_loaded:nTF {longtable}
595                  {
596                    \bool_if:nTF {#1}
597                      {
598                        \begin {longtable}[l]
599                    { @{}ll>{\raggedright\arraybackslash}p{0.6\textwidth}@{} }
600                      }
601                      {
602                        \begin {longtable}[l]
603                    { @{}l>{\raggedright\arraybackslash}p{0.9\textwidth}@{} }
604                      }
605                    \seq_use:Nn \l__chemmacros_reactants_tmpa_seq { }
606                    \end{longtable}
607                  }
```

```
608                        {
609                          \msg_expandable_error:nnnn
610                            {chemmacros}
611                            {package-not-loaded}
612                            { \printreactants }
613                            {longtable}
614                        }
615                    }
616                }
617        }
618          {
619              \msg_warning:nn {chemmacros} {missing-printreactants-style}
620          }
621      \group_end:
622  }
623
624 % #1: full ID
625 % #2: star, include ID in table
626 \cs_gset_protected:Npn \chemmacros_reactants_list_subreactant:nn #1#2
627  {
628      \chemnum_if_subcompounds:nT {#1}
629        {
630          \int_step_variable:nNn
631            { \chemnum_count_subcompounds:n {#1} }
632          \l__chemmacros_reactants_tmpa_tl
633          {
634              \seq_put_right:Nx
635              \l__chemmacros_reactants_tmpa_seq
636                {
637                  \chemnum_cmpd:nnne { \c_false_bool } { \c_true_bool } {}
638                    {
639                      \exp_not:n {#1}
640                      \exp_not:V \l_chemnum_compound_separator_tl
641                      \chemnum_get_subcompound:nV
642                        {#1}
643                        \l__chemmacros_reactants_tmpa_tl
644                    }
645                  &
646                  \bool_if:nT {#2}
647                    {
648                      #1
649                      \l_chemnum_compound_separator_tl
650                      \chemnum_get_subcompound:nV
651                        {#1}
652                        \l__chemmacros_reactants_tmpa_tl
653                      &
654                    }
655                  % TODO: expl3-command ??
656                  \solvent
657                    {
658                      #1
659                      \l_chemnum_compound_separator_tl
660                      \chemnum_get_subcompound:nV
661                        {#1}
662                        \l__chemmacros_reactants_tmpa_tl
663                    }
664                  \tabularnewline
665                }
666          }
667        }
```

```
668    }
669  \cs_generate_variant:Nn \chemmacros_reactants_list_subreactant:nn {V}
670
671  \StopDefiningTabulars%         lwarp
```

§ 193.16 **Redox**

```
672  \AtBeginDocument{
673  \chemmacros_module_if_loaded:nTF{{redox}}{
674  \PackageInfo{lwarp}{Patching~chemmacros~module~redox}
675
676  \NewDocumentCommand \LWR@chemmacros@ox { s m >{\SplitArgument{1}{,}}m }
677    {
678        \IfBooleanTF {#1}
679          { \chemmacros_ox:nnnn {#1} {#2} #3 }
680          { \chemmacros_ox:nnnn {  } {#2} #3 }
681    }
682
683  \RenewDocumentCommand \ox { s O{} m }
684    {
685        \begingroup
686        \boolfalse{mathjax}
687        \IfBooleanTF {#1}
688          {
689            \LWR@subsingledollar*{% yes hash
690                \textbackslash{}%
691                ox*%
692                \{%
693                    \LWR@HTMLsanitizedetokenized{\detokenize{#3}}%
694                \}% alt
695            }{%
696            star \protect\LWR@HTMLsanitizedetokenized{\detokenize\expandafter{#2}}%
697            }{%
698                \LWR@chemmacros@ox* {#2} {#3}% contents
699            }%
700          }
701          {
702            \LWR@subsingledollar*{% yes hash
703                \textbackslash{}%
704                ox%
705                \{%
706                    \LWR@HTMLsanitizedetokenized{\detokenize{#3}}%
707                \}% alt
708            }{%
709              \protect\LWR@HTMLsanitizedetokenized{\detokenize\expandafter{#2}}%
710            }{%
711                \LWR@chemmacros@ox {#2} {#3}% contents
712            }%
713          }
714        \endgroup
715    }
716
717  }{}% Module loaded.
718  }% AtBeginDocument
```

§ 193.17 **Scheme**

Fix for chemmacros as of v5.8b, when using newfloat and babel:

```
719 \AtBeginDocument{
720 \chemmacros_module_if_loaded:nTF{{scheme}}{
721 \PackageInfo{lwarp}{Patching~chemmacros~module~scheme}
722
723 \ifdefstring{\schemename}{los}{
724 \SetupFloatingEnvironment{scheme}{
725 name = \chemmacros_translate:n {scheme-name}
726 }
727 }{}
728
729 }{}% Module loaded.
730 }% AtBeginDocument
```

§ 193.18 **Spectroscopy**

```
731 \AtBeginDocument{
732 \chemmacros_module_if_loaded:nTF{{spectroscopy}}{
733 \PackageInfo{lwarp}{Patching~chemmacros~module~spectroscopy}
734
735 \cs_gset_protected:Npn \__chemmacros_nmr_base:nn #1#2
736   {
737     \group_begin:
738       \tl_use:N \l__chemmacros_nmr_base_format_tl
739       \tl_if_blank:VF \g__chemmacros_nmr_element_coupled_tl
740         {
741           \tl_put_left:Nn \g__chemmacros_nmr_element_coupled_tl { \{ }
742           \tl_put_right:Nn \g__chemmacros_nmr_element_coupled_tl { \} }
743         }
744       \tl_put_left:Nn \g__chemmacros_nmr_element_coupled_tl {#2}
745 %       \chemmacros_formula:n { ^{#1} }
746       \textsuperscript{#1}
747       \tl_if_blank:VF \g__chemmacros_nmr_element_coupled_tl
748         {
749           \bool_if:NTF \l__chemmacros_nmr_parse_bool
750             { \chemformula_ch:nV {} \g__chemmacros_nmr_element_coupled_tl }
751             { \chemmacros_formula:V \g__chemmacros_nmr_element_coupled_tl }
752         }
753       \tl_use:N \l__chemmacros_nmr_element_method_connector_tl
754       \tl_use:N \l__chemmacros_nmr_method_tl
755     \group_end:
756   }
757
758
759 \cs_gset_protected:Npn \chemmacros_nmr_position:n #1
760   {
761     \chemmacros_formula:x
762       {
763         \exp_not:V \g__chemmacros_nmr_element_tl
764         \bool_if:NF \l__chemmacros_nmr_position_side_bool
765           {
766             \tl_if_eq:NnTF \l__chemmacros_nmr_position_tl {^}% lwarp
767             { \textsuperscript{\exp_not:n { {#1} }} }% lwarp
768             { \textsubscript{\exp_not:n { {#1} }} }% lwarp
769 %             \exp_not:V \l__chemmacros_nmr_position_tl
770 %             \exp_not:n { {#1} }
771           }
772       }
773     \bool_if:NT \l__chemmacros_nmr_position_side_bool
774       {
775         \tl_use:N \l__chemmacros_nmr_position_tl
```

```
776          \__chemmacros_nmr_position:n {#1}
777        }
778    }
779
780 \cs_gset_protected:Npn \__chemmacros_nmr_coupling:w (#1;#2)
781    {
782      \tl_set:Nn \l__chemmacros_nmr_coupling_bonds_tl
783        {
784          \l__chemmacros_nmr_coupling_bonds_pre_tl
785          #1
786          \l__chemmacros_nmr_coupling_bonds_post_tl
787        }
788      \bool_if:NTF \l__chemmacros_nmr_coupling_nuclei_sub_bool
789        {
790          \tl_set:Nn \l__chemmacros_nmr_coupling_nuclei_tl
791            {
792 %              \c_math_subscript_token
793            \textsubscript% lwarp
794              {
795                \l__chemmacros_nmr_coupling_nuclei_pre_tl
796                \chemmacros_formula:n {#2}
797                \l__chemmacros_nmr_coupling_nuclei_post_tl
798              }
799          }
800        }
801        {
802          \tl_set:Nn \l__chemmacros_nmr_coupling_nuclei_tl
803            {
804              \l__chemmacros_nmr_coupling_nuclei_pre_tl
805              \chemmacros_formula:n {#2}
806              \l__chemmacros_nmr_coupling_nuclei_post_tl
807            }
808        }
809      \__chemmacros_nmr_coupling_aux_i:w
810    }
811 \AfterEndPreamble{% After \AtBeginDocument
812 % \NMR{<num>,<elem>}(<num>,<unit>)[<solvent>] ALL arguments are optional
813 % \NMR* same but without ": $\delta$" at end
814 \cs_gset_protected:Npn \chemmacros_nmr:nnnn #1#2#3#4
815    {
816    \bool_if:NT \l__chemmacros_nmr_list_bool { \item \scan_stop: }
817    \group_begin:

818        \mode_leave_vertical:

819        \bool_set_false:N \l__chemmacros_nmr_frequency_bool
820        \bool_set_false:N \l__chemmacros_nmr_solvent_bool
821        \tl_if_empty:nF {#3}
822        { \bool_set_true:N \l__chemmacros_nmr_frequency_bool }
823        \tl_if_empty:nF {#4}
824        { \bool_set_true:N \l__chemmacros_nmr_solvent_bool }
825        \bool_if:nT
826        {
827            \l__chemmacros_nmr_frequency_bool
828            ||
829            \l__chemmacros_nmr_solvent_bool
830        }
831        { \bool_set_true:N \l__chemmacros_nmr_delimiters_bool }
832        \bool_if:nT
833        {
```

```
834                  \l__chemmacros_nmr_frequency_bool
835                  &&
836                  \l__chemmacros_nmr_solvent_bool
837              }
838          { \bool_set_true:N \l__chemmacros_nmr_comma_bool }
839          \tl_if_empty:nTF {#2}
840          {
841                  \__chemmacros_nmr_nucleus:VV
842                  \l__chemmacros_nmr_isotope_default_tl
843                  \l__chemmacros_nmr_element_default_tl
844          }
845          { \__chemmacros_nmr_nucleus:w #2 \q_stop }
846          \mode_if_math:TF
847          {
848              \text
849              {
850                  \group_begin:
851                  \tl_use:N \l__chemmacros_nmr_format_tl
852 \LWR@textcurrentcolor{\LWR@textcurrentfont{% lwarp
853                  \__chemmacros_nmr_base:VV
854                      \g__chemmacros_nmr_isotope_tl
855                      \g__chemmacros_nmr_element_tl
856                  \bool_if:NT \l__chemmacros_nmr_delimiters_bool
857                      { ~ ( }
858                  \bool_if:NT \l__chemmacros_nmr_frequency_bool
859                      { \__chemmacros_nmr_frequency:n {#3} }
860                  \bool_if:NT \l__chemmacros_nmr_comma_bool
861                      { , ~ }
862                  \bool_if:NT \l__chemmacros_nmr_solvent_bool
863                      { \chemmacros_formula:n {#4} }
864                  \bool_if:NT \l__chemmacros_nmr_delimiters_bool
865                      { ) }
866                  \tl_if_blank:nT {#1} {:~}
867 }}% lwarp
868                  \group_end:
869              }
870              \tl_if_blank:nT {#1}
871              {
872                  \delta
873                  \text { \l__chemmacros_nmr_delta_tl }
874                  \bool_if:NT \l__chemmacros_nmr_use_equal_bool {=}
875              }
876          }
877          {
878              \group_begin:
879              \tl_use:N \l__chemmacros_nmr_format_tl
880 \LWR@textcurrentcolor{\LWR@textcurrentfont{% lwarp
881              \__chemmacros_nmr_base:VV
882                  \g__chemmacros_nmr_isotope_tl
883                  \g__chemmacros_nmr_element_tl
884              \bool_if:NT \l__chemmacros_nmr_delimiters_bool
885                  {~(}
886              \bool_if:NT \l__chemmacros_nmr_frequency_bool
887                  { \__chemmacros_nmr_frequency:n {#3} }
888              \bool_if:NT \l__chemmacros_nmr_comma_bool
889                  {,~}
890              \bool_if:NT \l__chemmacros_nmr_solvent_bool
891                  {
892                  \bool_if:NTF \l__chemmacros_nmr_parse_bool
```

```
893 %                    { \chemformula_ch:nn { } {#4} }%  original
894                   {\ch{#4}}%  lwarp
895                   {#4}
896                 }
897             \bool_if:NT \l__chemmacros_nmr_delimiters_bool
898               ()}
899 }}% lwarp
900             \tl_if_blank:nT {#1} {:}
901             \group_end:
902             \tl_if_blank:nT {#1}
903             {
904                 \tl_use:N \c_space_tl
905                 \c_math_toggle_token
906                 \delta
907                 \c_math_toggle_token
908                 \l__chemmacros_nmr_delta_tl
909                 \bool_if:NT \l__chemmacros_nmr_use_equal_bool {~=}
910             }
911           }
912     \group_end:
913     }
914 }% AfterEndPremble
915
916
917 \RenewDocumentCommand \chemmacros_data:w { smo }
918   {
919     \bool_if:NT \l__chemmacros_nmr_list_bool { \item }
920       {
921 %          \tl_use:N \l__chemmacros_nmr_format_tl #2
922         \tl_use:N \l__chemmacros_nmr_format_tl
923         \LWR@textcurrentcolor{\LWR@textcurrentfont{% lwarp
924             #2
925             \IfNoValueF {#3} { ~ ( #3 ) }
926         \IfBooleanT {#1} { \bool_if:NT \l__chemmacros_nmr_use_equal_bool { : } }
927         }}% lwarp
928       }
929     \IfBooleanF {#1} { \bool_if:NT \l__chemmacros_nmr_use_equal_bool { ~ = } }
930   }
931
932 }{}% Module loaded.
933 }% AtBeginDocument
```

§ 193.19   **Thermodynamics**

```
934 \AtBeginDocument{
935 \chemmacros_module_if_loaded:nTF{{thermodynamics}}{
936 \PackageInfo{lwarp}{Patching~chemmacros~module~thermodynamics}
937
938 \cs_gset_protected:Npn \chemmacros_state:nnnnnn #1#2#3#4#5#6
939   {
940     \group_begin:
941       \chemmacros_set_keys:ne {thermodynamics}
942         {
943           \exp_not:n {#1} ,
944           \tl_if_novalue:nF {#2} { subscript-left = \exp_not:n {#2} , }
945           \tl_if_novalue:nF {#3} { superscript-left = \exp_not:n {#3} , }
946           \tl_if_novalue:nF {#5} { subscript-right = \exp_not:n {#5} , }
947           \tl_if_novalue:nF {#6} { superscript-right = \exp_not:n {#6} }
948         }
949       \LWR@subsingledollar*{% yes hashing
```

```
950              \textbackslash{}state%
951              \{\LWR@HTMLsanitizedetokenized{\detokenize{#4}}\}%  alt
952         }{%
953              chemmacros_state% add'l hashing
954              #1% options
955              LSP \tl_use:N \l__chemmacros_state_sp_left_tl% super/subscripts
956              LSB \tl_use:N \l__chemmacros_state_sb_left_tl
957              RSP \tl_use:N \l__chemmacros_state_sp_right_tl
958              RSB \tl_use:N \l__chemmacros_state_sb_right_tl
959         }
960         {
961              \LWR@origensuredmath
962                  {
963                  \chemmacros_text:V \l__chemmacros_state_pre_tl
964                  \c_math_superscript_token
965                      { \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*.

```
966                  \tl_if_empty:NTF \l__chemmacros_state_sb_left_tl
967                  {}
968                  {
969                      \c_math_subscript_token
970                        { \chemmacros_text:V \l__chemmacros_state_sb_left_tl }
971                  }
972                  #4
973                  \c_math_superscript_token
974                      { \chemmacros_text:V \l__chemmacros_state_sp_right_tl }
975                  \tl_if_empty:NTF \l__chemmacros_state_sb_right_tl
976                  {}
977                  {
978                      \c_math_subscript_token
979                      { \chemmacros_text:V \l__chemmacros_state_sb_right_tl }
980                  }
981                  \chemmacros_text:V \l__chemmacros_state_post_tl
982                  }
983          }
984     \group_end:
985  }
986 \cs_generate_variant:Nn \chemmacros_state:nnnnnn { nVVVVV }
987
988 \cs_gset_protected:Npn \chemmacros_declare_state:Nn #1#2
989 {
990     \chemmacros_define_keys:xn
991       {thermodynamics/\chemmacros_remove_backslash:N #1}
992       {
993        pre                .meta:nn = {chemmacros/thermodynamics} { pre = ##1 } ,
994        post               .meta:nn = {chemmacros/thermodynamics} { post = ##1 } ,
995       superscript-left  .meta:nn = {chemmacros/thermodynamics} { superscript-left = ##1 } ,
996       superscript-right .meta:nn = {chemmacros/thermodynamics} { superscript-right = ##1 } ,
997         superscript       .meta:n  = { superscript-right = ##1 } ,
998       subscript-left    .meta:nn = {chemmacros/thermodynamics} { subscript-left = ##1 } ,
999       subscript-right   .meta:nn = {chemmacros/thermodynamics} { subscript-right = ##1 } ,
1000        subscript         .meta:n     = { subscript-left = ##1 } ,
1001        subscript-pos     .choices:nn =
1002          { left , right }
1003          { \tl_set_eq:NN \l__chemmacros_state_sb_pos_tl \l_keys_choice_tl } ,
1004        symbol            .tl_set:N = \l__chemmacros_state_symbol_tl ,
1005        unit              .tl_set:N = \l__chemmacros_state_unit_tl
```

```
1006        }
1007    \DeclareDocumentCommand #1 { sO{}D(){}m }
1008      {
1009        \group_begin:
1010          \chemmacros_set_keys:en
1011            {thermodynamics/\chemmacros_remove_backslash:N #1}
1012            {#2}
1013          \tl_if_blank:nF {##3}
1014            {
1015              \chemmacros_set_keys:ne {thermodynamics}
1016              { subscript-\l__chemmacros_state_sb_pos_tl = \exp_not:n {##3} }
1017            }
1018 %        \LWR@origensuredmath
1019 %          {
1020            \chemmacros_state:nVVVVV
1021              {##2}
1022              \c_novalue_tl
1023              \c_novalue_tl
1024              \l__chemmacros_state_symbol_tl
1025              \c_novalue_tl
1026              \c_novalue_tl
1027          \chemmacros_set_keys_groups:nnn {thermodynamics} {variables} {##2}
1028          \IfBooleanF {##1} { = \qty {##4} { \l__chemmacros_state_unit_tl } }
1029 %          }
1030        \group_end:
1031      }
1032  }
```

The pre-existing macros are redefined with the new definition:

```
1033 \RenewChemState \enthalpy { symbol = H , unit = \kilo\joule\per\mole }
1034 \RenewChemState \entropy { symbol = S , unit = \joule\per\kelvin\per\mole , pre = }
1035 \RenewChemState \gibbs    { symbol = G , unit = \kilo\joule\per\mole }
1036
1037 }{}% Module loaded.
1038 }% AtBeginDocument

1039 \ExplSyntaxOff
```

File 85   `lwarp-chemnum.sty`

§ 194   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 \cs_gset_protected:Npn \chemnum_compound_write:n #1
5   {
6     \chemnum_get_compound_property:nn {#1} {pre-main-label-code}
7     \group_begin:
8       \bool_if:NTF \l__chemnum_compound_local_bool
9         { \l__chemnum_local_label_format_tl }
```

```
10        { \chemnum_get_compound_property:nn {#1} {label-format} }
11      {
12        \LWR@textcurrentfont{
13            \chemnum_get_compound_property:nn {#1} {counter-representation}
14        }
15      }
16    \group_end:
17    \chemnum_get_compound_property:nn {#1} {post-main-label-code}
18  }
19
20 \cs_gset_protected:Npn \chemnum_subcompound_write:nn #1#2
21  {
22    \group_begin:
23      \bool_if:NTF \l__chemnum_compound_local_bool
24        { \l__chemnum_local_label_format_tl }
25        { \chemnum_get_compound_property:nn {#1} {label-format} }
26      {
27        \LWR@textcurrentfont{
28            \chemnum_get_subcompound_property:nnn {#1} {#2}
29            {counter-representation}
30        }
31      }
32    \group_end:
33  }
34
35 \ExplSyntaxOff
```

File 86   **`lwarp-chkfloat.sty`**

§ 195   Package   # chkfloat

chkfloat (*Pkg*)   chkfloat is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{chkfloat}[2012/08/19]

File 87   **`lwarp-chngpage.sty`**

§ 196   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 88   **`lwarp-cite.sty`**

§ 197   Package   # cite

(*Emulates or patches code by* DONALD ARSENEAU.)

cite (*Pkg*)   cite is patched for use by lwarp.

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 89   **lwarp-citeref.sty**

§ 198   Package   # citeref

(*Emulates or patches code by* Björn Briel.)

citeref (*Pkg*)   citeref is patched for use by lwarp.

1 \LWR@ProvidesPackagePass{citeref}[1999/27/05]

```
2 \def\@cprwrite#1={%
3     \write\@auxout{\string\citepageref{#1}{\theLWR@previousautopagelabel}}%
4 }
5
6 \def\citepageref#1#2{%
7     \xdef\cpr@testa{\@nameuse{cpr@last@#1}}%letzte Zitatstelle
8     \xdef\cpr@testb{#2}% Seite dieser Zitatstelle
9     \ifx\cpr@testa\cpr@testb%
10        \relax% Konsekutive identische Seitenangaben weglassen
11    \else%
12        \@namexdef{cpr@last@#1}{#2}%
13        \@ifundefined{cpr@#1}%
14        {\@namexdef{cpr@#1}{\LWR@refwithsection{\BaseJobname-autopage-#2}}}%  lwarp
15            {%      lwarp
16                \@namexdef{cpr@#1}{\@nameuse{cpr@#1}, % space
17                \LWR@refwithsection{\BaseJobname-autopage-#2}}%
18            }%
19    \fi
20    }
```

File 90   **lwarp-CJK.sty**

§ 199   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 91 **`lwarp-CJKutf8.sty`**

§ 200   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 92   **`lwarp-classicthesis.sty`**

§ 201   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 93 **`lwarp-cleveref.sty`**

§ 202 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 734 to redefine the message which is printed for page number references.

Table 16 on page 495 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*⟩}

\@templabel becomes 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
108 \IfClassLoadedTF{memoir}{
109 \AtBeginDocument{
110 \def\sf@memsub@label(#1)#2{%
111   \protected@edef\mem@currentlabelname{#1}%
112   \sf@@memsub@label{#2}}
113 }
114 }{}
115
116 \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 94   **`lwarp-clrdblpg.sty`**

§ 203   Package   # clrdblpg

clrdblpg (*Pkg*)   clrdblpg is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{clrdblpg}[2018/04/21]

File 95   **`lwarp-cmbright.sty`**

§ 204   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 96 **`lwarp-cmdtrack.sty`**

## § 205    Package **cmdtrack**

cmdtrack (*Pkg*)    cmdtrack is ignored.

for HTML output:    1 `\LWR@ProvidesPackageDrop{cmdtrack}[2012/12/18]`

2 `\newcommand{\untrack}[1]{}`

---

File 97    **`lwarp-colonequals.sty`**

## § 206    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 98    `lwarp-color.sty`

## § 207    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 99   `lwarp-colortbl.sty`

§ 208   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.

for HTML output: A placeholder definition is forgotten first:

```
1 \let\rowcolor\relax
2
3 \LWR@ProvidesPackagePass{colortbl}[2018/12/12]
```

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}}
```

For MathJax, use the MathJax package. The unused macro options are ignored.

```
41 \begin{warpMathJax}
```

```
42
43 \CustomizeMathJax{\require{colortbl}}
44 \CustomizeMathJax{\let\LWRorigcolumncolor\columncolor}
45 \CustomizeMathJax{\renewcommand{\columncolor}[2][named]{%
46     \LWRorigcolumncolor[#1]{#2}%
47     \LWRabsorbtwooptions%
48 }}
49
50 \CustomizeMathJax{\let\LWRorigrowcolor\rowcolor}
51 \CustomizeMathJax{\renewcommand{\rowcolor}[2][named]{%
52     \LWRorigrowcolor[#1]{#2}%
53     \LWRabsorbtwooptions%
54 }}
55
56 \CustomizeMathJax{\let\LWRorigcellcolor\cellcolor}
57 \CustomizeMathJax{\renewcommand{\cellcolor}[2][named]{%
58     \LWRorigcellcolor[#1]{#2}%
59     \LWRabsorbtwooptions%
60 }}
61
62 \end{warpMathJax}
```

File 100   `lwarp-continue.sty`

§ 209   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 101   `lwarp-copyrightbox.sty`

§ 210   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 102 **`lwarp-crop.sty`**

§ 211   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 103 **`lwarp-ctable.sty`**

§ 212   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 104 **`lwarp-cuted.sty`**

§ 213   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 105   **`lwarp-cutwin.sty`**

§ 214   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 106 **`lwarp-dblfloatfix.sty`**

§ 215   Package **dblfloatfix**

dblfloatfix (*Pkg*)   dblfloatfix is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{dblfloatfix}[2012/12/31]

File 107 **`lwarp-dblfnote.sty`**

§ 216   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 108 **`lwarp-dcolumn.sty`**

§ 217   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 109   **lwarp-decimal.sty**

§ 218   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 110   **lwarp-decorule.sty**

§ 219   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 111   **lwarp-diagbox.sty**

§ 220   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 {⟨*E/W*⟩} {⟨*A*⟩} {⟨*E/W*⟩} {⟨*B*⟩}

```
 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 {⟨*A*⟩} {⟨*B*⟩}

```
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 {⟨*keys*⟩} {⟨*A*⟩} {⟨*B*⟩}

```
26 \def\diagbox@double#1#2#3{%
27 \setkeys{diagbox}{dir=NW,#1}%
28 \@nameuse{LWR@diagbox\diagbox@dir}{#2}{#3}%
29 }
```

\LWR@diagboxTNW {⟨*title*⟩} {⟨*A*⟩} {⟨*B*⟩}

```
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 112   **`lwarp-dingbat.sty`**

§ 221   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 113 **`lwarp-DotArrow.sty`**

§ 222   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 114 **`lwarp-dotlessi.sty`**

§ 223   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 115 **`lwarp-dprogress.sty`**

§ 224 Package **dprogress**

dprogress (*Pkg*) dprogress is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{dprogress}[2008/02/21]

File 116 **`lwarp-draftcopy.sty`**

§ 225 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 117 **`lwarp-draftfigure.sty`**

§ 226 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 118   `lwarp-draftwatermark.sty`

§ 227   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 119   `lwarp-drftcite.sty`

§ 228   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 \def\@lbibitem[#1]#2{\global\@HighCite\z@
3   \item[
4     \textsuperscript{\@nameuse{DCN@#2\@extra@b@citeb}}~%      lwarp
5     \@biblabel{\@ifundefined{DCN@#2\@extra@b@citeb}{\@warning
6     {Reference '#2' on page \thepage\space was never cited}}{}%
7 %     \DC@llap{$^{\@nameuse{DCN@#2\@extra@b@citeb}}$\ \ }%%o
8     \@citeverb{#2}}\hfil]\if@filesw{\def\protect##1{\string ##1\space}%
9     \immediate\write\@auxout{\string\bibcite{#2}{#1}}}\fi\ignorespaces}
```

File 120   `lwarp-easy-todo.sty`

§ 229   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 \renewcommand{\todoii}[2]{%
11 \ifthenelse{\equal{\@todoobeyfinal}{true}}%
12   {%
13   \ifoptionfinal{\todoenable{false}}{\todoenable{true}}%
14   }%
15   {}%
16 \ifthenelse{\equal{\@todoenable}{true}}%
17   {%
18   \refstepcounter{todos}%
19   \noindent{%
20   \todocolor%
21   \LWR@textcurrentcolor{%
22   \normalfont\scriptsize{\bfseries{\thetodos.#1}}%
23   }%
24   }%
25   \addcontentsline{lod}{todos}{\protect{\thetodos. }\LWR@isolate{#2}}%
26   }%
27   {}%
28 }

File 121   **lwarp-ebook.sty**

§ 230   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 122 **`lwarp-econometrics.sty`**

§ 231   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 123   **`lwarp-ed.sty`**

§ 232   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 124 **`lwarp-ellipsis.sty`**

§ 233   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 125   **`lwarp-embrac.sty`**

§ 234   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}[2017/07/04]
```

```
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 \ifxetexorluatex
15     \LetLtxMacro\LWR@orig@HTML@textsi\LWR@HTML@textsi
16     \RenewDocumentCommand{\LWR@HTML@textsi}{s m}{%
17     \LWR@orig@HTML@textsi{#2}}
18 \fi
19
20 \AtBeginDocument{
21     \LWR@formatted{emph}
22     \LWR@formatted{textit}
23     \LWR@formatted{textsl}
24     \ifxetexorluatex
25         \LWR@formatted{textsi}
26     \fi
27 }
28
```

```
29 \newcommand{\LWR@HTML@EmbracOff}{}
30 \LWR@formatted{EmbracOff}
31
32 \newcommand{\LWR@HTML@EmbracOn}{}
33 \LWR@formatted{EmbracOn}
```

File 126   `lwarp-emptypage.sty`

§ 235   Package   # emptypage

emptypage (*Pkg*)   emptypage is ignored.

for HTML output: Discard all options for lwarp-emptypage:

```
1 \LWR@ProvidesPackageDrop{emptypage}[2010/05/30]
```

File 127   `lwarp-endfloat.sty`

§ 236   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 128   `lwarp-endheads.sty`

§ 237   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 129   **`lwarp-endnotes.sty`**

§ 238   Package   # endnotes

(*Emulates or patches code by* JOHN LAVAGNINO.)

endnotes (*Pkg*)   Patched for HTML.

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   If using MATHJAX, see section 8.5.4 regarding the use of \endnotemark and
numbering   \endnotetext.

for HTML output: `1 \LWR@ProvidesPackagePass{endnotes}`

```
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 }{}{}
```

For MATHJAX:

```
17 \begin{warpMathJax}
18 \def\endnotename{endnote}
19 \appto\LWR@synconotenumbers{\LWR@synconenotenumber{LWRendnote}{\theendnote}}
20 \appto\LWR@synconotenames{\LWR@synconenotename{LWRendnote}{\endnotename}}
21 \CustomizeMathJax{\def\LWRendnote{1}}
22 \CustomizeMathJax{\newcommand{\endnote}[2][\LWRendnote]{{}^{\mathrm{#1}}}}
23 \CustomizeMathJax{\newcommand{\endnotemark}[1][\LWRendnote]{{}^{\mathrm{#1}}}}
24 \end{warpMathJax}
```

File 130   `lwarp-engtlc.sty`

§ 239   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 131 **`lwarp-enotez.sty`**

§ 240   Package **enotez**

(*Emulates or patches code by* Clemens Niederberger.)

enotez (*Pkg*)   enotez is patched for use by lwarp.

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 \cs_gset_protected:Npn \enotez_write_mark:nn #1#2
11   {
12     \bool_if:NTF \l__enotez_hyperfootnotes_bool
13       {
14         \enotezwritemark { \hyperlink {enz.#1} { \enmarkstyle #2 } }
15         \bool_if:NT \l__enotez_hyperbackref_bool
16           {
17 %            \box_move_up:nn {1em} {
```

```
18 %                        \hbox:n {
19                            \hypertarget {enz.#1.backref} { }
20 %                        }
21 %                    }
22                }
23            }
24        { \enotezwritemark { \enmarkstyle #2 } }
25  }
26 \cs_generate_variant:Nn \enotez_write_mark:nn {x}
```

Do not move or \hbox the \hypertarget:

```
27 \cs_gset_protected:Npn \enotez_write_list_number:n #1
28  {
29    \bool_if:NT \l__enotez_hyperfootnotes_bool
30      {
31 %        \box_move_up:nn {1em} { \hbox:n {
32            \hypertarget {enz.#1} { }
33 %        } }
34      }
35    \tl_use:N \l__enotez_list_number_format_tl
36    \tl_if_eq:nxTF {a} { \prop_item:Nn \g__enotez_endnote_man_prop {#1} }
37      {
38        \bool_if:nTF
39          { \l__enotez_hyperfootnotes_bool && \l__enotez_hyperbackref_bool }
40          {
41            \exp_args:Nnx
42            \hyperlink {enz.#1.backref}
43              { \exp_not:V \l__enotez_endnote_mark_tl }
44          }
45          { \prop_item:Nn \g__enotez_endnote_mark_prop {#1} }
46      }
47      {
48        \bool_if:nTF
49          { \l__enotez_hyperfootnotes_bool && \l__enotez_hyperbackref_bool }
50          {
51            \exp_args:Nnx
52            \hyperlink {enz.#1.backref}
53              { \exp_not:V \l__enotez_endnote_mark_tl }
54          }
55          { \tl_use:N \l__enotez_endnote_mark_tl }
56      }
57  }
```

Do not move the label to the left:

```
58 \DeclareTemplateCode {enotez-list} {paragraph} {1}
59  {
60    heading       = \enotez_list_heading:n            ,
61    format        = \l__enotez_list_format_tl         ,
62    number        = \enotez_list_number:n             ,
63    number-format = \l__enotez_list_number_format_tl  ,
64    notes-sep     = \l__enotez_list_notes_sep_dim
65  }
66  {
67    \AssignTemplateKeys
68    \enotez_set_totoc:
69    \enotez_list_heading:n { \l__enotez_list_name_tl }
70    \enotez_list_preamble:
71    \enotez_build_print_list:nnnn {#1}
```

```
72      {}
73      {
74        \par\noindent
75        \group_begin:
76          \tl_use:N \l__enotez_list_format_tl
77 %          \hbox_overlap_left:n
78 %            {
79              \enotez_list_number:n
80                { \enotez_write_list_number:n {##1} }
81              \tl_use:N \c_space_tl
82 %            }
83          % \cs_set:cpn {@currentlabel}
84          %   { \p@endnote \l__enotez_endnote_mark_tl }
85          \tl_use:N \g__enotez_endnote_text_tl
86          \par
87          \dim_compare:nT { \l__enotez_list_notes_sep_dim != 0pt }
88            { \addvspace { \l__enotez_list_notes_sep_dim } }
89        \group_end:
90      }
91      {}
92    \enotez_list_postamble:
93  }
94
95 \ExplSyntaxOff
```

For MATHJAX:

```
96  \begin{warpMathJax}
97  \def\endnotename{endnote}
98  \appto\LWR@syncnotenumbers{\LWR@synconenotenumber{LWRendnote}{\theendnote}}
99  \appto\LWR@syncnotenames{\LWR@synconenotename{LWRendnote}{\endnotename}}
100 \CustomizeMathJax{\def\LWRendnote{1}}
101 \CustomizeMathJax{\newcommand{\endnote}[2][\LWRendnote]{{}^{\mathrm{#1}}}}
102 \CustomizeMathJax{\newcommand{\endnotemark}[1][\LWRendnote]{{}^{\mathrm{#1}}}}
103 \end{warpMathJax}
```

File 132   `lwarp-enumerate.sty`

§ 241   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 133   `lwarp-enumitem.sty`

§ 242   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 134   **lwarp-epigraph.sty**

§ 243   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 135   **`lwarp-epsf.sty`**

§ 244   Package   **epsf**

(*Emulates or patches code by* Том Rокicki.)

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 136   **`lwarp-epsfig.sty`**

§ 245   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 137 **`lwarp-epstopdf.sty`**

§ 246   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 138   **`lwarp-epstopdf-base.sty`**

§ 247   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 139 **`lwarp-eqlist.sty`**

§ 248   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 140 **`lwarp-eqparbox.sty`**

§ 249   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 141 **`lwarp-errata.sty`**

§ 250   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 \xpatchcmd{\erratumAdd}
4     {$_a^{\arabic{erratum}}$}
5 %     {\inlinemathother$_a^{\arabic{erratum}}$\inlinemathnormal}
6     {\textsubscript{a}\textsuperscript{\arabic{erratum}}}
7     {}
8     {\LWR@patcherror{erratum}{erratumAdd}}
9
10 \xpatchcmd{\erratumDelete}
11     {$_d^{\arabic{erratum}}$}
12 %     {\inlinemathother$_d^{\arabic{erratum}}$\inlinemathnormal}
13     {\textsubscript{d}\textsuperscript{\arabic{erratum}}}
14     {}
15     {\LWR@patcherror{erratum}{erratumDelete}}
16
17 \xpatchcmd{\erratumReplace}
18     {$_r^{\arabic{erratum}}$}
19 %     {\inlinemathother$_r^{\arabic{erratum}}$\inlinemathnormal}
20     {\textsubscript{r}\textsuperscript{\arabic{erratum}}}
21     {}
22     {\LWR@patcherror{erratum}{erratumReplace}}
23
24 \xpatchcmd{\erratum}
25     {$_a$}
26 %     {\inlinemathother$_a$\inlinemathnormal}
27     {\textsubscript{a}}
28     {}
29     {\LWR@patcherror{erratum}{erratumDelete}}
30
31 \xpatchcmd{\erratum}
32     {$_d^{\@thefnmark}$}
33 %     {\inlinemathother$_d^{\@thefnmark}$\inlinemathnormal}
34     {\textsubscript{d}\textsuperscript{\@thefnmark}}
35     {}
36     {\LWR@patcherror{erratum}{eDelete}}
37
38 \xpatchcmd{\erratum}
39     {$_r^{\@thefnmark}$}
40 %     {\inlinemathother$_r^{\@thefnmark}$\inlinemathnormal}
41     {\textsubscript{r}\textsuperscript{\@thefnmark}}
42     {}
43     {\LWR@patcherror{erratum}{eReplace}}
```

Finish the current page's errata before closing and reloading the list:

```
44 \preto\PrintErrata{\LWR@maybe@orignewpage}
```

No longer defining math macros with the HTML $:

```
45 \StopDefiningMath
```

File 142 **`lwarp-eso-pic.sty`**

§ 251 Package **eso-pic**

(*Emulates or patches code by* ROLF NIEPRASCHK.)

eso-pic (*Pkg*)   eso-pic is ignored.

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 143   **lwarp-esvect.sty**

§ 252   Package   # esvect

(*Emulates or patches code by* EDDIE SAUDRAIS.)

esvect (*Pkg*)   esvect is used as-is for SVG math, and emulated for MATHJAX.

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 144   **lwarp-etoc.sty**

§ 253   Package   # etoc

etoc (*Pkg*)   etoc is ignored. All commands are nullified.

⚠ \tableofcontents with   The etoc package uses a non-standard syntax which looks ahead after a \tableofcontents
\ref   for a following \ref. These \refs appear in the HTML result unless they are re-
moved. 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.

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 145 **`lwarp-eurosym.sty`**

§ 254   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 146   **`lwarp-everypage.sty`**

§ 255   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 147   **`lwarp-everyshi.sty`**

§ 256   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 148 `lwarp-extarrows.sty`

§ 257   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 149   `lwarp-extramarks.sty`

§ 258   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}[2019/01/31]

2 \newcommand*{\extramarks}[2]{}
3 \newcommand*{\firstleftxmark}{}
4 \newcommand*{\lastleftxmark}{}
5 \newcommand*{\firstrightxmark}{}
6 \newcommand*{\lastrightxmark}{}
7 \newcommand*{\firstxmark}{}
8 \newcommand*{\lastxmark}{}
9 \newcommand*{\topxmark}{}
10 \newcommand*{\topleftxmark}{}
11 \newcommand*{\toprightxmark}{}
12 \newcommand*{\firstleftmark}{}
13 \newcommand*{\lastrightmark}{}
14 \newcommand*{\firstrightmark}{}
15 \newcommand*{\lastleftmark}{}

File 150 **`lwarp-fancybox.sty`**

§ 259   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.

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.

At present there is a bug such that paragraph closing tags are not present in footnotes when \VerbatimFootnotes are selected. The browser usually compensates.

```
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
```

Use HTML superscripts in the footnote even inside a lateximage:

```
16     \renewrobustcmd{\textsuperscript}[1]{\LWR@htmlspan{sup}{##1}}%
```

Use paragraph tags if in a tabular data cell or a lateximage:

```
17     \LWR@htmltagc{\LWR@tagregularparagraph}\LWR@orignewline%
```

Append the footnote to the list:

```
18      \@makefntext{}%
```

The footnote text will follow after \V@@@footnotetext has completed.

```
19    \bgroup%
20    \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.

```
21      \let\LWR@newautopagelabel\LWR@null@newautopagelabel%

22    \ignorespaces%
23 }%
```

\V@@@footnotetext

```
24 \def\V@@@footnotetext{%
25    \LWR@origtilde\LWR@orignewline%
26    \LWR@htmltagc{/\LWR@tagregularparagraph}\LWR@orignewline%
27    \strut\egroup%
28 }
```

```
29 }% AfterEndPreamble
```

```
30 \renewcommand*{\@shadowbox}[1]{%
31 \ifbool{FormatWP}%
32 {\InlineClass[border:1px solid black]{shadowbox}{#1}}%
33 {\InlineClass{shadowbox}{#1}}%
34 }
35
36 \renewcommand*{\@doublebox}[1]{%
37 \ifbool{FormatWP}%
38 {\InlineClass[border:1px double black]{doublebox}{#1}}%
39 {\InlineClass{doublebox}{#1}}%
40 }
41
42 \renewcommand*{\@ovalbox}[2]{%
43 \ifbool{FormatWP}%
44 {\InlineClass[border:1px solid black; border-radius:1ex]{ovalbox}{#2}}%
45 {%
46    \ifthenelse{\isequivalentto{#1}{\thinlines}}%
47        {\InlineClass{ovalbox}{#2}}%
48        {\InlineClass{Ovalbox}{#2}}%
49 }%
50 }
```

Convert minipages, parboxes, and lists into linear text using the LWR@nestspan environment:

```
51 \let\LWR@origSbox\Sbox
52
53 \def\Sbox{\LWR@origSbox\LWR@nestspan}
54
55
56 \let\LWR@origendSbox\endSbox
57
58 \def\endSbox{\endLWR@nestspan\LWR@origendSbox}
```

Beqnarray is adapted for MATHJAX or enclosed inside a lateximage:

```
59 \RenewEnviron{Beqnarray}
60 {\LWR@eqnarrayfactor}
61
62 \csgpreto{Beqnarray*}{\boolfalse{LWR@numbereqnarray}}
```

\GenericCaption is enclosed in an HTML block:

```
63 \renewcommand{\GenericCaption}[1]{%
64     \LWR@figcaption%
65     \LWR@isolate{#1}%
66     \endLWR@figcaption%
67 }
```

Btrivlist is enclosed in an HTML block. This is a tabular, and does not use \item.

\trivlist   {⟨l/c/r⟩} [⟨t/c/b⟩]

```
68 \RenewDocumentEnvironment{Btrivlist}{m o}
69 {%
70     \LWR@stoppars%
71     \begin{BlockClass}{Btrivlist}%
72     \tabular{#1}%
73 }
74 {%
75     \endtabular%
76     \end{BlockClass}%
77     \LWR@startpars%
78 }
```

Btrivlist is also neutralized when used inside a span:

```
79 \AtBeginEnvironment{LWR@nestspan}{%
80     \RenewDocumentEnvironment{Btrivlist}{m o}{}{}%
81 }
```

lwarp's handling of \item is patched to accept fancybox's optional arguments:

```
82 \let\LWRFB@origitemizeitem\LWR@itemizeitem
83 \let\LWRFB@origdescitem\LWR@descitem
84
85 \RenewDocumentCommand{\LWR@itemizeitem}{d()o}{%
86     \IfValueTF{#2}{%
87         \LWRFB@origitemizeitem[#2]%
88     }{%
89         \LWRFB@origitemizeitem%
90     }%
91 }
92
93 \RenewDocumentCommand{\LWR@descitem}{d()o}{%
94     \IfValueTF{#2}{%
95         \LWRFB@origdescitem[#2]~%
96     }{%
97         \LWRFB@origdescitem%
98     }%
99 }
```

```
100 \RenewDocumentCommand{\LWR@nestspanitem}{d()}{%
101     \if@newlist\else{
102         \LWR@htmltagc{br /}%
103         \LWR@orignewline%
104     }\fi%
105     \LWR@origitem%
106 }
```

The various boxed lists become regular lists:

```
107 \renewenvironment{Bitemize}[1][]
108     {%
109         \LWR@spanwarnformat{Bitemize}%
110         \booltrue{LWR@starting@fancybox}%
111         \begin{itemize}%
112         \boolfalse{LWR@starting@fancybox}%
113     }
114     {\end{itemize}}
115
116 \renewenvironment{Benumerate}[1][]
117     {%
118         \LWR@spanwarnformat{Benumerate}%
119         \booltrue{LWR@starting@fancybox}%
120         \begin{enumerate}%
121         \boolfalse{LWR@starting@fancybox}%
122     }
123     {\end{enumerate}}
124
125 \renewenvironment{Bdescription}[1][]
126     {%
127         \LWR@spanwarnformat{Bdescription}%
128         \booltrue{LWR@starting@fancybox}%
129         \begin{description}%
130         \boolfalse{LWR@starting@fancybox}%
131     }
132     {\end{description}}
```

\boxput simply prints one then the other argument, side-by-side instead of above and behind:

```
133 \RenewDocumentCommand{\boxput}{s d() m m}{%
134     \IfBooleanTF{#1}{#3\quad#4}{#4\quad#3}%
135 }
```

Neutralized commands:

```
136 \RenewDocumentCommand{\fancyput}{s d() m}{}
137 \RenewDocumentCommand{\thisfancyput}{s d() m}{}
138
139 \RenewDocumentCommand{\fancypage}{m m}{}
140 \RenewDocumentCommand{\thisfancypage}{m m}{}
141
142 \def\LandScape#1{}
143 \def\endLandScape{}
144 \def\@Landscape#1#2#3{}
145 \def\endLandscape{}
```

Low-level patches for `UseVerbatim` and friends:

```
146 \let\LWRFB@UseVerbatim\UseVerbatim
147 \renewcommand*{\UseVerbatim}[1]{%
148     \LWR@atbeginverbatim{Verbatim}%
149     \LWRFB@UseVerbatim{#1}%
150     \LWR@afterendverbatim%
151 }
152
153 \let\LWRFB@LUseVerbatim\LUseVerbatim
154
155 \renewcommand*{\LUseVerbatim}[1]{%
156     \LWR@atbeginverbatim{LVerbatim}%
157     \noindent%
158     \LWRFB@LUseVerbatim{#1}%
159     \LWR@afterendverbatim%
160 }
161
162 \def\@BUseVerbatim[#1]#2{%
163     \LWR@atbeginverbatim{BVerbatim}%
164     \LWRFB@UseVerbatim{#2}%
165     \LWR@afterendverbatim%
166 }
```

File 151   **`lwarp-fancyhdr.sty`**

§ 260   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}[2021/01/04]
```

```
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 \newcommand{\fancypagestyle}[1]{%
35   \@ifnextchar[{\f@nch@pagestyle{#1}}{\f@nch@pagestyle{#1}[]}%
36 }
37 \long\def\f@nch@pagestyle#1[#2]#3{}
```

File 152   **`lwarp-fancypar.sty`**

§ 261   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 153 `lwarp-fancyref.sty`

§ 262   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 154 `lwarp-fancytabs.sty`

§ 263   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 155   **`lwarp-fancyvrb.sty`**

§ 264   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.

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.

At present there is a bug such that paragraph closing tags are not present in footnotes when \VerbatimFootnotes are selected. The browser usually compensates.

```
1 \AtBeginDocument{\RequirePackage{xcolor}}% for \convertcolorspec
2
3 \LWR@ProvidesPackagePass{fancyvrb}[2008/02/07]
```

Initial default patch for fancyvrb:

```
4 \fvset{frame=none}%
```

After the preamble is loaded, after any patches to Verbatim:

```
5 \AfterEndPreamble{
6 \LWR@traceinfo{Patching fancyvrb.}
```

\VerbatimFootnotes   Patched to use the new version.

```
7 \def\VerbatimFootnotes{%
8     \let\@footnotetext\V@footnotetext%
```

```
9        \let\footnote\V@footnote%
10       \let\LWR@footnotetext\V@footnotetext% lwarp
11 }
```

\V@@footnotetext   Patches in a subset of lwarp's \LWR@footnotetext to the fancyvrb version of \V@@footnotetext.

```
12 \def\V@@footnotetext{%
13 \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.

```
14       \LWR@newautopagelabel{page}%
```

Take the current footnote box, then append:

```
15       \global\setbox\LWR@footnotebox=\vbox\bgroup%
```

Add to any current footnotes:

```
16       \unvbox\LWR@footnotebox%
```

Remember the footnote number for \ref:

```
17       \protected@edef\@currentlabel{%
18           \csname p@footnote\endcsname\@thefnmark%
19       }% @currentlabel
```

Use HTML superscripts in the footnote even inside a lateximage:

```
20       \renewrobustcmd{\textsuperscript}[1]{\LWR@htmlspan{sup}{##1}}%
```

Use paragraph tags if in a tabular data cell or a lateximage:

```
21       \LWR@htmltagc{\LWR@tagregularparagraph}\LWR@orignewline%
```

Append the footnote mark to the list:

```
22       \@makefntext{}%
```

The footnote text will follow after \V@@@footnotetext has completed.

```
23       \bgroup%
24       \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.

```
25       \let\LWR@newautopagelabel\LWR@null@newautopagelabel%

26       \ignorespaces%
27 }%
```

\V@@@footnotetext

```
28 \def\V@@@footnotetext{%
29       \LWR@origtilde\LWR@orignewline%
30       \LWR@htmltagc{/\LWR@tagregularparagraph}\LWR@orignewline%
31       \strut\egroup%
32 }
```

```
33 \preto\FVB@Verbatim{\LWR@forcenewpage}
34 \preto\FVB@LVerbatim{\LWR@forcenewpage}
35 % \preto\FVB@BVerbatim{\LWR@forcenewpage}% Fails, so done below.
```

Simplified to remove PDF formatting:

```
36 \def\FV@BeginListFrame@Single{%
37   \FV@SingleFrameLine{\z@}%
38 }
39
40 \def\FV@EndListFrame@Single{%
41   \FV@SingleFrameLine{\@ne}%
42 }
43
44 \def\FV@BeginListFrame@Lines{%
45   \FV@SingleFrameLine{\z@}%
46 }
47
48 \def\FV@EndListFrame@Lines{%
49     \FV@SingleFrameLine{\@ne}%
50 }
51
52 \renewcommand*{\FV@SingleFrameSep}{}
```

Adds HTML formatting:

```
53 \def\FV@BUseVerbatim#1{%
54     \FV@BVerbatimBegin#1\FV@BVerbatimEnd%
55 }
```

\LWR@FVstyle Holds the style of the verbatim.

```
56 \newcommand*{\LWR@FVstyle}{}
```

The following patches to Verbatim are executed at the start and end of the environment, depending on the choice of frame. Original code is from the fancyvrb package.

```
57 \newcommand*{\LWR@fvstartnone}{%
58 \LWR@traceinfo{fvstartnone}%
59 % \hbox to\z@{
60 \BlockClass[\LWR@FVstyle]{fancyvrb}
61 \LWR@stoppars
62 \ifx\FV@LabelPositionTopLine\relax\else
63     \ifx\FV@LabelBegin\relax\else
64         \FancyVerbRuleColor{\LWR@FVfindbordercolor}
65         \LWR@htmltagc{%
66             div class=\textquotedbl{}fancyvrblabel\textquotedbl\ % space
67           style=\textquotedbl{}color: \LWR@origpound\LWR@tempcolor\textquotedbl%
68         }
69         \LWR@print@textrm{\FV@LabelBegin}% \textrm preserves emdash
70         \LWR@htmltagc{/div}\LWR@orignewline%
71     \fi
72 \fi
73 \LWR@atbeginverbatim{verbatim}%
74 % }%
75 }
76
77 \newcommand*{\LWR@fvendnone}{%
78 \LWR@traceinfo{fvendnone}%
79 % \hbox to\z@{
80 \LWR@afterendverbatim%
```

```
81 \LWR@stoppars%
82 \ifx\FV@LabelPositionBottomLine\relax\else
83     \ifx\FV@LabelEnd\relax\else
84         \FancyVerbRuleColor{\LWR@FVfindbordercolor}
85         \LWR@htmltagc{%
86             div class=\textquotedbl{}fancyvrblabel\textquotedbl\ % space
87             style=\textquotedbl{}color: \LWR@origpound\LWR@tempcolor\textquotedbl%
88         }
89         \LWR@print@textrm{\FV@LabelEnd}
90         \LWR@htmltagc{/div}\LWR@orignewline%
91     \fi
92 \fi
93 \endBlockClass
94 }
95
96 \newcommand*{\LWR@fvstartsingle}{%
97 \LWR@traceinfo{fvstartsingle}%
98 \LWR@fvstartnone%
99 \FV@BeginListFrame@Single%
100 }
101
102 \newcommand*{\LWR@fvendsingle}{%
103 \LWR@traceinfo{fvendsingle}%
104 \FV@EndListFrame@Single%
105 \LWR@fvendnone%
106 }
107
108 \newcommand*{\LWR@fvstartline}{%
109 \LWR@traceinfo{fvstartline}%
110 \LWR@fvstartnone%
111 % \setlength{\LWR@templengthone}{\baselineskip}%
112 \FV@BeginListFrame@Lines%
113 % \setlength{\baselineskip}{\LWR@templengthone}%
114 % \setlength{\baselineskip}{5pt}%
115 }
116
117 \newcommand*{\LWR@fvendline}{%
118 \LWR@traceinfo{fvendline}%
119 \FV@EndListFrame@Lines%
120 \LWR@fvendnone%
121 }
```

The following patches select the start/left/right/end behaviors depending on frame. Original code is from the fancyvrb package.

```
122 \newcommand*{\LWR@FVfindbordercolor}{%
123 \FancyVerbRuleColor%
124 \LWR@findcurrenttextcolor%
125 \color{black}%
126 }
127
128 % border width of \FV@FrameRule
129 \newcommand*{\LWR@FVborderstyle}[1]{%
130 padding#1: \strip@pt\dimexpr \FV@FrameSep\relax\relax pt ; % space
131 \LWR@FVfindbordercolor\LWR@indentHTMLtwo%
132 border#1: \strip@pt\dimexpr \FV@FrameRule\relax\relax pt % space
133 solid {\FancyVerbRuleColor{\LWR@origpound\LWR@tempcolor}} ; % space
134 }
135
136 \def\FV@Frame@none{%
```

```
137 \renewcommand*{\LWR@FVstyle}{\LWR@currenttextcolorstyle}%
138 \let\FV@BeginListFrame\LWR@fvstartnone%
139 \let\FV@LeftListFrame\relax%
140 \let\FV@RightListFrame\relax%
141 \let\FV@EndListFrame\LWR@fvendnone}
142
143 \FV@Frame@none% default values
144
145 \def\FV@Frame@single{%
146 \renewcommand*{\LWR@FVstyle}{%
147     \LWR@currenttextcolorstyle\LWR@indentHTMLtwo%
148     \LWR@FVborderstyle{}%
149 }%
150 \let\FV@BeginListFrame\LWR@fvstartsingle%
151 \let\FV@LeftListFrame\FV@LeftListFrame@Single%
152 \let\FV@RightListFrame\FV@RightListFrame@Single%
153 \let\FV@EndListFrame\LWR@fvendsingle}
154
155 \def\FV@Frame@lines{%
156 \renewcommand*{\LWR@FVstyle}{%
157     \LWR@currenttextcolorstyle\LWR@indentHTMLtwo%
158     \LWR@FVborderstyle{-top}%
159     \LWR@indentHTMLtwo%
160     \LWR@FVborderstyle{-bottom}%
161 }%
162 \let\FV@BeginListFrame\LWR@fvstartline%
163 \let\FV@LeftListFrame\relax%
164 \let\FV@RightListFrame\relax%
165 \let\FV@EndListFrame\LWR@fvendline}
166
167 \def\FV@Frame@topline{%
168 \renewcommand*{\LWR@FVstyle}{%
169     \LWR@currenttextcolorstyle\LWR@indentHTMLtwo%
170     \LWR@FVborderstyle{-top}%
171 }%
172 \let\FV@BeginListFrame\LWR@fvstartline%
173 \let\FV@LeftListFrame\relax%
174 \let\FV@RightListFrame\relax%
175 \let\FV@EndListFrame\LWR@fvendnone}
176
177 \def\FV@Frame@bottomline{%
178 \renewcommand*{\LWR@FVstyle}{%
179     \LWR@currenttextcolorstyle\LWR@indentHTMLtwo%
180     \LWR@FVborderstyle{-bottom}%
181 }%
182 \let\FV@BeginListFrame\LWR@fvstartnone%
183 \let\FV@LeftListFrame\relax%
184 \let\FV@RightListFrame\relax%
185 \let\FV@EndListFrame\LWR@fvendline}
```

Seems to be required in some situations:

```
186 \def\FV@FrameFillLine{}%
```

```
187 \def\FV@Frame@leftline{%
188 \renewcommand*{\LWR@FVstyle}{%
189     \LWR@currenttextcolorstyle\LWR@indentHTMLtwo%
190     \LWR@FVborderstyle{-left}%
191 }%
192 % To define the \FV@FrameFillLine macro (from \FV@BeginListFrame)
```

```
193 \ifx\FancyVerbFillColor\relax%
194 \let\FV@FrameFillLine\relax%
195 \else%
196 \@tempdima\FV@FrameRule\relax%
197 \multiply\@tempdima-\tw@%
198 \edef\FV@FrameFillLine{%
199 {\noexpand\FancyVerbFillColor{\vrule\@width\number\@tempdima sp}%
200 \kern-\number\@tempdima sp}}%
201 \fi%
202 \let\FV@BeginListFrame\LWR@fvstartnone%
203 \let\FV@LeftListFrame\FV@LeftListFrame@Single%
204 \let\FV@RightListFrame\relax%
205 \let\FV@EndListFrame\LWR@fvendnone}
```

Adds the optional label to the top and bottom edges. Original code is from the fancyvrb package.

```
206 \def\FV@SingleFrameLine#1{%
207 %    \hbox to\z@{%
208 %       \kern\leftmargin
209    \ifnum#1=\z@\relax
210      \let\FV@Label\FV@LabelBegin
211    \else
212      \let\FV@Label\FV@LabelEnd
213    \fi
214    \ifx\FV@Label\relax
215 %       \FancyVerbRuleColor{\vrule \@width\linewidth \@height\FV@FrameRule}%
216    \else
217      \ifnum#1=\z@
218 %         \setbox\z@\hbox{\strut\enspace\FV@LabelBegin\enspace\strut}%
219        \ifx\FV@LabelPositionTopLine\relax
220        \else
221        \fi
222      \else
223 %         \setbox\z@\hbox{\strut\enspace\FV@LabelEnd\enspace\strut}%
224        \ifx\FV@LabelPositionBottomLine\relax
225        \else
226        \fi
227      \fi
228    \fi
229 %    \hss
230 %    }
231 }
```

Processes each line, adding optional line numbers. Original code is from the fancyvrb package.

```
232 \def\FV@ListProcessLine#1{%
233    \hbox to \hsize{%
234 %       \kern\leftmargin
235      \hbox to \VerbatimHTMLWidth {%
236      \ifcsvoid{FV@LeftListNumber}{}{\kern 2.5em}%
237        \FV@LeftListNumber%
238 %       \FV@LeftListFrame
239      \FancyVerbFormatLine{#1}%
240      \hss%
241 %       \FV@RightListFrame
242      \FV@RightListNumber%
243      }%
```

```
244      \hss% required to avoid underfull hboxes
245 }
246 }
```

```
247 \def\FV@ListProcessLine@i#1{%
248 %   \hbox{%
249     \ifvoid\@labels\else
250       \hbox to \z@{\kern\@totalleftmargin\box\@labels\hss}%
251     \fi
252     \FV@ListProcessLine{#1}%
253 %   }%
254 %   \let\FV@ProcessLine\FV@ListProcessLine@ii%
255 }
```

```
256 \def\FV@ListProcessLastLine{}
```

BVerbatim (*env.*)

```
257
258 \xpretocmd{\FV@BeginVBox}
259     {%
260         \LWR@forcenewpage% instead of \preto
261         \LWR@atbeginverbatim{bverbatim}%
262     }
263     {}
264     {\LWR@patcherror{fancyvrb}{FV@BeginVBox}}
265
266 \xapptocmd{\FV@EndVBox}
267     {%
268         \LWR@afterendverbatim%
269     }
270     {}
271     {\LWR@patcherror{fancyvrb}{FV@EndVBox}}
```

End of the modifications to make at the end of the preamble:

```
272 } % \AfterEndPreamble
```

File 156   **`lwarp-fbox.sty`**

§ 265   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 $\{\langle \textit{1: top/bottom/left/right}\rangle\}$ $\{\langle \textit{2: t/b/l/r}\rangle\}$ $\{\langle \textit{3: padding, or empty}\rangle\}$

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}
```

Acculumate 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 157 **`lwarp-fewerfloatpages.sty`**

§ 266   Package # fewerfloatpages

fewerfloatpages (*Pkg*)   fewerfloatpages is ignored.

1 \LWR@ProvidesPackageDrop{fewerfloatpages}[2020/02/14]

2 \newcommand\floatpagekeepfraction{\textfraction}
3 \newcounter{floatpagedeferlimit}
4 \newcounter{floatpagekeeplimit}

File 158   **`lwarp-figcaps.sty`**

§ 267   Package # figcaps

(*Emulates or patches code by* PATRICK W. DALY.)

figcaps (*Pkg*)   figcaps is ignored.

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 159   **`lwarp-figsize.sty`**

§ 268   Package # figsize

(*Emulates or patches code by* ANTHONY A. TANBAKUCHI.)

figsize (*Pkg*)   figsize is emulated.

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 160 **`lwarp-fitbox.sty`**

§ 269   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 161 **`lwarp-fix2col.sty`**

§ 270   Package **fix2col**

fix2col (*Pkg*)   fix2col is ignored.

for HTML output: 1 `\LWR@ProvidesPackageDrop{fix2col}[2015/11/13]`

File 162 **`lwarp-fixmath.sty`**

§ 271   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 163 **`lwarp-fixme.sty`**

§ 272   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 164   **lwarp-fixmetodonotes.sty**

§ 273   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 \renewcommand{\NOTES@addtolist}[2]{%
3     \refstepcounter{NOTES@note}%
4 %    \phantomsection% REMOVED
5     \addcontentsline{notes}{NOTES@note}{%
6         \protect\numberline{\theNOTES@note}{{#1}: {#2}}%
7     }%
8 }
9
10 \renewcommand{\NOTES@marker}[2]{\fbox{%
11     \textcolor{#2}{% WAS \color
12         \textbf{#1}}%
13     }}
14
15 \renewcommand{\NOTES@colorline}[2]{%
```

```
16    \bgroup%
17      \ULon{\LWR@backgroundcolor{#1}{#2}}%
18  }
```

File 165   **`lwarp-flafter.sty`**

§ 274   Package   # flafter

flafter (*Pkg*)   flafter is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{flafter}[2018/01/08]
2 \providecommand\fl@trace[1]{}

File 166   **`lwarp-flippdf.sty`**

§ 275   Package   # flippdf

flippdf (*Pkg*)   flippdf is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{flippdf}[2006/06/30]

2 \newcommand\FlipPDF{}
3 \newcommand\UnFlipPDF{}

File 167   **`lwarp-float.sty`**

§ 276   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 78.2 for the \LWR@listof command in the lwarp core.

28 \newcommand{\listof}{\LWR@listof}

File 168   **`lwarp-floatflt.sty`**

§ 277   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 169 **`lwarp-floatpag.sty`**

§ 278   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 170   **`lwarp-floatrow.sty`**

§ 279   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 tab character &   Use \StartDefiningTabulars and \StopDefiningTabulars before and after defining 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.

⚠ \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.

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 171 **lwarp-fltrace.sty**

§ 280   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 172 **lwarp-flushend.sty**

§ 281   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 173 **lwarp-fnbreak.sty**

§ 282   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 174 `lwarp-fncychap.sty`

§ 283 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 175 `lwarp-fnlineno.sty`

§ 284 Package **fnlineno**

fnlineno (*Pkg*) fnlineno is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{fnlineno}[2011/01/07]`

File 176 `lwarp-fnpara.sty`

§ 285 Package **fnpara**

fnpara (*Pkg*) fnpara is ignored.

for HTML output: 1 `\LWR@ProvidesPackageDrop{fnpara}`

File 177 **`lwarp-fnpos.sty`**

§ 286   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 178   **`lwarp-fontawesome.sty`**

§ 287   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. X$\exists$LaTeX 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 X$\exists$LaTeX 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][fontawesomexetex#1SZ\LWR@font@size{}CL\LWR@tempcolor]%`
11 `    \csuse{\LWR@font@size}%`
12 `    \LWR@orig@FA%`
13 `    \LWR@orig@symbol{#1}%`
14 `    \end{lateximage}%`
15 `}`
16
17 `\RenewDocumentCommand{\FA}{}{%`

```
18      \LetLtxMacro\symbol\LWR@fontawesome@xelatex@symbol%
19 }
20
21 \else
22
23 \newcommand*{\LWR@fontawesome@symbolX}[2]{%
24      \LWR@findcurrenttextcolor%
25   \begin{lateximage}*[icon #1][fontawesome#2#1SZ\LWR@font@size{}CL\LWR@tempcolor]%
26      \csuse{\LWR@font@size}%
27      \fontencoding{U}\fontfamily{fontawesome#2}\selectfont%
28      \LWR@orig@symbol{#1}%
29      \end{lateximage}%
30 }
31
32 \newcommand*{\LWR@fontawesome@symbolone}[1]{%
33      \LWR@fontawesome@symbolX{#1}{one}%
34 }
35
36 \newcommand*{\LWR@fontawesome@symboltwo}[1]{%
37      \LWR@fontawesome@symbolX{#1}{two}%
38 }
39
40 \newcommand*{\LWR@fontawesome@symbolthree}[1]{%
41      \LWR@fontawesome@symbolX{#1}{three}%
42 }
43
44 \renewrobustcmd\FAone{%
45      \LetLtxMacro\symbol\LWR@fontawesome@symbolone%
46 }
47
48 \renewrobustcmd\FAtwo{%
49      \LetLtxMacro\symbol\LWR@fontawesome@symboltwo%
50 }
51
52 \renewrobustcmd\FAthree{%
53      \LetLtxMacro\symbol\LWR@fontawesome@symbolthree%
54 }
55 \fi
```

File 179   `lwarp-fontawesome5.sty`

§ 288   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}[2018/07/27]

```
2 \ExplSyntaxOn
3 \cs_set:Nn\fontawesome_use_icon:nn{
4      \LWR@findcurrenttextcolor
5   \cs_if_exist:cTF{c__fontawesome_slot_#2_tl}{
```

```
6    \begin{lateximage}*[#2][fontawesome5#1SZ\LWR@font@size{}CL\LWR@tempcolor]
7      \csuse{\LWR@font@size}
8      \exp_last_unbraced:Nv
9        \__fontawesome_icon_at:nnnn
10       {c__fontawesome_slot_#2_tl}
11         {#1}{#2}
12    \end{lateximage}
13  }{
14      \msg_error:nnxx{fontawesome5}{icon-not-found}{#2}{#1}
15  }
16 }
17 \ExplSyntaxOff
```

File 180 · `lwarp-fontaxes.sty`

§ 289 · 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 181 · `lwarp-fontenc.sty`

§ 290 · 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.

for HTML output:
```
1 \LWR@ProvidesPackagePass{fontenc}[2017/04/05]
```

File 182 **`lwarp-footmisc.sty`**

§ 291 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 183   **lwarp-footnote.sty**

§ 292   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 \def\spewnotes{%
7   \endgroup%
8   \if@savingnotes\else\ifvoid\fn@notes\else\begingroup%
9     \let\@makefntext\@empty%
10     \let\@finalstrut\@gobble%
11     \let\rule\@gobbletwo%
12     \booltrue{LWR@spewingnotes}%        lwarp
13     \@footnotetext{\unvbox\fn@notes}%
14   \endgroup\fi\fi%
15 }
16 \let\endsavenotes\spewnotes
17
18
19 \def\fn@fntext#1{%
20   \ifx\ifmeasuring@\@@undefined%
21     \expandafter\@secondoftwo\else\expandafter\@iden%
22   \fi%
23   {\ifmeasuring@\expandafter\@gobble\else\expandafter\@iden\fi}%
24   {%
25     \global\setbox\fn@notes\vbox{%
26       \unvbox\fn@notes%
27       \LWR@htmltagc{\LWR@tagregularparagraph}%        lwarp
28       \LWR@orignewline%                               lwarp
29       \fn@startnote%
30       \@makefntext{%
31         \rule\z@\footnotesep%
32         \ignorespaces%
33         #1%
34         \@finalstrut\strutbox%
35       }%
36       \fn@endnote%
37     }%
38   }%
39 }
```

Removed print-version formatting:

```
40 \def\fn@startnote{%
```

```
41 %    \@parboxrestore%
42    \protected@edef\@currentlabel{\csname p@\@mpfn\endcsname\@thefnmark}%
43 %    \color@begingroup% *** conflicts with lwarp
44 }
45
46 % \let\fn@endnote\color@endgroup% *** conflicts with lwarp
47 \def\fn@endnote{%
48      \LWR@origtilde\LWR@orignewline%
49      \LWR@htmltagc{/\LWR@tagregularparagraph}\LWR@orignewline%
50      \LWR@origtilde\LWR@orignewline%
51 }
```

Removed print-version formatting:

```
52 \def\fn@startfntext{%
53    \setbox\z@\vbox\bgroup%
54      \LWR@htmltagc{\LWR@tagregularparagraph}%         lwarp
55      \LWR@orignewline%                                lwarp
56      \fn@startnote%
57      \fn@prefntext% Req'd for numbering.
58 %      \rule\z@\footnotesep%
59      \ignorespaces%
60 }
61
```

Removed print-version formatting, added closing paragraph tag:

```
62 \def\fn@endfntext{%
63      \fn@postfntext%
64      \LWR@origtilde\LWR@orignewline%
65      \LWR@htmltagc{/\LWR@tagregularparagraph}%
66      \LWR@orignewline%
67    \egroup%
68    \begingroup%
69      \let\@makefntext\@empty%
70      \let\@finalstrut\@gobble%
71      \LetLtxMacro\rule\@gobbletwo%
72      \booltrue{LWR@spewingnotes}%     lwarp
73      \@footnotetext{\unvbox\z@}%
74    \endgroup%
75 }
```

These have been redefined, so re-\let them again:

```
76 \let\endfootnote\fn@endfntext
77 \let\endfootnotetext\endfootnote
```

File 184   **lwarp-footnotebackref.sty**

§ 293   Package   **footnotebackref**

footnotebackref (*Pkg*)   **footnotebackref** is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{footnotebackref}[2012/07/01]

File 185   `lwarp-footnotehyper.sty`

§ 294   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 186   `lwarp-footnoterange.sty`

§ 295   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 187   `lwarp-footnpag.sty`

§ 296   Package   # footnpag

footnpag (*Pkg*)   footnpag is ignored.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{footnpag}
```

File 188   `lwarp-foreign.sty`

§ 297   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 189   `lwarp-forest.sty`

§ 298   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 \RenewDocumentCommand{\Forest}{s D(){} m}{%
9   \forest@config{#2}%
10  \IfBooleanTF{#1}{%
11      \PackageError{lwarp-forest}%
12      {\protect\Forest* is not supported}%
13      {Lwarp uses an environment for images,\MessageBreak
14          but \protect\Forest* cannot work in an environment.}%
15      \let\forest@next\forest@env%
16  }{\let\forest@next\forest@group@env}%
17      \begin{lateximage}[-forest-~\PackageDiagramAltText]%      lwarp
18  \forest@next{#3}%
19      \end{lateximage}%                          lwarp
20 }
```

File 190   `lwarp-fouridx.sty`

§ 299   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 191 **`lwarp-fourier.sty`**

§ 300 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 192 **`lwarp-framed.sty`**

§ 301 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 193   **`lwarp-froufrou.sty`**

§ 302   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 \RenewDocumentCommand{\froufrou}{s O{}}{%
10  \nopagebreak[4]\par
11
12  \IfBooleanTF{#1}{\@afterindenttrue}{\@afterindentfalse}
13
14  \nopagebreak[4]\@froufrouspacebefore\nopagebreak[4]
15
16  \bgroup
17    \setfroufrou{#2}%
18    \normalsize
19  \ifdefvoid{\setstretch}{}{\setstretch{\setspace@singlespace}}% normally 1
20    \setlength{\parskip}{0pt}
21    \noindent\centering\bgroup%
22      \begin{center}%                                            lwarp
23      \begin{lateximage}*[froufrou][\LWR@latestfroufrou]%        lwarp
24      \@froufrouOrnament%
25      \end{lateximage}%                                         lwarp
26      \end{center}%                                             lwarp
27    \egroup\par
28  \egroup
29
30  \nopagebreak[4]\@froufrouspaceafter\nopagebreak[4]
31
32  \@froufrouFixSpacingAfter
33
34  \nopagebreak[3]
35
36  \@afterheading
37 }
```

File 194   `lwarp-ftcap.sty`

§ 303   Package   # ftcap

ftcap (*Pkg*)   ftcap is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{ftcap}

File 195   `lwarp-ftnright.sty`

§ 304   Package   # ftnright

ftnright (*Pkg*)   ftnright is ignored.

for HTML output: Discard all options for lwarp-ftnright:

1 \LWR@ProvidesPackageDrop{ftnright}[2014/10/28]

File 196 **`lwarp-fullminipage.sty`**

§ 305 Package **fullminipage**

fullminipage (*Pkg*)   fullminipage is ignored.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{fullminipage}[2014/07/06]
```

```
2 \newenvironment{fullminipage}[1][]{}{}
```

File 197 **`lwarp-fullpage.sty`**

§ 306 Package **fullpage**

fullpage (*Pkg*)   fullpage is ignored.

for HTML output: Discard all options for lwarp-fullpage:

```
1 \LWR@ProvidesPackageDrop{fullpage}[1994/06/01]
```

File 198 **`lwarp-fullwidth.sty`**

§ 307 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 199 **`lwarp-fvextra.sty`**

§ 308 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}[2022/11/30]
```

Ignored are highlight, showtabs, obeytabs, tab, tabcolor. Also ignored are all
options regarding line breaking except breaklines, which is emulated as true.

tabsize is honored.

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}
5   {\let\FV@ObeyTabsInit\relax}
6
7 \define@key{FV}{tabcolor}%
8 {}
9
10 \define@key{FV}{tab}{}
11
12 \define@booleankey{FV}{showtabs}%
13 %  {\def\FV@TabChar{\FV@TabColor{\FancyVerbTab}}}%
14 {\let\FV@TabChar\relax}
15 {\let\FV@TabChar\relax}
16
17 \newbool{LWR@FV@breaklines}
18
19 \define@booleankey{FV}{breaklines}%
20   {\boolfalse{FV@breaklines}
21     \booltrue{LWR@FV@breaklines}
22     \let\FV@ListProcessLine\FV@ListProcessLine@NoBreak}
23   {\boolfalse{FV@breaklines}
24     \boolfalse{LWR@FV@breaklines}
25     \let\FV@ListProcessLine\FV@ListProcessLine@NoBreak}
26 % \fvset{breaklines}
27
28 \define@key{FV}{breakanywheresymbolpre}{\def\FancyVerbBreakAnywhereSymbolPre{}}
29 \fvset{breakanywheresymbolpre={}}
30
31 \define@key{FV}{breakanywheresymbolpost}{\def\FancyVerbBreakAnywhereSymbolPost{}}
32 \fvset{breakanywheresymbolpost={}}
33
34 \define@key{FV}{breakbeforesymbolpre}{\def\FancyVerbBreakBeforeSymbolPre{}}
35 \fvset{breakbeforesymbolpre={}}
36
37 \define@key{FV}{breakbeforesymbolpost}{\def\FancyVerbBreakBeforeSymbolPost{}}
38 \fvset{breakbeforesymbolpost={}}
39
40 \define@key{FV}{breakaftersymbolpre}{\def\FancyVerbBreakAfterSymbolPre{}}
41 \fvset{breakaftersymbolpre={}}
42
43 \define@key{FV}{breakaftersymbolpost}{\def\FancyVerbBreakAfterSymbolPost{}}
44 \fvset{breakaftersymbolpost={}}
45
46 \define@key{FV}{breaksymbolleft}{\def\FancyVerbBreakSymbolLeft{}}
47
48 \define@key{FV}{breaksymbol}{\fvset{breaksymbolleft={}}}
49
50 \fvset{breaksymbolleft={}}
51
52 \define@key{FV}{breaksymbolright}{\def\FancyVerbBreakSymbolRight{}}
53 \fvset{breaksymbolright={}}
```

```
54
55 \def\FV@ListProcessLine@NoBreak#1{%
56 %    \hbox to \hsize{%
57 %      \kern\leftmargin
58 %      \hbox to \linewidth{%
59         \FV@LeftListNumber%
60         \FV@LeftListFrame%
61         \FancyVerbFormatLine{%
62           \FancyVerbHighlightLine{%
63             \FV@ObeyTabs{\FancyVerbFormatText{#1}}}}%\hss
64         \FV@RightListFrame%
65         \FV@RightListNumber%
66 %      }%
67 %      \hss}%
68 \null\par%                            lwarp
69 }
70
71
72 \newcommand*{\LWR@FV@linethensep}{%
73     \ifbool{LWR@FV@breaklines}%
74         {\theFancyVerbLine\kern\FV@NumberSep}%
75         {\hbox to\z@{\hss\theFancyVerbLine\kern\FV@NumberSep}}%
76 }
77
78 \newcommand*{\LWR@FV@septhenline}{%
79     \ifbool{LWR@FV@breaklines}%
80         {\kern\FV@NumberSep\theFancyVerbLine}%
81         {\hbox to\z@{\kern\FV@NumberSep\theFancyVerbLine\hss}}%
82 }
83
84 \xpatchcmd{\FV@Numbers@left}
85     {\hbox to\z@{\hss\theFancyVerbLine\kern\FV@NumberSep}}
86     {\LWR@FV@linethensep}
87     {}
88     {\LWR@patcherror{fvextra}{FV@Numbers@left A}}
89
90 \xpatchcmd{\FV@Numbers@left}
91     {\hbox to\z@{\hss\theFancyVerbLine\kern\FV@NumberSep}}
92     {\LWR@FV@linethensep}
93     {}
94     {\LWR@patcherror{fvextra}{FV@Numbers@left B}}
95
96 \xpatchcmd{\FV@Numbers@left}
97     {\hbox to\z@{\hss\theFancyVerbLine\kern\FV@NumberSep}}
98     {\LWR@FV@linethensep}
99     {}
100     {\LWR@patcherror{fvextra}{FV@Numbers@left C}}
101
102 \xpatchcmd{\FV@Numbers@right}
103     {\hbox to\z@{\kern\FV@NumberSep\theFancyVerbLine\hss}}
104     {\LWR@FV@septhenline}
105     {}
106     {\LWR@patcherror{fvextra}{FV@Numbers@right A}}
107
108 \xpatchcmd{\FV@Numbers@right}
109     {\hbox to\z@{\kern\FV@NumberSep\theFancyVerbLine\hss}}
110     {\LWR@FV@septhenline}
111     {}
112     {\LWR@patcherror{fvextra}{FV@Numbers@right B}}
113
```

```
114 \xpatchcmd{\FV@Numbers@right}
115     {\hbox to\z@{\hss\theFancyVerbLine\kern\FV@NumberSep}}
116     {\LWR@FV@linethensep}
117     {}
118     {\LWR@patcherror{fvextra}{FV@Numbers@right C}}
119
120 \xpatchcmd{\FV@Numbers@both}
121     {\hbox to\z@{\hss\theFancyVerbLine\kern\FV@NumberSep}}
122     {\LWR@FV@linethensep}
123     {}
124     {\LWR@patcherror{fvextra}{FV@Numbers@both A}}
125
126 \xpatchcmd{\FV@Numbers@both}
127     {\hbox to\z@{\hss\theFancyVerbLine\kern\FV@NumberSep}}
128     {\LWR@FV@linethensep}
129     {}
130     {\LWR@patcherror{fvextra}{FV@Numbers@both B}}
131
132 \xpatchcmd{\FV@Numbers@both}
133     {\hbox to\z@{\hss\theFancyVerbLine\kern\FV@NumberSep}}
134     {\LWR@FV@linethensep}
135     {}
136     {\LWR@patcherror{fvextra}{FV@Numbers@both C}}
137
138 \xpatchcmd{\FV@Numbers@both}
139     {\hbox to\z@{\kern\FV@NumberSep\theFancyVerbLine\hss}}
140     {\LWR@FV@septhenline}
141     {}
142     {\LWR@patcherror{fvextra}{FV@Numbers@both D}}
143
144 \xpatchcmd{\FV@Numbers@both}
145     {\hbox to\z@{\kern\FV@NumberSep\theFancyVerbLine\hss}}
146     {\LWR@FV@septhenline}
147     {}
148     {\LWR@patcherror{fvextra}{FV@Numbers@both E}}
149
150 \xpatchcmd{\FV@Numbers@both}
151     {\hbox to\z@{\hss\theFancyVerbLine\kern\FV@NumberSep}}
152     {\LWR@FV@linethensep}
153     {}
154     {\LWR@patcherror{fvextra}{FV@Numbers@both F}}
```

File 200   **`lwarp-fwlw.sty`**

§ 309   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 201 **`lwarp-gensymb.sty`**

§ 310 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 202 **`lwarp-gentombow.sty`**

§ 311 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 203 **`lwarp-geometry.sty`**

§ 312 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 204 **`lwarp-ghsystem.sty`**

§ 313   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 \cs_set_protected:Npn \ghsystem_filler:n #1
5   { \emph { \textless #1 \textgreater } }
6
7 \cs_set_protected:Npn \ghsystem_pic:n #1
8   {
9     \__ghsystem_includegraphics:xn
10      {
11 %        scale = \fp_to_tl:N \l__ghsystem_picture_scale_fp
12        width = 1.25cm
13        \exp_not:V \l__ghsystem_picture_includegraphics_tl
14      }
15      { ghsystem_ #1 . \l__ghsystem_picture_type_tl }
16  }
17
18 \ExplSyntaxOff
```

File 205 **`lwarp-gindex.sty`**

§ 314   Package # gindex

(*Emulates or patches code by* JAVIER BEZOS.)

gindex (*Pkg*)   gindex is patched for use by lwarp.

See section 8.6.15.

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 con-
tain \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 206   **`lwarp-gloss.sty`**

§ 315   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

1 \LWR@ProvidesPackagePass{gloss}[2002/07/26]

\BaseJobname is added to the label in case xr or xr-hyper are used.

```
2 \xpatchcmd{\gls@gloss@iii}
3    {\thepage}
4    {\theLWR@previousautopagelabel}
5    {}
6    {\LWR@patcherror{gloss}{gls@gloss@iii}}
7
8 \def\gls@page@i#1#2{%
9   \endgroup%
10  \global\@namedef{glsp@#1}{\nameref{\BaseJobname-autopage-#2}}}%
```

File 207 **lwarp-glossaries.sty**

§ 316   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 \renewcommand*{\@p@glossarysection}[2]{%
 8   \glsclearpage
 9   \LWR@phantomsection
10   \ifdefempty\@@glossarysecstar
11   {%
12     \csname\@@glossarysec\endcsname{#2}%
13   }%
14   {%
```

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:

```
15     \csname\@@glossarysec\endcsname*{#2}%
16     \@gls@toc{#1}{\@@glossarysec}% Moved after the previous line.
17   }%
18   \@@glossaryseclabel
19 }
```

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.

```
20 \newcommand*{\LWR@comp@glossaryname}{\translate{Glossary}}
21
22 \ifdefstrequal{\glossaryname}{\LWR@comp@glossaryname}{
23   \translatelet\LWR@translatetemp{Glossary}
24   \edef\glossaryname{\LWR@translatetemp}
25 }{}
26
27 \newcommand*{\LWR@comp@acronymname}{\translate{Acronym}}
28
29 \ifdefstrequal{\acronymname}{\LWR@comp@acronymname}{
30   \translatelet\LWR@translatetemp{Acronym}
31   \edef\acronymname{\LWR@translatetemp}
32 }{}
```

```
33
34 \newcommand*{\LWR@comp@glssymbolsgroupname}{\translate{Symbols (glossaries)}}
35
36 \ifdefstrequal{\glssymbolsgroupname}{\LWR@comp@glssymbolsgroupname}{
37     \translatelet\LWR@translatetemp{Symbols (glossaries)}
38     \edef\glssymbolsgroupname{\LWR@translatetemp}
39 }{}
40
41 \newcommand*{\LWR@comp@glsnumbersgroupname}{\translate{Numbers (glossaries)}}
42
43 \ifdefstrequal{\glsnumbersgroupname}{\LWR@comp@glsnumbersgroupname}{
44     \translatelet\LWR@translatetemp{Numbers (glossaries)}
45     \edef\glsnumbersgroupname{\LWR@translatetemp}
46 }{}
```

File 208   **`lwarp-gmeometric.sty`**

§ 317   Package   # gmeometric

gmeometric (*Pkg*)   gmeometric is ignored.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{gmeometric}[2008/11/22]
2 \RequirePackageWithOptions{geometry}
```

File 209   **`lwarp-graphics.sty`**

§ 318   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]
```

§ 318.1   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
```

§ 318.2 **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}{}
```

§ 318.3 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 }
```

§ 318.4 \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 }
```

\LWR@ig@useactualimagesize

* [⟨*2: options*⟩] [⟨*3: options*⟩] {⟨*4: filename*⟩}

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@origtilde \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     \LWR@href{\LWR@parsedfilename}%
315     {% start of href
316         \LWR@traceinfo{LWR@includegraphicsb: about to LWR@htmltag}%
317         \LWR@htmltag{\LWR@ig@htmltag}%
318     }% end of href
```

Return to original page size and font size:

```
319     \end{LWR@setvirtualpage}%
```

Clear the single-use alt text:

```
320     \gdef\LWR@ThisAltText{}%
321     \LWR@traceinfo{LWR@includegraphicsb done}%
322 }
```

\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.

```
323 \AtBeginDocument{
324
325 \LWR@traceinfo{Patching includegraphics.}
326
327 \LetLtxMacro\LWR@origincludegraphics\includegraphics
328 \renewrobustcmd*{\includegraphics}
329 {%
```

This graphic should trigger an HTML paragraph even if alone, so ensure that are
doing paragraph handling:

```
330 \LWR@traceinfo{includegraphics}%
331 \LWR@ensuredoingapar%
332 \LWR@includegraphicsb%
333 }% includegraphics
334 }% AtBeginDocument
```

§ 318.5 **Boxes**

\LWR@rotboxorigin   Holds the origin key letters.

```
335 \newcommand*{\LWR@rotboxorigin}{}
```

\LWR@originname   {⟨*letter*⟩}

Given one LATEX origin key value, translate into an HTML origin word:

```
336 \newcommand*{\LWR@originname}[1]{%
337     \ifthenelse{\equal{#1}{t}}{top}{}%
338     \ifthenelse{\equal{#1}{b}}{bottom}{}%
339     \ifthenelse{\equal{#1}{c}}{center}{}%
340     \ifthenelse{\equal{#1}{l}}{left}{}%
341     \ifthenelse{\equal{#1}{r}}{right}{}%
342 }
```

`\LWR@originnames` {⟨*letters*⟩}

Given one- or two-letter LATEX origin key values, translate into HTML origin words:

```
343 \newcommand*{\LWR@originnames}[1]{%
344 \StrChar{#1}{1}[\LWR@strresult]%
345 \LWR@originname{\LWR@strresult}
346 \StrChar{#1}{2}[\LWR@strresult]%
347 \LWR@originname{\LWR@strresult}
348 }
```

Handle the origin key for `\rotatebox`:

```
349 \define@key{krotbox}{origin}{%
350 \renewcommand*{\LWR@rotboxorigin}{#1}%
351 }
```

These keys are ignored:

```
352 \define@key{krotbox}{x}{}
353 \define@key{krotbox}{y}{}
354 \define@key{krotbox}{units}{}
```

`\rotatebox` [⟨*keyval list*⟩] {⟨*angle*⟩} {⟨*text*⟩}

```
355 \AtBeginDocument{
```

The HTML version:

```
356 \NewDocumentCommand{\LWR@HTML@rotatebox}{O{} m +m}{%
```

Reset the origin to "none-given":

```
357 \renewcommand*{\LWR@rotboxorigin}{}
```

Process the optional keys, which may set `\LWR@rotateboxorigin`:

```
358 \setkeys{krotbox}{#1}%
```

Select `inline-block` so that HTML will transform this span:

```
359 \LWR@htmltagc{%
360     span\LWR@indentHTML
361     style=\textquotedbl\LWR@indentHTML
362     display: inline-block;\LWR@indentHTML
```

If an origin was given, translate and print the origin information:

```
363     \ifthenelse{\NOT\equal{\LWR@rotboxorigin}{}}%
364       {transform-origin: \LWR@originnames{\LWR@rotboxorigin};\LWR@indentHTML}%
365         {}%
```

Print the rotation information:

```
366     \LWR@rotstyle{-ms-}{#2}\LWR@indentHTML
367     \LWR@rotstyle{-webkit-}{#2}\LWR@indentHTML
368     \LWR@rotstyle{}{#2}\textquotedbl\LWR@orignewline%
369 }\LWR@orignewline%
```

Print the text to be rotated:

```
370 \begin{LWR@nestspan}%
371 #3%
```

Close the span:

```
372 \LWR@htmltagc{/span}%
373 \end{LWR@nestspan}%
374 }
```

The high-level interface:

```
375 \LWR@formatted{rotatebox}
376
377 }% AtBeginDocument
```

\scalebox {⟨*h-scale*⟩} [⟨*v-scale*⟩] {⟨*text*⟩}

```
378 \AtBeginDocument{
```

The HTML version:

```
379 \NewDocumentCommand{\LWR@HTML@scalebox}{m o m}{%
```

Select `inline-block` so that HTML will transform this span:

```
380 \LWR@htmltagc{%
381     span\LWR@indentHTML
382     style=\textquotedbl\LWR@indentHTML
383     display: inline-block;\LWR@indentHTML
```

Print the scaling information:

```
384     \LWR@scalestyle{-ms-}{#1}{\IfNoValueTF{#2}{#1}{#2}}\LWR@indentHTML
385     \LWR@scalestyle{-webkit-}{#1}{\IfNoValueTF{#2}{#1}{#2}}\LWR@indentHTML
386     \LWR@scalestyle{}{#1}{\IfNoValueTF{#2}{#1}{#2}}
387     \textquotedbl\LWR@orignewline
388 }\LWR@orignewline%
```

Print the text to be scaled:

```
389 \begin{LWR@nestspan}%
390 #3%
```

Close the span:

```
391 \LWR@htmltagc{/span}%
392 \end{LWR@nestspan}%
393 }
```

The high-level interface:

```
394 \LWR@formatted{scalebox}
395
396 }% AtBeginDocument
```

\reflectbox {⟨*text*⟩}

```
397 \AtBeginDocument{
398
399 \newcommand{\LWR@HTML@reflectbox}[1]{%
400     \scalebox{-1}[1]{#1}%
401 }% \reflectbox
402
403 \LWR@formatted{reflectbox}
404
405 }% AtBeginDocument
```

\resizebox {⟨*h-length*⟩} {⟨*v-length*⟩} {⟨*text*⟩}

Simply prints its text argument.

```
406 \AtBeginDocument{
407
408 \NewDocumentCommand{\LWR@HTML@resizebox}{s m m m}{%
409     #4%
410 }
411
412 \LWR@formatted{resizebox}
413
414 }% AtBeginDocument
```

File 210   **`lwarp-graphicx.sty`**

§ 319   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 211   **`lwarp-grffile.sty`**

§ 320   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 212 `lwarp-grid.sty`

§ 321   Package **grid**

grid (*Pkg*)   grid is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{grid}[2009/06/16]

2 \newenvironment*{gridenv}{}{}

File 213   `lwarp-grid-system.sty`

§ 322   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 214   `lwarp-gridset.sty`

§ 323   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 215   **lwarp-hang.sty**

§ 324   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 style=\textquotedbl%
29             \LWR@print@mbox{list-style-type:none;} % extra space
30             \LWR@print@mbox{%
31                 margin-left:\LWR@printlength{\LWR@templengthone}%
32             } ; % extra space
33             \LWR@print@mbox{%
34                 text-indent:-\LWR@printlength{\hangingindent}%
35             }%
36         \textquotedbl%
37     }%
38     \LetLtxMacro\item\LWR@itemizeitem%
39     \list{}{}%
40 }
41 {\endlist}
42
43 \newenvironment{compacthang}
44 {\hanginglist}
45 {\endhanginglist}
46
47 \newlength{\labeledleftmargin}
```

```
48 \setlength{\labeledleftmargin}{0em}
49
50 \newenvironment{labeledpar}[2]
51 {%
52     \BlockClass[%
53         \LWR@findhangingleftmargin%
54         \LWR@print@mbox{margin-left:\LWR@printlength{\LWR@templengthone}} ; %
55         \LWR@print@mbox{text-indent:-\LWR@printlength{\hangingindent}}%
56     ]{labeledpar}#2%
57 }
58 {\endBlockClass}
59
60 \newenvironment{labeledlist}[1]
61 {\hanginglist}
62 {\endhanginglist}
63
64 \newenvironment{compactlabel}[1]
65 {\hanginglist}
66 {\endhanginglist}
```

File 216   **`lwarp-hanging.sty`**

§ 325   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 217   **`lwarp-hepunits.sty`**

§ 326   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 218 **`lwarp-hhline.sty`**

§ 327   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 219   `lwarp-hhtensor.sty`

§ 328   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 220   `lwarp-hypbmsec.sty`

§ 329   Package   # hypbmsec

hypbmsec (*Pkg*)   hypbmsec is emulated by the lwarp core.

for HTML output: 1 `\LWR@ProvidesPackageDrop{hypbmsec}[2016/05/16]`

File 221   `lwarp-hypcap.sty`

§ 330   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 222   `lwarp-hypdestopt.sty`

§ 331   Package   # hypdestopt

hypdestopt (*Pkg*)   hypdestopt is ignored.

for HTML output:

```
1 \LWR@ProvidesPackageDrop{hypdestopt}[2016/05/21]
```

File 223   `lwarp-hypernat.sty`

§ 332   Package   # hypernat

hypernat (*Pkg*)   hypernat is ignored.

for HTML output:

```
1 \LWR@ProvidesPackageDrop{hypernat}[2001/07/09]
```

File 224   `lwarp-hyperref.sty`

§ 333   Package   # hyperref

(*Emulates or patches code by* SEBASTIAN RAHTZ, HEIKO OBERDIEK, THE LATEX3 PROJECT.)

hyperref (*Pkg*)   hyperref is emulated.

for HTML output:

```
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
```

```
254 \LetLtxMacro\href\LWR@href
255 \LetLtxMacro\nolinkurl\LWR@nolinkurl
256 \LetLtxMacro\url\LWR@url
257 \LetLtxMacro\phantomsection\LWR@phantomsection
```

```
258 \newcommand*{\hyperbaseurl}[1]{}
```

No application for lwarp:

```
259 \newcommand*{\HyperDestNameFilter}[1]{#1}
260 \newcommand*{\HyperDestLabelReplace}[1]{#1}
261 \newcommand*{\HyperDestRename}[2]{}
```

No application for lwarp:

```
262 \newcommand*{\hyperget}[2]{}
```

\hyperimage   {⟨*URL*⟩} {⟨*alt text*⟩}

Insert an image with alt text:

```
263 \NewDocumentCommand{\LWR@hyperimageb}{m +m}{%
264     \LWR@ensuredoingapar%
265     \def\LWR@templink{#1}%
266     \@onelevel@sanitize\LWR@templink%
267     \LWR@htmltag{%
268         img src=\textquotedbl\LWR@templink\textquotedbl\ %
269         alt=\textquotedbl#2\textquotedbl\ %
270         class=\textquotedbl{}hyperimage\textquotedbl%
271     }%
272     \LWR@ensuredoingapar%
273     \endgroup%
274 }
275
276 \newrobustcmd*{\hyperimage}{%
277     \begingroup%
278     \LWR@linkcatcodes%
279     \LWR@hyperimageb%
280 }
281
```

\hyperdef   {⟨*1: category*⟩} {⟨*2: name*⟩} {⟨*3: text*⟩}

Creates an HTML anchor to category.name with the given text.

```
282 \NewDocumentCommand{\LWR@hyperdefb}{m m +m}{%
283     \LWR@ensuredoingapar%
284     \LWR@label@createtag{#1.#2}%
285     #3%
286     \endgroup%
287 }
288
289 \newcommand*{\hyperdef}{%
290     \begingroup%
291     \LWR@linkcatcodes%
292     \LWR@hyperdefb%
293 }
294
```

\LWR@hyperrefb $\quad\quad\quad\quad\quad$ {⟨*1: URL*⟩} {⟨*2: category*⟩} {⟨*3: name*⟩} {⟨*4: text*⟩}

Creates an HTML link to URL#category.name with the given text.

```
295 \newcommand{\LWR@hyperreffinish}[1]{%
296     \begingroup%
297     \RenewDocumentCommand{\ref}{s m}{\LWR@print@ref{##2}}%
298     #1%
299     \endgroup%
300     \LWR@htmltag{/a}%
301 }
302
303 \newcommand*{\LWR@hyperrefbb}[3]{%
304     \LWR@htmltag{%
305         a href=\textquotedbl%
306             \detokenize\expandafter{#1}\LWR@hashmark%
307             \detokenize\expandafter{#2}.\detokenize\expandafter{#3}%
308         \textquotedbl%
309         \LWR@addlinktitle%
310     }%
311     \endgroup%
312     \LWR@hyperreffinish%
313 }
314
315 \newrobustcmd*{\LWR@hyperrefb}{%
316     \begingroup%
317     \LWR@linkcatcodes%
318     \LWR@hyperrefbb%
319 }
```

\LWR@hyperrefc $\quad\quad\quad\quad\quad$ [⟨*label*⟩] {⟨*text*⟩}

Creates text as an HTML link to the LATEX label.

```
320 \NewDocumentCommand{\LWR@hyperrefcb}{O{label}}{%
321     \LWR@startref{#1}%
322     \endgroup%
323     \LWR@hyperreffinish%
324 }
325
326 \newcommand*{\LWR@hyperrefc}{%
327     \begingroup%
328     \LWR@linkcatcodes%
329     \LWR@hyperrefcb%
330 }
```

\hyperref $\quad\quad\quad\quad\quad$ {⟨*1: URL*⟩} {⟨*2: category*⟩} {⟨*3: name*⟩} {⟨*4: text*⟩} $\quad$ — or —
$\quad\quad\quad\quad\quad\quad\quad$ [⟨*1: label*⟩] {⟨*2: text*⟩}

```
331 \DeclareRobustCommand*{\hyperref}{%
332     \LWR@ensuredoingapar%
333     \@ifnextchar[\LWR@hyperrefc\LWR@hyperrefb%
334 }
```

\hypertarget $\quad\quad\quad\quad\quad$ {⟨*name*⟩} {⟨*text*⟩}

Creates an anchor to name with the given text.

```
335 \NewDocumentCommand{\LWR@hypertargetb}{m +m}{%
336     \label{LWR-ht-#1}%
337     #2%
```

```
338     \endgroup%
339 }
340
341 \newcommand*{\hypertarget}{%
342     \LWR@ensuredoingapar%
343     \begingroup%
344     \LWR@linkcatcodes%
345     \LWR@hypertargetb%
346 }
```

\hyperlink   {⟨*name*⟩} {⟨*text*⟩}

Creates a link to the anchor created by hypertarget, with the given link text.

Declared because also defined by memoir.

```
347 \DeclareDocumentCommand{\LWR@hyperlinkb}{m}{%
348     \ifbool{LWR@insidemathcomment}%
349         {\endgroup}%
350         {\LWR@hyperrefcb[LWR-ht-#1]}%
351 }
352
353 \DeclareDocumentCommand{\hyperlink}{}{%
354     \LWR@ensuredoingapar%
355     \begingroup%
356     \LWR@linkcatcodes%
357     \LWR@hyperlinkb%
358 }
```

\LWR@nullify@hyperref   {⟨*1: URL*⟩} {⟨*2: category*⟩} {⟨*3: name*⟩} {⟨*4: text*⟩}   — or —
                                  [⟨*1: label*⟩] {⟨*2: text*⟩}

```
359 \newcommand{\LWR@nullify@hyperrefb}[2][]{}
360
361 \newcommand*{\LWR@nullify@hyperref}{%
362     \@ifnextchar[\LWR@nullify@hyperrefb\@fourthoffour%
363 }
```

To nullify in a lateximage or svg math. \hypertarget must be left active for references to work, and does not harm.

```
364 \appto\LWR@restoreorigformatting{%
365     \LetLtxMacro\hyperdef\@thirdofthree
366     \LetLtxMacro\hyperlink\@secondoftwo%
367     \LetLtxMacro\hyperref\LWR@nullify@hyperref%
368 }
```

\autoref   * {⟨*label*⟩}

For html, \cleveref is used instead.

```
369 \NewDocumentCommand{\autoref}{s m}{%
370     \IfBooleanTF{#1}{\ref{#2}}{\cref{#2}}%
371 }
```

\autopageref   {⟨*label*⟩}

For html, \cleveref is used instead.

```
372 \NewDocumentCommand{\autopageref}{s m}{%
```

```
373     \IfBooleanTF{#1}{\cpageref{#2}}{\cref{#2}}%
374 }
```

Default names:

```
375 \def\equationautorefname{Equation}%
376 \def\footnoteautorefname{footnote}%
377 \def\itemautorefname{item}%
378 \def\figureautorefname{Figure}%
379 \def\tableautorefname{Table}%
380 \def\partautorefname{Part}%
381 \def\appendixautorefname{Appendix}%
382 \def\chapterautorefname{chapter}%
383 \def\sectionautorefname{section}%
384 \def\subsectionautorefname{subsection}%
385 \def\subsubsectionautorefname{subsubsection}%
386 \def\paragraphautorefname{paragraph}%
387 \def\subparagraphautorefname{subparagraph}%
388 \def\FancyVerbLineautorefname{line}%
389 \def\theoremautorefname{Theorem}%
390 \def\pageautorefname{page}%
```

\pdfstringdef   {⟨*macroname*⟩} {⟨*TEXstring*⟩}

```
391 \newcommand{\pdfstringdef}[2]{}
```

\pdfbookmark   [⟨*level*⟩] {⟨*text*⟩} {⟨*name*⟩}

```
392 \newcommand{\pdfbookmark}[3][]{}
```

\currentpdfbookmark   {⟨*text*⟩} {⟨*name*⟩}

```
393 \newcommand{\currentpdfbookmark}[2]{}
```

\subpdfbookmark   {⟨*text*⟩} {⟨*name*⟩}

```
394 \newcommand{\subpdfbookmark}[2]{}
```

\belowpdfbookmark   {⟨*text*⟩} {⟨*name*⟩}

```
395 \newcommand{\belowpdfbookmark}[2]{}
```

\texorpdfstring   {⟨*TEXstring*⟩} {⟨*PDFstring*⟩}

```
396 \let\texorpdfstring\relax
397 \newcommand{\texorpdfstring}[2]{#1}
```

                               {⟨*commands*⟩}

\pdfstringdefDisableCommands
```
398 \newcommand{\pdfstringdefDisableCommands}[1]{}
```

\hypercalcbp   {⟨*dimen*⟩}   From hyperref.

```
399 \def\hypercalcbp#1{%
400     \strip@pt\dimexpr 0.99626401\dimexpr(#1)\relax\relax
401 }%
```

\Acrobatmenu   {⟨*menuoption*⟩} {⟨*text*⟩}

402 \newcommand{\Acrobatmenu}[2]{}

\TextField   [⟨*parameters*⟩] {⟨*label*⟩}

403 \DeclareRobustCommand{\TextField}[2][]{}

\CheckBox   [⟨*parameters*⟩] {⟨*label*⟩}

404 \DeclareRobustCommand{\CheckBox}[2][]{}

\ChoiceMenu   [⟨*parameters*⟩] {⟨*label*⟩} {⟨*choices*⟩}

405 \DeclareRobustCommand{\ChoiceMenu}[3][]{}

\PushButton   [⟨*parameters*⟩] {⟨*label*⟩}

406 \DeclareRobustCommand{\PushButton}[2][]{}

\Submit   [⟨*parameters*⟩] {⟨*label*⟩}

407 \DeclareRobustCommand{\Submit}[2][]{}

\Reset   [⟨*parameters*⟩] {⟨*label*⟩}

408 \DeclareRobustCommand{\Reset}[2][]{}

\Gauge   [⟨*parameters*⟩] {⟨*label*⟩}

409 \DeclareRobustCommand{\Gauge}[2][]{}

\LayoutTextField   {⟨*label*⟩} {⟨*field*⟩}

410 \newcommand*{\LayoutTextField}[2]{}

\LayoutChoiceField   {⟨*label*⟩} {⟨*field*⟩}

411 \newcommand*{\LayoutChoiceField}[2]{}

\LayoutCheckField   {⟨*label*⟩} {⟨*field*⟩}

412 \newcommand*{\LayoutCheckField}[2]{}

\MakeRadioField   {⟨*width*⟩} {⟨*height*⟩}

413 \newcommand*{\MakeRadioField}[2]{}

\MakeCheckField   {⟨*width*⟩} {⟨*height*⟩}

414 \newcommand*{\MakeCheckField}[2]{}

\MakeTextField   {⟨*width*⟩} {⟨*height*⟩}

415 \newcommand*{\MakeTextField}[2]{}

\MakeChoiceField   {⟨*width*⟩} {⟨*height*⟩}

416 \newcommand*{\MakeChoiceField}[2]{}

\MakeFieldButton   {⟨*text*⟩}

417 \newcommand{\MakeFieldButton}[1]{}

File 225   **lwarp-hyperxmp.sty**

§ 334   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 226   **`lwarp-hyphenat.sty`**

§ 335   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\texttt
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 227   **`lwarp-idxlayout.sty`**

§ 336   Package   # idxlayout

(*Emulates or patches code by* THOMAS TITZE.)

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 228   **`lwarp-ifoddpage.sty`**

§ 337   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 229 **`lwarp-imakeidx.sty`**

§ 338   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 8.6.18 for how to setup *lwarpmk* to process the indexes with imakeidx, both with and without shell escape.

for HTML output: 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 \catcode`\_=12%
23
24 \def\@index[#1]{%
25     \ifstrequal{#1}{\imki@jobname}%
26     {%
27         \@ifundefined{#1@idxfile}%
```

```
28        {%
29            \PackageWarning{lwarp-imakeidx}{Undefined index file '#1'}%
30            \begingroup
31            \@sanitize
32            \imki@nowrindex%
33        }%
34        {%
35            \edef\@idxfile{#1}%
36            \begingroup
37            \@sanitize
38            \@wrindex\@idxfile%
39        }%
40    }%
41    {%
42        \@ifundefined{#1_html@idxfile}%
43        {%
44            \PackageWarning{lwarp-imakeidx}{Undefined index file '#1_html'}%
45            \begingroup
46            \@sanitize
47            \imki@nowrindex%
48        }%
49        {%
50            \edef\@idxfile{#1_html}%
51            \begingroup
52            \@sanitize
53            \@wrindex\@idxfile%
54        }%
55    }%
56 }
57
58 \catcode'\_=8%
```

\item

\subitem

\subsubitem   HTML versions of \item, etc.:

```
59 \appto\theindex{%
60    \LetLtxMacro\item\LWR@indexitem%
61    \LetLtxMacro\subitem\LWR@indexsubitem%
62    \LetLtxMacro\subsubitem\LWR@indexsubsubitem%
63 }
```

\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:

```
64 \renewcommand\imki@wrindexentrysplit[3]{%
65    \addtocounter{LWR@autoindex}{1}%
66    \expandafter\protected@write\csname#1@idxfile\endcsname{}%
67        {\string\indexentry{#2}{\arabic{LWR@autoindex}}}%
```

The label is assigned after the file write to avoid conflict with cleveref.

```
68    \label{LWRindex-\arabic{LWR@autoindex}}%
69 }
70
71 \renewcommand\imki@wrindexentryunique[3]{%
72    \addtocounter{LWR@autoindex}{1}%
```

```
73      \protected@write\@indexfile{}%
74          {\string\indexentry[#1]{#2}{\arabic{LWR@autoindex}}}%
```

The label is assigned after the file write to avoid conflict with cleveref.

```
75      \label{LWRindex-\arabic{LWR@autoindex}}%
76 }
77
78 \def\imki@wrindexsplit#1#2{%
79 \imki@wrindexentrysplit{#1}{#2}{\thepage}%
80 \endgroup\imki@showidxentry{#1}{#2}%
81   \@esphack%
82 }
83
84 \def\imki@wrindexunique#1#2{%
85 \imki@wrindexentryunique{#1}{#2}{\thepage}%
86 \endgroup\imki@showidxentry{#1}{#2}%
87   \@esphack%
88   }
89
```

\LWR@imki@setxdydefopts

Sets the *xindy* HTML options, ignoring the user's settings.

```
90 \newcommand*{\LWR@imki@setxdydefopts}{%
91     \edef\imki@options{ \space %
92         -M \space \LWR@xindyStyle\space %
93         -L \space \LWR@xindyLanguage\space %
94         -C \space \LWR@xindyCodepage\space %
95     }%
96 }
```

\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.

```
97 \newcommand*{\LWR@imki@setdefopts}[1]{%
98 \ifblank{#1}{%
99     \edef\imki@options{\space -s \space \LWR@makeindexStyle \space}%
100    \ifdefstring{\imki@progdefault}{xindy}{\LWR@imki@setxdydefopts}{}%
101    \ifdefstring{\imki@progdefault}{texindy}{\LWR@imki@setxdydefopts}{}%
102    \ifdefstring{\imki@progdefault}{truexindy}{\LWR@imki@setxdydefopts}{}%
103 }{%
104    \edef\imki@options{\space #1 \space}%
105 }%
106 }
```

\imki@makeindex   Use the new HTML options:

```
107 \xpatchcmd{\imki@makeindex}
108     {\let\imki@options\space}
109     {\LWR@imki@setdefopts{}}%
110     {}
111     {\LWR@patcherror{imakeidx}{makeindex}}
```

Use the new HTML options.

```
112 \define@key{imki}{options}{\LWR@imki@setdefopts{#1}}
```

\imki@resetdefaults   Use the new HTML options:

```
113 \xpatchcmd{\imki@resetdefaults}
114     {\def\imki@options{ }}
115     {\LWR@imki@setdefopts{}}
116     {}
117     {\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

```
118 \AtBeginDocument{
119 \renewenvironment*{theindex}{%
120     \imki@maybeaddtotoc
121      \imki@indexlevel{\indexname}
122     \LetLtxMacro\item\LWR@indexitem%
123     \LetLtxMacro\subitem\LWR@indexsubitem%
124     \LetLtxMacro\subsubitem\LWR@indexsubsubitem%
125 }{}
126 }% AtBeginDocument
```

Update to the new defaults:

```
127 \imki@resetdefaults
```

Update to the new patches:

\AtBeginDocument is because \@wrindex is previously defined as \AtBeginDocument in the lwarp core.

```
128 \ifimki@splitindex
129   \let\imki@startidx\imki@startidxunique
130   \AtBeginDocument{\let\@wrindex\imki@wrindexunique}
131   \let\imki@putindex\imki@putindexunique
132   \let\imki@wrindexentry\imki@wrindexentryunique
133   \let\imki@startidxsplit\@undefined
134   \let\imki@wrindexsplit\@undefined
135   \let\imki@putindexsplit\@undefined
136 \else
137   \let\imki@startidx\imki@startidxsplit
138   \AtBeginDocument{\let\@wrindex\imki@wrindexsplit}
139   \let\imki@putindex\imki@putindexsplit
140   \let\imki@wrindexentry\imki@wrindexentrysplit
141   \let\imki@startidxunique\@undefined
142   \let\imki@wrindexunique\@undefined
143   \let\imki@putindexunique\@undefined
144 \fi
```

File 230 **`lwarp-impnattypo.sty`**

§ 339   Package **impnattypo**

impnattypo (*Pkg*)   impnattypo is ignored.

for HTML output: 1 `\LWR@ProvidesPackageDrop{impnattypo}[2019/03/04]`

File 231   **`lwarp-index.sty`**

§ 340   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 \xpatchcmd{\newindex}
 3    {\x@newindex[thepage]}
 4    {%
 5        \@tempswatrue%
 6        \x@newindex[theLWR@autoindex]%
 7    }
 8    {}
 9    {\LWR@patcherror{index}{newindex}}
10
11 \xpatchcmd{\renewindex}
12    {\x@renewindex[thepage]}
13    {%
14        \@tempswatrue%
15        \x@renewindex[theLWR@autoindex]%
16    }
17    {}
18    {\LWR@patcherror{index}{renewindex}}
```

Patched to set a new autoindex:

```
19 \xpatchcmd{\@wrindex}
20    {\begingroup}
21    {%
22        \addtocounter{LWR@autoindex}{1}%                    lwarp
23        \label{LWRindex-\arabic{LWR@autoindex}}%     lwarp
24        \begingroup%
25    }
26    {}
27    {\LWR@patcherror{index}{@wrindex}}
```

\AtBeginDocument lwarp core \lets \@wrindex to \LWR@wrindex. Since the index package has been loaded, \let to its version instead:

```
28 \let\LWR@index@wrindex\@wrindex
29
30 \AtBeginDocument{
31 \let\@wrindex\LWR@index@wrindex
32 }
```

Modified to add \index@prologue:

```
33 \AtBeginDocument{
34 \renewenvironment*{theindex}{%
35     \LWR@indexsection{\indexname}%
36     \ifx\index@prologue\@empty\else
37         \index@prologue
38         \bigskip
39     \fi
40     \LetLtxMacro\item\LWR@indexitem%
41     \LetLtxMacro\subitem\LWR@indexsubitem%
42     \LetLtxMacro\subsubitem\LWR@indexsubsubitem%
43 }{}
44 }% AtBeginDocument
```

Disabled:

```
45 \def\@showidx#1{}
46 \let\@texttop\relax
47 \renewcommand*{\raggedbottom}{}
48 \renewcommand*{\flushbottom}{}
49 \renewcommand*{\markboth}[2]{}
50 \renewcommand*{\markright}[1]{}
```

File 232   **`lwarp-inputtrc.sty`**

§ 341   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 \renewcommand*{\IT@prim@input}[1]{%
3   \typeout{\IT@indent\IT@currfile INPUTTING #1}%
4 %% ... TODO: option to write to '.log' only.
5   \xdef\IT@filestack{{\IT@currfile}\IT@filestack}%
6   \xdef\IT@currfile{#1}%
7   \expandafter \gdef\expandafter \IT@indent\expandafter{%
8       \IT@indent \IT@indent@unit}%                    lwarp
9   \@@input#1%                                         lwarp
10  \expandafter\IT@pop@indent\IT@indent   \@nil% lwarp
11  \expandafter\IT@pop@file  \IT@filestack\@nil% lwarp
```

12 \IT@maybe@returnmessage%% v0.2   lwarp
13 }

File 233 `lwarp-intopdf.sty`

§ 342 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 234 `lwarp-isomath.sty`

§ 343 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 235 `lwarp-isotope.sty`

§ 344 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 236 **lwarp-jurabib.sty**

§ 345   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 \renewcommand*{\jb@biblaw@item}{%
7      \hspace{0.5em}%
8 %      $\triangleright$
9      \HTMLunicode{25B7}%        lwarp%
10      \hspace{0.5em}%
11 }
12
13 \renewrobustcmd{\jbarchsig}[2]{%
14      \ifjbweareinbib
15         \settowidth{\jb@subarchitemwidth}{\jbsamesubarchindent+#1}%
16      \setlength{\jb@subarchentrywidth}{\textwidth-\jb@subarchitemwidth-4em}%
17 %      \begin{tabular}{@{}p{\jb@subarchitemwidth}@{}j{\jb@subarchentrywidth}@{}}%
18         #1\ifjb@dot\unskip\unskip\unskip.\fi
19 %          &
20         \quad%           lwarp
21         \ifthenelse{\equal{#2}{}}{}{\jbarchnameformat{#2}}%
22 %      \end{tabular}
23      \fi
24 }%
25
26
27 \xpatchcmd{\jb@do@post@item}
28      {\begin{tabular}{p{\jb@biblaw@item@width}j{\jb@biblaw@entry@width}}}
29      {}
30      {}
31      {\LWR@patcherror{jurabib}{jb@do@post@item 1}}
32
33 \xpatchcmd{\jb@do@post@item}
34      {\multicolumn{2}{p{\columnwidth}}{\jb@@name}}
35      {\jb@@name}
36      {}
37      {\LWR@patcherror{jurabib}{jb@do@post@item 2}}
38
39 \xpatchcmd{\jb@do@post@item}
40      {\jb@biblaw@item & \jb@@fulltitle}
41      {\jb@biblaw@item \quad \jb@@fulltitle}
42      {}
```

```
43        {\LWR@patcherror{jurabib}{jb@do@post@item 3}}

44

45 \xpatchcmd{\jb@do@post@item}
46        {\end{tabular}}
47        {}
48        {}
49        {\LWR@patcherror{jurabib}{jb@do@post@item 4}}

50

51 \xpatchcmd{\jb@do@post@item}
52        {\begin{minipage}[t]{\bibnumberwidth}}
53        {}
54        {}
55        {\LWR@patcherror{jurabib}{jb@do@post@item 5}}

56

57 \xpatchcmd{\jb@do@post@item}
58        {\end{minipage}}
59        {\quad}
60        {}
61        {\LWR@patcherror{jurabib}{jb@do@post@item 6}}
```

File 237   **`lwarp-karnaugh-map.sty`**

§ 346   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 \RenewDocumentEnvironment{karnaugh-map}{s O{4} O{4} O{1} O{$X_1X_0$} O{$X_3X_2$} O{$X_5X_4$}} {%
3   \begingroup
4     % store map size {[START]
5       \renewcommand{\@karnaughmap@var@mapsizex@}{#2}%
6       \renewcommand{\@karnaughmap@var@mapsizey@}{#3}%
7       \renewcommand{\@karnaughmap@var@mapsizez@}{#4}%
8     % [END]}
9     % determinate if markings should be color or black and white
10    \IfBooleanTF{#1}{%
11      % should be black and white
12      \renewcommand{\@karnaughmap@var@bw@}{1}%
13    }{%
14      % should be color
15      \renewcommand{\@karnaughmap@var@bw@}{0}%
16    }%
17    %
18    % find matching matrix template and alignment parameters {[START]
19    \newcommand{\@karnaughmap@local@matrixtemplate@}{0}% '0' is considered as missing matrix template
20      \newcommand{\@karnaughmap@local@maprealignmentx@}{0}%
21      \newcommand{\@karnaughmap@local@maprealignmenty@}{0}%
22    \ifnum\@karnaughmap@var@mapsizex@\@karnaughmap@var@mapsizey@\@karnaughmap@var@mapsizez@=221
23        \renewcommand{\@karnaughmap@local@matrixtemplate@}{%
```

```
24              \&                   0 \&                   1 \& \phantom{0} \\
25         0 \& |(000000)|  \phantom{0} \& |(000001)|  \phantom{0} \&            \\
26         1 \& |(000010)|  \phantom{0} \& |(000011)|  \phantom{0} \&            \\
27      \phantom{0} \&                   \&                   \&            \\
28      }%
29    \fi
30  \ifnum\@karnaughmap@var@mapsizex@\@karnaughmap@var@mapsizey@\@karnaughmap@var@mapsizez@=241
31      \renewcommand{\@karnaughmap@local@matrixtemplate@}{%
32              \&                   0 \&                   1 \& \phantom{00} \\
33         00 \& |(000000)|  \phantom{0} \& |(000001)|  \phantom{0} \&            \\
34         01 \& |(000010)|  \phantom{0} \& |(000011)|  \phantom{0} \&            \\
35         11 \& |(000110)|  \phantom{0} \& |(000111)|  \phantom{0} \&            \\
36         10 \& |(000100)|  \phantom{0} \& |(000101)|  \phantom{0} \&            \\
37      \phantom{00} \&                   \&                   \&            \\
38      }%
39    \fi
40  \ifnum\@karnaughmap@var@mapsizex@\@karnaughmap@var@mapsizey@\@karnaughmap@var@mapsizez@=421
41      \renewcommand{\@karnaughmap@local@matrixtemplate@}{%
42              \&                  00 \&                  01 \&                  11 \&                  10 \& \phan
43         0 \& |(000000)|  \phantom{0} \& |(000001)|  \phantom{0} \& |(000011)|  \phantom{0} \& |(000
44         1 \& |(000100)|  \phantom{0} \& |(000101)|  \phantom{0} \& |(000111)|  \phantom{0} \& |(000
45      \phantom{00} \&                   \&                   \&                   \&            \&
46      }%
47    \fi
48  \ifnum\@karnaughmap@var@mapsizex@\@karnaughmap@var@mapsizey@\@karnaughmap@var@mapsizez@=441
49      \renewcommand{\@karnaughmap@local@matrixtemplate@}{%
50              \&                  00 \&                  01 \&                  11 \&                  10 \& \phan
51         00 \& |(000000)|  \phantom{0} \& |(000001)|  \phantom{0} \& |(000011)|  \phantom{0} \& |(00
52         01 \& |(000100)|  \phantom{0} \& |(000101)|  \phantom{0} \& |(000111)|  \phantom{0} \& |(00
53         11 \& |(001100)|  \phantom{0} \& |(001101)|  \phantom{0} \& |(001111)|  \phantom{0} \& |(00
54         10 \& |(001000)|  \phantom{0} \& |(001001)|  \phantom{0} \& |(001011)|  \phantom{0} \& |(00
55      \phantom{00} \&                   \&                   \&                   \&            \&
56      }%
57    \fi
58  \ifnum\@karnaughmap@var@mapsizex@\@karnaughmap@var@mapsizey@\@karnaughmap@var@mapsizez@=442
59      \renewcommand{\@karnaughmap@local@matrixtemplate@}{%
60              \&                  00 \&                  01 \&                  11 \&                  10 \& \phan
61         00 \& |(000000)|  \phantom{0} \& |(000001)|  \phantom{0} \& |(000011)|  \phantom{0} \& |(00
62         01 \& |(000100)|  \phantom{0} \& |(000101)|  \phantom{0} \& |(000111)|  \phantom{0} \& |(00
63         11 \& |(001100)|  \phantom{0} \& |(001101)|  \phantom{0} \& |(001111)|  \phantom{0} \& |(00
64         10 \& |(001000)|  \phantom{0} \& |(001001)|  \phantom{0} \& |(001011)|  \phantom{0} \& |(00
65      \phantom{00} \&                   \&                   \&                   \&            \&
66      }%
67      \renewcommand{\@karnaughmap@local@maprealignmentx@}{2.5}%
68    \fi
69  \ifnum\@karnaughmap@var@mapsizex@\@karnaughmap@var@mapsizey@\@karnaughmap@var@mapsizez@=444
70      \renewcommand{\@karnaughmap@local@matrixtemplate@}{%
71              \&                  00 \&                  01 \&                  11 \&                  10 \& \phan
72         00 \& |(000000)|  \phantom{0} \& |(000001)|  \phantom{0} \& |(000011)|  \phantom{0} \& |(00
73         01 \& |(000100)|  \phantom{0} \& |(000101)|  \phantom{0} \& |(000111)|  \phantom{0} \& |(00
74         11 \& |(001100)|  \phantom{0} \& |(001101)|  \phantom{0} \& |(001111)|  \phantom{0} \& |(00
75         10 \& |(001000)|  \phantom{0} \& |(001001)|  \phantom{0} \& |(001011)|  \phantom{0} \& |(00
76      \phantom{00} \&                   \&                   \&                   \&            \&
77         00 \& |(100000)|  \phantom{0} \& |(100001)|  \phantom{0} \& |(100011)|  \phantom{0} \& |(10
78         01 \& |(100100)|  \phantom{0} \& |(100101)|  \phantom{0} \& |(100111)|  \phantom{0} \& |(10
79         11 \& |(101100)|  \phantom{0} \& |(101101)|  \phantom{0} \& |(101111)|  \phantom{0} \& |(10
80         10 \& |(101000)|  \phantom{0} \& |(101001)|  \phantom{0} \& |(101011)|  \phantom{0} \& |(10
81      \phantom{00} \&                   \&                   \&                   \&            \&
82      }%
83      \renewcommand{\@karnaughmap@local@maprealignmentx@}{2.5}%
```

```
84           \renewcommand{\@karnaughmap@local@maprealignmenty@}{-2.5}%
85         \fi
86      % [END]}
87    % test if a matrix template is found or not(aka "\@karnaughmap@local@matrixtemplate@" equals to '0')
88      \ifdefstring{\@karnaughmap@local@matrixtemplate@}{0}{%  lwarp
89 %     \ifnum0=\@karnaughmap@local@matrixtemplate@%  original
90        % print error if no template could be found
91        \PackageError{lwarp-karnaugh-map}{%
92          Can not find a template fitting your specification
93         (\@karnaughmap@var@mapsizex@\space x \@karnaughmap@var@mapsizey@\space x
94          \@karnaughmap@var@mapsizez@)%
95        }{%
96          Existing templates have the following dimensions:
97          2x2x1, 2x4x1, 4x2x1, 4x4x1, 4x4x2, and 4x4x4.
98        }%
99 %     \fi   original
100     }{\relax}%     lwarp
101     \begin{tikzpicture}
102        % grid
103        % for all dimensions
104       \draw[color=black, ultra thin] (0,0) grid (\@karnaughmap@var@mapsizex@,\@karnaughmap@var@mapsizey@
105        % when there are 2 sub maps
106        \ifnum\@karnaughmap@var@mapsizez@=2
107          \draw[color=black, ultra thin] (5,0) grid (9,4);
108        \fi
109        % when there are 4 sub maps
110        \ifnum\@karnaughmap@var@mapsizez@=4
111          \draw[color=black, ultra thin] (5,0) grid (9,4);
112          \draw[color=black, ultra thin] (0,-5) grid (4,-1);
113          \draw[color=black, ultra thin] (5,-5) grid (9,-1);
114        \fi
115        % labels
116        % for all dimensions
117       \node[above] at (\@karnaughmap@var@mapsizex@*0.5,\@karnaughmap@var@mapsizey@+0.9) {\small{#5}};
118        \node[left] at (-0.9,\@karnaughmap@var@mapsizey@*0.5) {\small{#6}};
119        % when there are 2 sub maps
120        \ifnum\@karnaughmap@var@mapsizez@=2
121          \node[above] at (7,4.9) {\small{#5}};
122          % extra sub maps labels
123          \node[below] at (2,-0.1) {\small{#7$=0$}};
124          \node[below] at (7,-0.1) {\small{#7$=1$}};
125        \fi
126        % when there are 4 sub maps
127        \ifnum\@karnaughmap@var@mapsizez@=4
128          \node[above] at (7,4.9) {\small{#5}};
129          \node[left] at (-0.9,-3) {\small{#6}};
130          % extra sub maps labels
131          \node[below] at (2,-0.1) {\small{#7$=00$}};
132          \node[below] at (7,-0.1) {\small{#7$=01$}};
133          \node[below] at (2,-5.1) {\small{#7$=10$}};
134          \node[below] at (7,-5.1) {\small{#7$=11$}};
135        \fi
136        % data
137        \matrix[
138          matrix of nodes,
139          ampersand replacement=\&,
140          column sep={1cm,between origins},
141          row sep={1cm,between origins},
142        ] at (\@karnaughmap@var@mapsizex@*0.5+\@karnaughmap@local@maprealignmentx@,\@karnaughmap@var@maps
143          \@karnaughmap@local@matrixtemplate@%
```

```
144        };
145 }{
146     \end{tikzpicture}
147  \endgroup
148 }
```

File 238 **`lwarp-keyfloat.sty`**

§ 347   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 \renewcommand*{\KFLT@maybeendfloatrow}{%
34   \ifnumless{\value{KFLT@thiscol}}{\value{KFLT@numcols}}%
35   {}% thiscol < numcols
36   {% >=
37   \defcounter{KFLT@thiscol}{0}%
38   }%
39 }%
40
41 \renewcommand{\KFLT@trackrows}%
42 {%

If are nested inside a keyfloats or a subfloat:

43   \ifboolexpr{%
44   test {\ifnumgreater{\value{KFLT@keyfloatdepth}}{0}} or%
45   bool{KFLT@inkeysubfloats}%
46   }%
47   {% nested

Tracks row start and end:

48   \KFLT@maybestartfloatrow%

Possibly fill space between columns:

49   \ifnumgreater{\value{KFLT@thiscol}}{1}%
50   {%
51 %   \hfill%
52   }%
53   {}%
54   }% nested
55   {}% not nested
56 }

57 \RenewDocumentCommand{\KFLT@onefigureimage}{m}
58 {%
59 \LWR@traceinfo{KFLT@onefigureimage}%
60 % \begin{lrbox}{\KFLT@envbox}%
61 \ifthenelse{\NOT\equal{\KFLT@lw}{}}%
62   {%

63   \ifdimgreater{\KFLT@h}{0pt}%
64   {%
65   \KFLT@frame{%
66   \includegraphics%

```
67                  [%
68                      scale=\KFLT@s,%
69                      width=\KFLT@imagewidth,%
70                      height=\KFLT@h,%
71                      \KFLT@keepaspectratio,%
72                  ]{#1}%
73              }%
74          }%
75          {%
76              \KFLT@frame{\includegraphics%
77              [scale=\KFLT@s,width=\KFLT@imagewidth]{#1}}%
78          }%
79      }%
80      {% not linewidth
81          \ifthenelse{\dimtest{\KFLT@w}{>}{0pt}}%
82          {% width is given
83              \ifthenelse{\dimtest{\KFLT@h}{>}{0pt}}%
84              {% w and h
85                  \KFLT@frame{\includegraphics[%
86                      scale=\KFLT@s,%
87                      width=\KFLT@imagewidth,%
88                      height=\KFLT@h,%
89                      \KFLT@keepaspectratio,%
90                  ]{#1}}%
91              }% w and h
92              {% only w
93                  \KFLT@frame{\includegraphics%
94                  [scale=\KFLT@s,width=\KFLT@imagewidth]{#1}}%
95              }% only w
96          }% width is given
97          {% width is not given
98              \ifthenelse{\dimtest{\KFLT@h}{>}{0pt}}%
99              {%
100                 \KFLT@frame{\includegraphics%
101                 [scale=\KFLT@s,height=\KFLT@h]{#1}}%
102             }%
103             {%
104                 \KFLT@frame{\includegraphics%
105                 [scale=\KFLT@s]{#1}}%
106             }%
107         }% width is not given
108     }% not linewidth
109 % \end{lrbox}%
110 % \unskip%
111 % \KFLT@findenvboxwidth%
112 % \begin{turn}{\KFLT@r}%
113 % \KFLT@frame{\usebox{\KFLT@envbox}}%
114 % \unskip%
115 % \end{turn}%
116 \LWR@traceinfo{KFLT@onefigureimage: done}%
117 }


118 \RenewDocumentEnvironment{KFLT@boxinner}{}
119 {%
120     \LWR@traceinfo{KFLT@boxinner}%
121     \LWR@stoppars%
122     \minipagefullwidth%
123     \ifboolexpr{bool{KFLT@ft} or bool{KFLT@f}}{%
124         \fminipage{\KFLT@imagewidth}%
125     }{%
```

```
126            \minipage{\KFLT@imagewidth}%
127        }%
128 }
129 {%
130        \ifboolexpr{bool{KFLT@ft} or bool{KFLT@f}}{%
131            \endfminipage%
132        }{%
133            \endminipage%
134        }%
135        \LWR@startpars%
136        \LWR@traceinfo{KFLT@boxinner: done}%
137 }
```

```
138 \newcommand*{\LWR@KFLT@settextalign}[1]{%
139        \def\LWR@KFLT@textalign{justify}%
140        \ifcsstring{KFLT@#1textalign}{\centering}%
141            {\def\LWR@KFLT@textalign{center}}%
142            {}%
143        \ifcsstring{KFLT@#1textalign}{\raggedleft}%
144            {\def\LWR@KFLT@textalign{right}}%
145            {}%
146        \ifcsstring{KFLT@#1textalign}{\raggedright}%
147            {\def\LWR@KFLT@textalign{left}}%
148            {}%
149 }
150
151 \renewcommand{\KFLT@addtext}[1]
152 {%
```

Is there text to add?

```
153        \ifcsempty{KFLT@#1t}%
154        {}% no text
155        {% text to add
156            {% local
```

Add some space, then create a <div> to contain the text:

```
157            \addvspace{\smallskipamount}%
158            \LWR@KFLT@settextalign{#1}%
159            \begin{BlockClass}[text-align:\LWR@KFLT@textalign]{floatnotes}%
```

Set the alignment and some text parameters:

```
160 %          \csuse{KFLT@#1textalign}%
161 %          \footnotesize%
162            \setlength{\parskip}{1.5ex}%
163            \setlength{\parindent}{0em}%
```

Typeset the actual text:

```
164            \csuse{KFLT@#1t}%
```

Close it all out with a little more space:

```
165            \end{BlockClass}%
166 %          \par\addvspace{2ex}%
167        }% local
```

```
168        }% text to add
169 }
170
171 \IfPackageLoadedTF{tocdata}
172 {}
173 {% tocdata not loaded
174
175     \newcommand*{\LWR@KFLT@setnamealign}[1]{%
176         \def\LWR@KFLT@textalign{justify}%
177         \ifstrequal{#1}{\centering}%
178             {\def\LWR@KFLT@textalign{center}}%
179             {}%
180         \ifstrequal{#1}{\raggedleft}%
181             {\def\LWR@KFLT@textalign{right}}%
182             {}%
183         \ifstrequal{#1}{\raggedright}%
184             {\def\LWR@KFLT@textalign{left}}%
185             {}%
186     }
187
188     \renewcommand*{\KFLT@@addartisttext}[3]{%
```

Add space and create the name inside a <div>:

```
189 %          \addvspace{\medskipamount}%
190 %      \begin{minipage}{\linewidth}%
191         \LWR@KFLT@setnamealign{#3}%
192         \begin{BlockClass}[text-align:\LWR@KFLT@textalign]{floatnotes}%
```

Text alignment is #3, and depends on artist or author:

```
193 %      #3%
```

#1 is empty or 'subgrp'
#2 is empty for artist, 'u' for author:

```
194         \footnotesize\textsc{%
195             \KFLT@optionalname{\csuse{KFLT@#1a#2p}}%
196             \KFLT@optionalname{\csuse{KFLT@#1a#2f}}%
197             \csuse{KFLT@#1a#2l}%
198             \csuse{KFLT@#1a#2s}%
199         }%
200 %      \end{minipage}%
201         \end{BlockClass}
202 %          \par\addvspace{2ex}%
203     }
204
205 }% tocdata not loaded
```

Env   KFLT@marginfloat   [⟨*offset*⟩] {⟨*type*⟩}

```
206 \DeclareDocumentEnvironment{KFLT@marginfloat}{O{-1.2ex} m}
207 {%
208     \uselengthunit{PT}%
209     \LWR@BlockClassWP%
210         {float:right; width:2in; margin:10pt}%
211         {}%
212         (note)%
213         {marginblock}%
```

```
214       \renewcommand*{\@captype}{#2}%
215       \minipage{1.2\LWR@usersmarginparwidth}%
216       \setlength{\marginparwidth}{.95\LWR@usersmarginparwidth}%
217 }
218 {%
219       \endminipage%
220       \endLWR@BlockClassWP%
221 }


222 \DeclareDocumentEnvironment{marginfigure}{o}
223       {\begin{KFLT@marginfloat}{figure}}
224       {\end{KFLT@marginfloat}}
225
226 \DeclareDocumentEnvironment{margintable}{o}
227       {\begin{KFLT@marginfloat}{table}}
228       {\end{KFLT@marginfloat}}
```

Env keywrap   {⟨*width*⟩} {⟨*keyfloat*⟩}

```
229 \DeclareDocumentEnvironment{keywrap}{m +m}
230 {%
231       \begin{LWR@setvirtualpage}*
232       \setlength{\LWR@templengthone}{#1}%
233       \begin{LWR@BlockClassWP}%
234           {%
235           float:right; width:\LWR@printlength{\LWR@templengthone};  % extra space
236               margin:10pt%
237           }%
238           {}%
239           (note)%
240           {marginblock}%
241       \setlength{\linewidth}{.95\LWR@templengthone}%
242       \booltrue{KFLT@keywrap}%
243       #2%
244       \end{LWR@BlockClassWP}%
245       \end{LWR@setvirtualpage}%
246 }
247 {}


248 }% AtBeginDocument
```

File 239 **lwarp-keystroke.sty**

§ 348   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 \LetLtxMacro\LWR@HTML@Shift\Shift
31 \xpatchcmd{\LWR@HTML@Shift}
32     {$\Uparrow$}
33     {\HTMLunicode{21D1}}
34     {}
35     {}
36 \LWR@formatted{Shift}
37
38 \LetLtxMacro\LWR@HTML@PgUp\PgUp
39 \xpatchcmd{\LWR@HTML@PgUp}
40     {$\uparrow$}
41     {\HTMLunicode{2191}}
42     {}
43     {}
44 \LWR@formatted{PgUp}
45
46 \LetLtxMacro\LWR@HTML@PgDown\PgDown
47 \xpatchcmd{\LWR@HTML@PgDown}
48     {$\downarrow$}
49     {\HTMLunicode{2193}}
50     {}
51     {}
52 \LWR@formatted{PgDown}
```

File 240   **`lwarp-kpfonts.sty`**

§ 349   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 241   **`lwarp-kpfonts-otf.sty`**

§ 350   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 242   **`lwarp-layaureo.sty`**

§ 351   Package   # layaureo

layaureo (*Pkg*)   layaureo is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{layaureo}[2004/09/16]`

File 243   **`lwarp-layout.sty`**

§ 352   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 244   **`lwarp-layouts.sty`**

§ 353   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 245 **`lwarp-leading.sty`**

§ 354   Package **leading**

leading (*Pkg*)   leading is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{leading}[2008/12/11]

2 \newcommand\leading[1]{}

File 246   **`lwarp-leftidx.sty`**

§ 355   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 247   **`lwarp-letterspace.sty`**

§ 356   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 248   **`lwarp-lettrine.sty`**

§ 357   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 249   **`lwarp-libertinust1math.sty`**

§ 358   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 250   **`lwarp-lineno.sty`**

§ 359   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 251   **lwarp-lips.sty**

§ 360   Package   **lips**

(*Emulates or patches code by* MATT SWIFT.)

lips (*Pkg*)   lips is emulated.

```
1 % \LWR@ProvidesPackageDrop{lips}
2 \PackageInfo{lwarp}{Using the lwarp version of package 'lips'.}%
```

```
3 \ProvidesPackage{lwarp-lips}[2001/08/31]
4
5 \NewDocumentCommand{\Lips}{}{\textellipsis}
6
7 \NewDocumentCommand{\BracketedLips}{}{[\textellipsis]}
8
9 \let\lips\Lips
10 \let\olips\lips
11
12 \DeclareOption*{}
13 \DeclareOption{mla}{
14 \let\lips\BracketedLips
15 }
16 \ProcessOptions\relax
17
18 \newcommand \LPNobreakList {}
```

File 252   `lwarp-lipsum.sty`

§ 361   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}[2021-03-03]
```

```
2 \SetLipsumParListItemEnd{%
3     \LWR@closeparagraph%

4     \leavevmode\LWR@orignewline%

5 }
```

File 253   `lwarp-listings.sty`

§ 362   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}[2018/09/02]
```

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 \renewcommand{\lst@Init}[1]{%
```

Perform the listings initialization:

```
24 \LWR@traceinfo{lst@Init}%
```

\LWR@forcenewpage is moved to the start to avoid a spurious bug with paragraph handling and conditionals.

```
25 \lst@ifdisplaystyle%         lwarp
26     \LWR@forcenewpage%       lwarp
27 \fi%                         lwarp
```

Escapes do not work yet, and are disabled:

```
28 \let\lst@ifmathescape\iffalse%       lwarp
29 \let\lst@DefEsc\relax%               lwarp
30 \def\lst@escapebegin{}%              lwarp
31 \def\lst@escapeend{}%                lwarp

32 \renewcommand*{\@captype}{lstlisting}%       lwarp
33     \let\lst@aboveskip\z@\let\lst@belowskip\z@%       lwarp
34     \gdef\lst@boxpos{t}%              lwarp
35     \let\lst@frame\@empty%           lwarp
36     \let\lst@frametshape\@empty%     lwarp
37     \let\lst@framershape\@empty%     lwarp
38     \let\lst@framebshape\@empty%     lwarp
39     \let\lst@framelshape\@empty%     lwarp
40     \lstframe@\lst@frameround ffff\relax%   lwarp
41     \lst@multicols\@empty%  lwarp
42     \begingroup%
```

Inside the listing, temporarily prevent underfull \hbox warnings.

```
43              \hbadness=10000\relax%
```

```
44     \ifx\lst@float\relax\else%
45         \edef\@tempa{\noexpand\lst@beginfloat{lstlisting}[\lst@float]}%
46         \expandafter\@tempa%
47     \fi%
48     \ifx\lst@multicols\@empty\else%
49         \edef\lst@next{\noexpand\multicols{\lst@multicols}}%
50         \expandafter\lst@next%
51     \fi%
52     \ifhmode\ifinner \lst@boxtrue \fi\fi%
53     \lst@ifbox%
54         \lsthk@BoxUnsafe%
55         \hbox to\z@\bgroup%
56             $\if t\lst@boxpos \vtop%
57         \else \if b\lst@boxpos \vbox%
58         \else \vcenter \fi\fi%
59         \bgroup \par\noindent%
60     \else%
61         \lst@ifdisplaystyle%
62             \lst@EveryDisplay%
63             \par\penalty-50\relax%
64             \vspace\lst@aboveskip%
65         \fi%
66     \fi%
67     \normalbaselines%
68     \abovecaptionskip\lst@abovecaption\relax%
69     \belowcaptionskip\lst@belowcaption\relax%
70     \lst@MakeCaption t%
```

Use the overall listing label instead of the line number label:

```
71 \LWR@traceinfo{lst@Init: defining current label !\@currentlabel!}%
72     \let\LWR@listings@currentlabel\@currentlabel%              lwarp
73 \LWR@traceinfo{lst@Init: defining current label !\cref@currentlabel!}%
74     \let\LWR@listings@cref@currentlabel\cref@currentlabel%   lwarp
```

```
75 \LWR@traceinfo{lst@Init: preinit and init}%
76     \lsthk@PreInit \lsthk@Init%
```

```
77     \let\@currentlabel\LWR@listings@currentlabel%             lwarp
78     \let\cref@currentlabel\LWR@listings@cref@currentlabel%   lwarp
```

```
79 \LWR@traceinfo{lst@Init: M}%
80     \lst@ifdisplaystyle
81         \global\let\lst@ltxlabel\@empty
82         \if@inlabel
83             \lst@ifresetmargins
84                 \leavevmode
85             \else
86                 \xdef\lst@ltxlabel{\the\everypar}%
87                 \lst@AddTo\lst@ltxlabel{%
88                     \global\let\lst@ltxlabel\@empty
89                     \everypar{\lsthk@EveryLine\lsthk@EveryPar}}%
90             \fi
91         \fi
92         \everypar\expandafter{\lst@ltxlabel
93                         \lsthk@EveryLine\lsthk@EveryPar}%
94     \else
95         \everypar{}%
96         \let\lst@NewLine\@empty
```

```
 97     \fi
 98 \LWR@traceinfo{lst@Init: P}%
 99     \lsthk@InitVars \lsthk@InitVarsBOL
100     \lst@Let{13}\lst@MProcessListing
101     \let\lst@Backslash#1%
102     \lst@EnterMode{\lst@Pmode}{\lst@SelectCharTable}%
103     \lst@InitFinalize%
104 \LWR@traceinfo{lst@Init: S}%
```

Avoids extra horizontal space:

```
105 \def\lst@framelr{}%      lwarp
```

```
106 \LWR@traceinfo{lst@Init: finished origlst@Init}%
107 \lst@ifdisplaystyle%     lwarp
```

Creating a display.

Disable line numbers, produce the <pre>, then reenable line numbers.

```
108     \LWR@traceinfo{lst@Init: About to create verbatim.}%  lwarp
109     \let\lsthk@EveryPar\relax%                    lwarp

110     \LWR@atbeginverbatim{programlisting}%      lwarp
111
112     \let\lsthk@EveryPar\LWR@origlsthkEveryPar%  lwarp
113 \else%                                          lwarp
```

Inline, so open a <span>:

```
114     \ifbool{LWR@verbtags}{\LWR@htmltag{%        lwarp
115       span class=\textquotedbl{}inlineprogramlisting\textquotedbl%    lwarp
116     }}{}%                                        lwarp
117 \fi%                                             lwarp
118 \LWR@traceinfo{lst@Init: done}%
119 }
```

\lst@DeInit   Done at the end of a listing.

```
120 \renewcommand*{\lst@DeInit}{%
121 \LWR@traceinfo{lst@DeInit}%
122 \lst@ifdisplaystyle%
```

Creating a display.

Disable line numbers, produce the </pre>, then reenable line numbers:

```
123     \let\lsthk@EveryPar\relax%
124     \LWR@afterendverbatim%
125     \let\lsthk@EveryPar\LWR@origlsthkEveryPar%
126 \else%
```

Inline, so create the closing :

```
127     \ifbool{LWR@verbtags}{\noindent\LWR@htmltag{/span}}{}%
128 \fi%
```

Final listings deinit:

```
129     \lst@XPrintToken \lst@EOLUpdate
130     \global\advance\lst@newlines\m@ne
131     \lst@ifshowlines
132        \lst@DoNewLines
133     \else
134        \setbox\@tempboxa\vbox{\lst@DoNewLines}%
135     \fi
```

```
136    \lst@ifdisplaystyle \par\removelastskip \fi
137    \lsthk@ExitVars\everypar{}\lsthk@DeInit\normalbaselines\normalcolor
138    \lst@MakeCaption b%
139    \lst@ifbox
140        \egroup $\hss \egroup
141        \vrule\@width\lst@maxwidth\@height\z@\@depth\z@
142    \else
143        \lst@ifdisplaystyle
144            \par\penalty-50\vspace\lst@belowskip
145        \fi
146    \fi
147    \ifx\lst@multicols\@empty\else
148        \def\lst@next{\global\let\@checkend\@gobble
149                      \endmulticols
150                      \global\let\@checkend\lst@@checkend}
151        \expandafter\lst@next
152    \fi
153    \ifx\lst@float\relax\else
154        \expandafter\lst@endfloat
155    \fi
156    \endgroup
157 \LWR@traceinfo{lst@DeInit done}%
158 }
```

\lst@MakeCaption   {⟨*t/b*⟩}

This is called BOTH at the top and at the bottom of each listing.

Patched for lwarp.

```
159 \def\lst@MakeCaption#1{%
160 \LWR@traceinfo{lst@MakeCaption at #1}%
161   \lst@ifdisplaystyle
162 \LWR@traceinfo{lst@MakeCaption: making a listings display caption}%
163     \ifx #1t%
164         \ifx\lst@@caption\@empty\expandafter\lst@HRefStepCounter \else
165                             \expandafter\refstepcounter
166     \fi {lstlisting}%
167 % \LWR@traceinfo{About to assign label: !\lst@label!}%
168 %        \ifx\lst@label\@empty\else
169 % \label{\lst@label}\fi
170 % \LWR@traceinfo{Finished assigning the label.}%
171         \let\lst@arg\lst@intname \lst@ReplaceIn\lst@arg\lst@filenamerpl
172         \global\let\lst@name\lst@arg \global\let\lstname\lst@name
173         \lst@ifnolol\else
174             \ifx\lst@@caption\@empty
175                 \ifx\lst@caption\@empty
176                     \ifx\lst@intname\@empty
177                     \else
178                         \def\lst@temp{ }%
179                         \ifx\lst@intname\lst@temp \else
```

This code places a contents entry for a non-float. This would have to be modified for lwarp:

```
180 \LWR@traceinfo{lst@MakeCaption: addcontents lst@name: -\lst@name-}%
181 %                           \addcontentsline{lol}{lstlisting}{\lst@name}
182                     \fi
183                 \fi
184             \fi
185         \else
```

This would have to be modified for lwarp:

```
186 \LWR@traceinfo{lst@MakeCaption: addcontents lst@@caption: -\lst@@caption-}%
187                 \addcontentsline{lol}{lstlisting}%
188 {\protect\numberline{\thelstlisting}%
189 {\protect\ignorespaces \LWR@isolate{\lst@@caption} \protect\relax}}%
190             \fi
191         \fi
192     \fi
193    \ifx\lst@caption\@empty\else
194 \LWR@traceinfo{lst@MakeCaption: lst@caption not empty-}%
195       \lst@IfSubstring #1\lst@captionpos
196          {\begingroup
197 \LWR@traceinfo{lst@MakeCaption: at the selected position}%
```

These space and box commands are not needed for HTML output:

```
198 %             \let\@@vskip\vskip
199 %             \def\vskip{\afterassignment\lst@vskip \@tempskipa}%
200 %             \def\lst@vskip{\nobreak\@@vskip\@tempskipa\nobreak}%
201 %             \par\@parboxrestore\normalsize\normalfont % \noindent (AS)
202 %             \ifx #1t\allowbreak \fi
203             \ifx\lst@title\@empty
```

New lwarp code to create a caption:

```
204                 \LWR@stoppars%      lwarp
205             \lst@makecaption\fnum@lstlisting{\ignorespaces \lst@caption}
206         \else
```

New lwarp code to create a title:

```
207 %                 \lst@maketitle\lst@title % (AS)
208 \LWR@traceinfo{lst@MakeCaption: Making title: \lst@title}%
209 \begin{BlockClass}{lstlistingtitle}%     lwarp
210 \lst@maketitle\lst@title%                lwarp
211 \end{BlockClass}%                        lwarp
212             \fi%
213 \LWR@traceinfo{lst@MakeCaption: About to assign label: !\lst@label!}%
214         \ifx\lst@label\@empty\else%
215 \leavevmode% gets rid of bad space factor error
216 \GetTitleStringExpand{\lst@caption}%
217 \edef\LWR@lntemp{\GetTitleStringResult}%
218 \edef\@currentlabelname{\detokenize\expandafter{\LWR@lntemp}}%
219 \label{\lst@label}\fi%
220 \LWR@traceinfo{lst@MakeCaption: Finished assigning the label.}%
```

Not needed for lwarp:

```
221 %             \ifx #1b\allowbreak \fi
222             \endgroup}{}%
223     \fi
224 \LWR@traceinfo{lst@MakeCaption: end of making a listings display caption}%
225   \else
226 \LWR@traceinfo{lst@MakeCaption: INLINE}%
227   \fi
228 \LWR@traceinfo{lst@MakeCaption: done at #1}%
229 }
230
231 \renewcommand{\lst@maketitle}[1]{%
232     \LWR@isolate{#1}%
233 }%
234
```

line numbers   Patched to keep left line numbers outside of the left margin, and place right line
                numbers in a field \VerbatimHTMLWidth wide.

```
235 \lst@Key{numbers}{none}{%
236     \let\lst@PlaceNumber\@empty
237     \lstKV@SwitchCases{#1}%
238     {none:\\%
239      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.

```
240 %          \LWR@origllap{
241              \LWR@orignormalfont%
242              \lst@numberstyle{\thelstnumber}\kern\lst@numbersep%
243 %          }
244     }\\%
245      right:\def\lst@PlaceNumber{\LWR@origrlap{\LWR@orignormalfont
246               \kern 6in \kern\lst@numbersep
247               \lst@numberstyle{\thelstnumber}}}%
248     }{\PackageError{lwarp-listings}{Numbers #1 unknown}\@ehc}}
```

File 254   **`lwarp-listliketab.sty`**

§ 363   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 255   **`lwarp-lltjext.sty`**

§ 364   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 256   **lwarp-lltjp-siunitx.sty**

§ 365   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}% 2021-10-31, no date assigned in file

This is the siunitx v3 file, as patched by lltjp-siunitx.

```
2 \ExplSyntaxOn
3
4 \cs_set_protected:Npn \siunitx_print_text:n #1
5   {
6     \text
7       {
8         \ltj@allalchar % <--- LuaTeX-ja
9         \bool_if:NT \l__siunitx_print_text_family_bool
10           { \fontfamily { \familydefault } }
11         \bool_if:NT \l__siunitx_print_text_series_bool
12           { \fontseries { \seriesdefault } }
13         \bool_if:NT \l__siunitx_print_text_shape_bool
14           { \fontshape { \shapedefault } }
15         \bool_lazy_any:nT
16           {
```

```
17          { \l__siunitx_print_text_family_bool }
18          { \l__siunitx_print_text_series_bool }
19          { \l__siunitx_print_text_shape_bool }
20        }
21        { \selectfont }
22      \tl_use:N \l__siunitx_print_text_font_tl
23    \exp_args:NnV \tl_if_head_eq_meaning:nNTF {#1} \l_siunitx_unit_fraction_tl
24        {
25          \__siunitx_print_text_fraction:Nnn #1
26        }
27        {
28          \__siunitx_print_text_replace:n {#1}
29        }
30    }
31  }
32
33 \ExplSyntaxOff
```

File 257   `lwarp-lltjp-tascmac.sty`

§ 366   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 258   `lwarp-longtable.sty`

§ 367   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 259 **`lwarp-lpic.sty`**

§ 368 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 260 **`lwarp-lscape.sty`**

§ 369 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 261 **`lwarp-ltablex.sty`**

§ 370 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 262   **`lwarp-ltcaption.sty`**

§ 371   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 263   **`lwarp-ltxgrid.sty`**

§ 372   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 264 **`lwarp-ltxtable.sty`**

§ 373   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 265   **`lwarp-lua-check-hyphen.sty`**

§ 374   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 266   **`lwarp-lua-visual-debug.sty`**

§ 375   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 267   **`lwarp-luacolor.sty`**

§ 376   Package **luacolor**

luacolor (*Pkg*)   luacolor is ignored.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{luacolor}[2016/05/16]
```

```
2 \newcommand{\luacolorProcessBox}[1]{}
```

File 268 **`lwarp-luamplib.sty`**

§ 377   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 269 **`lwarp-luatexko.sty`**

§ 378   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 270 **`lwarp-luatodonotes.sty`**

§ 379 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 \renewcommand{\@todoarea}[3][]{%
73     \@todonotes@areaselectedtrue%
74     \@todocommon{#1}{#2}%
75     \todonotes@textmark@highlight{#3}%
76     \zref@label{@todonotes@\arabic{@todonotes@numberoftodonotes}@end}%
77 }%
78
79
80 \DeclareDocumentCommand{\todonotes@textmark@highlight}{m}{%
81 \InlineClass[background:\LWR@origpound{}B3FFB3]{highlight}{#1}%
82 }
83
84 \fi% \if@todonotes@disabled
```

File 271 **`lwarp-luavlna.sty`**

§ 380   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 272   **`lwarp-lyluatex.sty`**

§ 381   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.

CSS   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
CSS

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}[2022/11/07]

User-redefinable ALT tag:

4 \newcommand*{\LyluatexImageAltText}{-lilypond-~\PackageDiagramAltText}

\ly@compilescore   {⟨*Lilypond object*⟩}

5 \LetLtxMacro\LWR@orig@ly@compilescore\ly@compilescore
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 \xpatchcmd{\endly@bufferenv}
45      {\hspace{0pt}\\}
46      {}
47      {}
48      {\LWR@patcherror{lyluatex}{endly@bufferenv}}
```

File 273 **`lwarp-magaz.sty`**

§ 382   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 274 **`lwarp-makeidx.sty`**

§ 383　Package # makeidx

(*Emulates or patches code by LATEX PROJECT TEAM.*)

makeidx (*Pkg*)　　makeidx is patched for use by lwarp.

for HTML output:　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 275　**`lwarp-manyfoot.sty`**

§ 384　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.

for HTML output:　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 276   `lwarp-marginal.sty`

§ 385   Package   # marginal

marginal (*Pkg*)   marginal is ignored.

for HTML output: 1 `\LWR@ProvidesPackageDrop{marginal}`

2 `\newcommand*{\showlostmarginals}{}`
3 `\newcommand*{\enlargefreelist}{}`
4 `\newcommand*{\onesidemarginals}{}`

File 277   `lwarp-marginfit.sty`

§ 386   Package   # marginfit

marginfit (*Pkg*)   marginfit is ignored.

for HTML output: Discard all options for lwarp-marginfit:

1 `\LWR@ProvidesPackageDrop{marginfit}[2018/06/08]`

File 278   `lwarp-marginfix.sty`

§ 387   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 279   `lwarp-marginnote.sty`

§ 388   Package   # marginnote

(*Emulates or patches code by* MARKUS KOHM.)

marginnote (*Pkg*)   marginnote is emulated.

   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 280   `lwarp-marvosym.sty`

§ 389   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 281 **`lwarp-mathalpha.sty`**

§ 390   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 282   **`lwarp-mathastext.sty`**

§ 391   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 283   `lwarp-mathcomp.sty`

§ 392   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 284   **`lwarp-mathdesign.sty`**

§ 393   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 285   **`lwarp-mathdots.sty`**

§ 394   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 286   `lwarp-mathfixs.sty`

§ 395   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 287   `lwarp-mathpazo.sty`

§ 396   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 288   `lwarp-mathptmx.sty`

§ 397   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 289   `lwarp-mathspec.sty`

§ 398   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 290   **`lwarp-mathtools.sty`**

§ 399   Package   # mathtools

(*Emulates or patches code by* MORTEN HØGHOLM, LARS MADSEN.)

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:

- **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}.

for HTML output: 1 \LWR@ProvidesPackagePass{mathtools}[2018/01/08]

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{\newenvironment{crampedsubarray}[1]{}{}}
26
27 \CustomizeMathJax{\newcommand{\smashoperator}[2][]{#2\limits}}
28
29 \CustomizeMathJax{\newcommand{\SwapAboveDisplaySkip}{}}
30
31 \CustomizeMathJax{\newcommand{\LaTeXunderbrace}[1]{\underbrace{#1}}}
32 \CustomizeMathJax{\newcommand{\LaTeXoverbrace}[1]{\overbrace{#1}}}
33
34
35 \CustomizeMathJax{\newcommand{\LWRmultlined}[1][]{\begin{multline*}}}
36 \CustomizeMathJax{\newenvironment{multlined}[1][]{\LWRmultlined}{\end{multline*}}}
37
38 \CustomizeMathJax{\let\LWRorigshoveleft\shoveleft}
39 \CustomizeMathJax{\renewcommand{\shoveleft}[1][]{\LWRorigshoveleft}}
40 \CustomizeMathJax{\let\LWRorigshoveright\shoveright}
41 \CustomizeMathJax{\renewcommand{\shoveright}[1][]{\LWRorigshoveright}}
42
43 \CustomizeMathJax{\newcommand{\shortintertext}[1]{\text{#1}\notag \\}}
44
45 \LetLtxMacro\LWR@mathtools@orig@DeclarePairedDelimiter\DeclarePairedDelimiter
46 \renewcommand{\DeclarePairedDelimiter}[3]{
47     \LWR@mathtools@orig@DeclarePairedDelimiter{#1}{#2}{#3}
48 % starred:
49     \appto\LWR@customizedMathJax{\LWRbackslash(}
50     \appto\LWR@customizedMathJax{%
51      \LWRbackslash}newcommand\{\LWRbackslash\macrotocsname{#1}LWRsubstar\}%
52     }%
53     \appto\LWR@customizedMathJax{[2][]}%
```

```
54        \appto\LWR@customizedMathJax{\{\{}%
55        \LWR@subcustomizedmathjax{##1\left#2##2##1\right#3}%
56        \appto\LWR@customizedMathJax{\}\}}%
57        \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
58 % not starred:
59        \appto\LWR@customizedMathJax{\LWRbackslash(}
60        \appto\LWR@customizedMathJax{%
61          \LWRbackslash{}newcommand\{\LWRbackslash\macrotocsname{#1}LWRsubnostar\}%
62        }%
63        \appto\LWR@customizedMathJax{[2][]}%
64        \appto\LWR@customizedMathJax{\{\{}%
65        \LWR@subcustomizedmathjax{##1#2##2##1#3}%
66        \appto\LWR@customizedMathJax{\}\}}%
67        \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
68 % user macro:
69        \appto\LWR@customizedMathJax{\LWRbackslash(}
70        \appto\LWR@customizedMathJax{%
71            \LWRbackslash{}newcommand\{\LWRbackslash{}\macrotocsname{#1}\}%
72            \{\LWRbackslash{}ifstar%
73                \LWRbackslash{}\macrotocsname{#1}LWRsubstar%
74                \LWRbackslash{}\macrotocsname{#1}LWRsubnostar%
75            \}%
76        }%
77        \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
78 }
79 \@onlypreamble\DeclareParedDelimiter
80
81 % (DeclarePairedDelimiterX is already defined to use \DeclarePairedDelimiterXPP.)
82
83 \LetLtxMacro\LWR@mathtools@orig@DeclarePairedDelimiterXPP\DeclarePairedDelimiterXPP
84 \DeclareDocumentCommand{\DeclarePairedDelimiterXPP}{m O{1} m m m m m}{
85    \LWR@mathtools@orig@DeclarePairedDelimiterXPP{#1}[#2]{#3}{#4}{#5}{#6}{#7}
86 % subsubstar, second opt arg
87        \appto\LWR@customizedMathJax{\LWRbackslash(}%
88        \appto\LWR@customizedMathJax{%
89          \LWRbackslash{}newcommand\{\LWRbackslash\macrotocsname{#1}LWRsubsubstar\}%
90        }%
91        \appto\LWR@customizedMathJax{[#2]}%
92        \appto\LWR@customizedMathJax{\{\{\LWRbackslash{}left}%
93        \LWR@subcustomizedmathjax{#3#4#7}%
94        \appto\LWR@customizedMathJax{\LWRbackslash{}right}%
95        \LWR@subcustomizedmathjax{#5#6}%
96        \appto\LWR@customizedMathJax{\}\}}%
97        \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
98 % substar, first opt arg
99        \appto\LWR@customizedMathJax{\LWRbackslash(}%
100       \appto\LWR@customizedMathJax{%
101         \LWRbackslash{}newcommand\{\LWRbackslash\macrotocsname{#1}LWRsubstar\}[1][]%
102       }%
103       \appto\LWR@customizedMathJax{%
104           \{
105           \LWRbackslash{}def\LWRbackslash{}delimsize\{\#1\}
106           \LWRbackslash\macrotocsname{#1}LWRsubsubstar
107           \}%
108       }%
109       \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
110 % subsubnostar, second opt arg
111       \appto\LWR@customizedMathJax{\LWRbackslash(}%
112       \appto\LWR@customizedMathJax{%
113         \LWRbackslash{}newcommand\{\LWRbackslash\macrotocsname{#1}LWRsubsubnostar\}%
```

```
114    }%
115    \appto\LWR@customizedMathJax{[#2]}%
116    \appto\LWR@customizedMathJax{\{\{\LWRbackslash{}delimsize}%
117    \LWR@subcustomizedmathjax{#3#4#7}%
118    \appto\LWR@customizedMathJax{\LWRbackslash{}delimsize}%
119    \LWR@subcustomizedmathjax{#5#6}%
120    \appto\LWR@customizedMathJax{\}\}}%
121    \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
122 % subnostar, first opt arg
123    \appto\LWR@customizedMathJax{\LWRbackslash(}%
124    \appto\LWR@customizedMathJax{%
125     \LWRbackslash{}newcommand\{\LWRbackslash\macrotocsname{#1}LWRsubnostar\}[1][]%
126    }%
127    \appto\LWR@customizedMathJax{%
128       \{
129       \LWRbackslash{}def\LWRbackslash{}delimsize\{\#1\}
130       \LWRbackslash\macrotocsname{#1}LWRsubsubnostar
131       \}%
132    }%
133    \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
134 % user macro:
135    \appto\LWR@customizedMathJax{\LWRbackslash(}
136    \appto\LWR@customizedMathJax{%
137       \LWRbackslash{}newcommand\{%
138          \LWRbackslash{}\macrotocsname{#1}%
139       \}%
140          \{\LWRbackslash{}ifstar%
141             \LWRbackslash{}\macrotocsname{#1}LWRsubstar%
142             \LWRbackslash{}\macrotocsname{#1}LWRsubnostar%
143          \}%
144    }%
145    \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
146 }
147 \@onlypreamble\DeclareParedDelimiterXPP
148 \@onlypreamble\DeclareParedDelimiterX
149
150 \CustomizeMathJax{\newcommand{\vcentcolon}{\mathrel{\unicode{x2236}}}}
151
152 \LetLtxMacro\LWR@mathtools@orig@newgathered\newgathered
153 \renewcommand{\newgathered}[4]{%
154    \LWR@mathtools@orig@newgathered{#1}{#2}{#3}{#4}%
155    \appto\LWR@customizedMathJax{\LWRbackslash(}%
156    \LWR@subcustomizedmathjax{%
157       \newenvironment{#1}{\begin{gathered}}{\end{gathered}}%
158    }%
159    \appto\LWR@customizedMathJax{\LWRbackslash)}%
160 }
161 \@onlypreamble\newgathered
162
163 \end{warpMathJax}
```

File 291   **`lwarp-mattens.sty`**

§ 400   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 }}

66 \CustomizeMathJax{\newcommand{\bSb}{%
67      \let\LWRmattenscross\LWRmattensnull%
68      \let\LWRmattensovercmd\overline%
69      \let\LWRmattensundercmd\underline%
70      \LWRmattens%
71 }}

73 \CustomizeMathJax{\newcommand{\aCSa}{%
74      \let\LWRmattenscross\tilde%
75      \let\LWRmattensovercmd\overrightarrow%
76      \let\LWRmattensundercmd\underrightarrow%
77      \LWRmattens%
78 }}

80 \CustomizeMathJax{\newcommand{\bCSb}{%
81      \let\LWRmattenscross\tilde%
82      \let\LWRmattensovercmd\overline%
83      \let\LWRmattensundercmd\underline%
84      \LWRmattens%
85 }}
86 \end{warpMathJax}
```

File 292   **`lwarp-maybemath.sty`**

§ 401   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 293   **lwarp-mcaption.sty**

§ 402   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 294   **lwarp-mdframed.sty**

§ 403   Package   # mdframed

(*Emulates or patches code by* MARCO DANIEL, ELKE SCHUBERT.)

mdframed (*Pkg*)   mdframed is loaded with options forced to framemethod=none.

§ 403.1 **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 selec-
tion 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.

§ 403.2 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}
```

§ 403.3 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
```

§ 403.4 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 }
```

§ 403.5   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 }
```

§ 403.6   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 }
```

§ 403.7 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 }
```

§ 403.8 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
```

§ 403.9 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 295   **`lwarp-mdwmath.sty`**

§ 404   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 296 **`lwarp-media9.sty`**

§ 405   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 297   **lwarp-memhfixc.sty**

§ 406   Package   # memhfixc

memhfixc (*Pkg*)   memhfixc is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{memhfixc}[2013/05/30]`

File 298   **lwarp-menukeys.sty**

§ 407   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 \xpatchcmd{\tw@define@menu@macro@}
3     {\@nameuse{tw@style@#4@pre}}
4     {%
5         \begin{lateximage}*[\detokenize{##2}]%
6         \@nameuse{tw@style@#4@pre}%
7     }
8     {}
9     {\LWR@patcherror{menukeys}{tw@define@menu@macro@}}
10
11 \xpatchcmd{\tw@define@menu@macro@}
12     {\@nameuse{tw@style@#4@post}}
13     {%
14         \@nameuse{tw@style@#4@post}%
15         \end{lateximage}%
16     }
```

```
17    {}
18    {\LWR@patcherror{menukeys}{tw@define@menu@macro@ B}}
```

Patch the existing macros:

```
19 \renewmenumacro{\menu}[>]{menus}
20 \renewmenumacro{\directory}[/]{paths}
21 \renewmenumacro{\keys}[+]{roundedkeys}
```

File 299   `lwarp-metalogo.sty`

§ 408   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 300   `lwarp-metalogox.sty`

§ 409   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 301   **`lwarp-mhchem.sty`**

§ 410   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}[2018/06/22]

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     \ifbool{LWR@xfakebold}%
6         {\def\LWR@tempone{Y}}%
7         {\def\LWR@tempone{N}}%
8     \begin{lateximage}%
9         *%
10        [%
11            \textbackslash{}%
12            ce%
13            \{\LWR@HTMLsanitizedetokenized{\detokenize{#1}}\}%
14        ]%
15        *%
16        [%
17            FM\LWR@f@family%
18            SR\LWR@f@series%
19            SH\LWR@f@shape%
20            SHC\LWR@f@shapecaps%
21            CL\LWR@tempcolor%
22            FB\LWR@tempone% xfakebold
23        ]%
24     \LWR@setcurrentfont%
25     \LWR@mhchem@origce{#1}%
26     \end{lateximage}%
27     \endgroup%
28     \addtocounter{LWR@mhchem@cedepth}{-1}%
29 }
```

Only set math shift if outer depth:

```
30 \newcounter{LWR@mhchem@cedepth}
31 \setcounter{LWR@mhchem@cedepth}{0}
```

The new \ce. Sets math shift then continues.

```
32 \renewcommand{\ce}{%
33     \begingroup%
34     \ifnumequal{\value{LWR@mhchem@cedepth}}{0}{%
35         \catcode'\$=3% math shift
36     }{}%
37     \addtocounter{LWR@mhchem@cedepth}{1}%
38     \LWR@mhchem@HTML@ce%
39 }
```

The original definition of \cesplit:

```
40 \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.

```
41 \newcommand*{\LWR@mhchem@HTML@cesplit}[2]
42 {%
43     \LWR@findcurrenttextcolor% sets \LWR@tempcolor
```

```
44    \ifbool{LWR@xfakebold}%
45        {\def\LWR@tempone{Y}}%
46        {\def\LWR@tempone{N}}%
47    \begin{lateximage}%
48        *%
49        [%
50            \textbackslash{}%
51            cesplit%
52            \{\LWR@HTMLsanitizedetokenized{\detokenize{#2}}\}%
53        ]%
54        *%
55        [%
56            FM\LWR@f@family%
57            SR\LWR@f@series%
58            SH\LWR@f@shape%
59            SHC\LWR@f@shapecaps%
60            CL\LWR@tempcolor%
61            FB\LWR@tempone% xfakebold
62        ]%
63    \LWR@setcurrentfont%
64    \LWR@mhchem@origcesplit{#1}{#2}%
65    \end{lateximage}%
66    \endgroup%
67 }
```

Only set math shift if outer depth:

```
68 \newcounter{LWR@mhchem@cesplitdepth}
69 \setcounter{LWR@mhchem@cesplitdepth}{0}
```

The new \cesplit. Sets math shift then continues.

```
70 \renewcommand{\cesplit}{%
71     \begingroup%
72     \ifnumequal{\value{LWR@mhchem@cesplitdepth}}{0}{%
73         \catcode`\$=3% math shift
74     }{}%
75     \addtocounter{LWR@mhchem@cesplitdepth}{1}%
76     \LWR@mhchem@HTML@cesplit%
77 }
```

Resore originals inside a lateximage:

```
78 \appto\LWR@restoreorigformatting{%
79 \LetLtxMacro\ce\LWR@mhchem@origce%
80 \LetLtxMacro\cesplit\LWR@mhchem@origcesplit%
81 }
82
83 \begin{warpMathJax}
84 \CustomizeMathJax{\require{mhchem}}
85 \end{warpMathJax}
```

File 302 **`lwarp-microtype.sty`**

§ 411   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 303   **`lwarp-midfloat.sty`**

§ 412   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 304   **`lwarp-midpage.sty`**

§ 413   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 305   **`lwarp-minibox.sty`**

§ 414   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 306   **lwarp-minitoc.sty**

§ 415   Package   # minitoc

minitoc (*Pkg*)   minitoc is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{minitoc}[2018/07/12]`

mtcoff disables minitoc.

`2 \usepackage{mtcoff}`

File 307   **lwarp-minted.sty**

§ 416   Package   # minted

(*Emulates or patches code by* Geoffrey M. Poore.)

minted (*Pkg*)   minted is patched for use by lwarp.

⚠ limitations   mathescape and `highlightlines` don't work. 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}[2022/12/12]`

```
2 \renewcommand{\setminted}[2][]{%
3   \ifthenelse{\equal{#1}{}}%
4     {\setkeys{minted@opt@g}{%
5         #2,
6       mathescape=false,breaklines,texcomments=false,highlightlines={}% lwarp
7     }}%
8     {\minted@configlang{#1}%
9       \setkeys{minted@opt@lang}{%
10         #2,
11       mathescape=false,breaklines,texcomments=false,highlightlines={}% lwarp
12     }}}
13
14 \renewcommand{\setmintedinline}[2][]{%
15   \ifthenelse{\equal{#1}{}}%
16     {\setkeys{minted@opt@g@i}{%
17         #2,
18       mathescape=false,breaklines,texcomments=false,highlightlines={}% lwarp
19     }}%
20     {\minted@configlang{#1}%
21       \setkeys{minted@opt@lang@i}{%
22         #2,
23       mathescape=false,breaklines,texcomments=false,highlightlines={}% lwarp
```

```
24     }}}
25
26 \xpatchcmd{\RobustMintInlineProcess}
27     {\setkeys{minted@opt@cmd}{#1}}
28     {%
29         \setkeys{minted@opt@cmd}{%
30             #1,%
31             mathescape=false,breaklines,texcomments=false,highlightlines={}%
32         }%
33     }
34     {}
35     {\LWR@patcherror{minted}{minted}}
36
37 \xpatchcmd{\RobustMintProcess}
38     {\setkeys{minted@opt@cmd}{#1}}
39     {%
40         \setkeys{minted@opt@cmd}{%
41             #1,%
42             mathescape=false,breaklines,texcomments=false,highlightlines={}%
43         }%
44     }
45     {}
46     {\LWR@patcherror{minted}{minted}}
47
48 \xpatchcmd{\minted}
49     {\setkeys{minted@opt@cmd}{#1}}
50     {%
51         \setkeys{minted@opt@cmd}{%
52             #1,%
53             mathescape=false,breaklines,texcomments=false,highlightlines={}%
54         }%
55     }
56     {}
57     {\LWR@patcherror{minted}{minted}}
58
59 \xpatchcmd{\inputminted}
60     {\setkeys{minted@opt@cmd}{#1}}
61     {\setkeys{minted@opt@cmd}{%
62             #1,%
63             mathescape=false,breaklines,texcomments=false,highlightlines={}%
64         }%
65     }
66     {}
67     {\LWR@patcherror{minted}{inputminted}}
68
69
70
71 %\xpatchcmd{\mintinline}
72 %     {\setkeys{minted@opt@cmd}{#1}}
73 %     {\setkeys{minted@opt@cmd}{%
74 %             #1,%
75 %             mathescape=false,breaklines,texcomments=false,highlightlines={}%
76 %         }%
77 %     }
78 %     {}
79 %     {\LWR@patcherror{minted}{mintinline}}
80
81 %\xpatchcmd{\mint}
82 %     {\setkeys{minted@opt@cmd}{#1}}
83 %     {%
```

```
84 %           \setkeys{minted@opt@cmd}{%
85 %               #1,%
86 %              mathescape=false,breaklines,texcomments=false,highlightlines={}%
87 %           }%
88 %       }
89 %       {}
90 %     {\LWR@patcherror{minted}{mint}}
91
92 \renewenvironment{minted@snugshade*}[1]%
93 {%
94      \colorlet{shadecolor}{#1}%
95      \begin{snugshade*}%
96 }
97 {%
98      \end{snugshade*}%
99 }
```

File 308   **`lwarp-mismath.sty`**

§ 417   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.

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}[2022/11/11]

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{\newcommand{\mathup}[1]{\mathrm{#1}}}
28 \CustomizeMathJax{\newcommand{\e}{\mathrm{e}}}
29 \CustomizeMathJax{\newcommand{\i}{\mathrm{i}}}
30 \CustomizeMathJax{\newcommand{\j}{\mathrm{j}}}
31
32 \CustomizeMathJax{\newcommand{\boldvect}{}}
33 \CustomizeMathJax{\newcommand{\arrowvect}{}}
34 \CustomizeMathJax{\newcommand{\pinumber}[1][]{}}
35 \CustomizeMathJax{\newcommand{\hvect}[1]{\vec{\vphantom{h}#1}}}
36 \CustomizeMathJax{\newcommand{\hvec}[1]{\vec{\vphantom{t}#1}}}
37 \CustomizeMathJax{%
38     \newcommand{\norm}[1]{\left\vert\left\vert#1\right\vert\right\vert}
39 }
40 \CustomizeMathJax{\newcommand{\di}{\mathop{}\!\mathrm{d}}}
41
42 \CustomizeMathJax{\newcommand{\P}{\operatorname{\probastyle{P}}}}
43 \CustomizeMathJax{\newcommand{\E}{\operatorname{\probastyle{E}}}}
44 \CustomizeMathJax{\newcommand{\V}{\operatorname{\probastyle{V}}}}
45 \CustomizeMathJax{\newcommand{\Par}{\unicode{x00B6}}}
46
47 \CustomizeMathJax{\DeclareMathOperator{\adj}{adj}}
48 \CustomizeMathJax{\DeclareMathOperator{\Aut}{Aut}}
49 \CustomizeMathJax{\DeclareMathOperator{\Conv}{Conv}}
50 \CustomizeMathJax{\DeclareMathOperator{\cov}{cov}}
51 \CustomizeMathJax{\DeclareMathOperator{\Cov}{Cov}}
52 \CustomizeMathJax{\newcommand{\curl}{\operatorname{\vect{\mathrm{curl}}}}}
53 \CustomizeMathJax{\DeclareMathOperator{\divg}{div}}
54 \CustomizeMathJax{\DeclareMathOperator{\End}{End}}
55
56 \CustomizeMathJax{\DeclareMathOperator{\erf}{erf}}
57 \CustomizeMathJax{\newcommand{\grad}{\operatorname{\vect{\mathrm{grad}}}}}
58 \CustomizeMathJax{\DeclareMathOperator{\id}{id}}
59 \CustomizeMathJax{\DeclareMathOperator{\Id}{Id}}
60 \CustomizeMathJax{\DeclareMathOperator{\im}{im}}
61 \CustomizeMathJax{\let\oldIm\Im}
62 \CustomizeMathJax{\renewcommand{\Im}{\operatorname{Im}}}
63 \CustomizeMathJax{\DeclareMathOperator{\lb}{lb}}
64 \CustomizeMathJax{\DeclareMathOperator{\lcm}{lcm}}
65
```

```
66 \CustomizeMathJax{\DeclareMathOperator{\rank}{rank}}
67 \CustomizeMathJax{\let\oldRe\Re}
68 \CustomizeMathJax{\renewcommand{\Re}{\operatorname{Re}}}
69 \CustomizeMathJax{\newcommand{\rot}{\operatorname{\vect{\mathrm{rot}}}}}
70 \CustomizeMathJax{\DeclareMathOperator{\sgn}{sgn}}
71 \CustomizeMathJax{\DeclareMathOperator{\spa}{span}}
72 \CustomizeMathJax{\DeclareMathOperator{\tr}{tr}}
73 \CustomizeMathJax{\DeclareMathOperator{\Var}{Var}}
74 \CustomizeMathJax{\DeclareMathOperator{\Zu}{Z}}
75
76 \CustomizeMathJax{\DeclareMathOperator{\arccot}{arccot}}
77 \CustomizeMathJax{\DeclareMathOperator{\sech}{sech}}
78 \CustomizeMathJax{\DeclareMathOperator{\csch}{csch}}
79 \CustomizeMathJax{\DeclareMathOperator{\arsinh}{arsinh}}
80 \CustomizeMathJax{\DeclareMathOperator{\arcosh}{arcosh}}
81 \CustomizeMathJax{\DeclareMathOperator{\artanh}{artanh}}
82 \CustomizeMathJax{\DeclareMathOperator{\arcoth}{arcoth}}
83 \CustomizeMathJax{\DeclareMathOperator{\arsech}{arsech}}
84 \CustomizeMathJax{\DeclareMathOperator{\arcsch}{arcsch}}
85
86 \CustomizeMathJax{\DeclareMathOperator{\bigO}{\mathcal{O}}}
87 \CustomizeMathJax{\DeclareMathOperator{\bigo}{O}}
88 \CustomizeMathJax{\DeclareMathOperator{\lito}{o}}
89
90 \CustomizeMathJax{\newcommand{\R}{\mathset{R}}}
91 \CustomizeMathJax{\newcommand{\C}{\mathset{C}}}
92 \CustomizeMathJax{\newcommand{\N}{\mathset{N}}}
93 \CustomizeMathJax{\newcommand{\Z}{\mathset{Z}}}
94 \CustomizeMathJax{\newcommand{\Q}{\mathset{Q}}}
95 \CustomizeMathJax{\newcommand{\F}{\mathset{F}}}
96 \CustomizeMathJax{\newcommand{\K}{\mathset{K}}}
97
98 \CustomizeMathJax{\newcommand{\ds}{\displaystyle}}
99 \CustomizeMathJax{\newcommand{\dlim}{\lim\limits}}
100 \CustomizeMathJax{\newcommand{\dsum}{\sum\limits}}
101 \CustomizeMathJax{\newcommand{\dprod}{\prod\limits}}
102 \CustomizeMathJax{\newcommand{\dcup}{\bigcup\limits}}
103 \CustomizeMathJax{\newcommand{\dcap}{\bigcap\limits}}
104 \CustomizeMathJax{\newcommand{\lbar}{\overline}}
105 \CustomizeMathJax{\newcommand{\hlbar}[1]{\overline{\vphantom{h}#1}}}
106 \CustomizeMathJax{\newcommand{\eqdef}{\stackrel{\mathrm{def}}{=}}}
107 \CustomizeMathJax{\newcommand{\unbr}{\underbrace}}
108 \CustomizeMathJax{\newcommand{\iif}{if and only if }}
109
110 \CustomizeMathJax{\newcommand{\mul}{\mathord{\times}}}
111 \CustomizeMathJax{\newcommand{\then}{\ \Longrightarrow \ \mbox{} }}
112 \CustomizeMathJax{\newcommand{\txt}[1]{\quad\text{#1}\quad}}
113 \CustomizeMathJax{\newcommand{\pow}[2]{\left( #1 \right)^{\!#2}}}
114 \CustomizeMathJax{\newcommand{\abs}[1]{\left\vert#1\right\vert}}
115 \CustomizeMathJax{\newcommand{\lfrac}[2]{\frac{\:#1\:}{\:#2\:}}}
116
117 \CustomizeMathJax{\newenvironment{system}[1][l]%
118     {\left\{\begin{array}{@{.15em}#1@{}}}
119     {\end{array}\right.}
120 }
121
122 \CustomizeMathJax{\newenvironment{spmatrix}
123     {\left(\begin{smallmatrix}}
124     {\end{smallmatrix}\right)}
125 }
```

```
126
127 \CustomizeMathJax{%
128     \newenvironment{mathcols}
129         {\begin{aligned}\displaystyle}
130         {\end{aligned}}
131 }
132 \CustomizeMathJax{\newcommand{\changecol}{\end{aligned}\qquad\begin{aligned}}}
```

User-adjustable settings, detected if in the preamble.

```
133 \AtBeginDocument{
134 \ifdef{\itpi}{
135     \CustomizeMathJax{\let\itpi\pi}
136 }{}
137 \ifdefstring{\boldvectcommand}{\mathbf}{
138     \CustomizeMathJax{\newcommand{\boldvectcommand}[1]{\mathbf{#1}}}
139 }{
140     \CustomizeMathJax{\newcommand{\boldvectcommand}[1]{\boldsymbol{#1}}}
141 }
142 \ifbool{arrowvect}{
143     \CustomizeMathJax{\newcommand{\vect}[1]{\overrightarrow{#1}}}
144 }{
145     \CustomizeMathJax{\newcommand{\vect}[1]{\boldvectcommand{#1}}}
146 }
147 \ifdefstring{\probastyle}{\mathbb}{
148     \CustomizeMathJax{\newcommand{\probastyle}[1]{\mathbb{#1}}}
149 }{
150     \CustomizeMathJax{\newcommand{\probastyle}[1]{\mathrm{#1}}}
151 }
152 \ifdefstring{\mathset}{\mathbb}{
153     \CustomizeMathJax{\newcommand{\mathset}[1]{\mathbb{#1}}}
154 }{
155     \CustomizeMathJax{\newcommand{\mathset}[1]{\mathbf{#1}}}
156 }
157 }
158 \end{warpMathJax}
```

File 309   `lwarp-mleftright.sty`

§ 418   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 310 **`lwarp-morefloats.sty`**

§ 419 Package **morefloats**

morefloats (*Pkg*)   morefloats is ignored.

for HTML output: 1 `\LWR@ProvidesPackageDrop{morefloats}[2015/07/22]`

File 311 **`lwarp-moreverb.sty`**

§ 420 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 312   `lwarp-movie15.sty`

§ 421   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/…

```
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 313   `lwarp-mparhack.sty`

§ 422   Package   # mparhack

mparhack (*Pkg*)   mparhack is ignored.

Discard all options for lwarp-mparhack:

```
1 \LWR@ProvidesPackageDrop{mparhack}[2005/04/17]
```

File 314   `lwarp-multibib.sty`

§ 423   Package   # multibib

(*Emulates or patches code by* THORSTEN HANSEN.)

multibib (*Pkg*)   multibib is patched for use by lwarp.

```
1 \LWR@ProvidesPackagePass{multibib}[2008/12/10]
```

```
2 \xpatchcmd{\newcites}
3    {{\@suffix}}
4    {{\@suffix_html}}
5    {}
6    {\LWR@patcherror{multibib}{newcites}}
```

File 315 **`lwarp-multicap.sty`**

§ 424   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 316   **`lwarp-multicol.sty`**

§ 425   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 317   `lwarp-multicolrule.sty`

§ 426   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 318   `lwarp-multimedia.sty`

§ 427   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 319   **`lwarp-multiobjective.sty`**

§ 428   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 320 **`lwarp-multirow.sty`**

§ 429   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

- See section 429.2 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.

⚠ skipped cells

⚠ empty 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 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.

In a lateximage, the print versions are restored.

See section 75.24 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}{}
```

§ 429.1   **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}
```

§ 429.2 **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 321 **`lwarp-multitoc.sty`**

§ 430 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 322 **`lwarp-musicography.sty`**

§ 431 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}[2019/05/28]`

```
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}{ m m }{%
36 \begin{lateximage}*[#1/#2][#1#2]*%
37 \musStack{#1 #2}\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@meterCThreeTwo}{}{%
68 \begin{lateximage}*[C3/2]*%
69 \meterCplus{\musStack{3 2}}%
70 \end{lateximage}%
71 }
72
73 \LWR@formatted{meterCThreeTwo}
74
75 \NewDocumentCommand{\LWR@HTML@meterO}{}{\HTMLunicode{25EF}}
76
```

```
77 \LWR@formatted{meterO}

78

79 \newcommand{\LWR@null@noFig}[1][]{}%

80

81 \NewDocumentCommand{\LWR@HTML@musFig}{ m }{%
82 \begin{lateximage}*[%
83     {% ALT text for copy/paste
84         \LetLtxMacro\noFig\LWR@null@noFig%
85         \LetLtxMacro\musSharp\LWR@HTML@musSharp%
86         \LetLtxMacro\musDoubleSharp\LWR@HTML@musDoubleSharp%
87         \LetLtxMacro\musFlat\LWR@HTML@musFlat%
88         \LetLtxMacro\musDoubleFlat\LWR@HTML@musDoubleFlat%
89         \LetLtxMacro\musNatural\LWR@HTML@musNatural%
90         {#1}% braces here because \noFig uses []
91     }%
92 ]*%
93     \musStack[\musFigFont]{#1}%
94 \end{lateximage}%
95 }

96

97 \LWR@formatted{musFig}

98

99 \NewDocumentCommand{\LWR@HTML@musFlat}        {}{\HTMLunicode{266D}}
100 \NewDocumentCommand{\LWR@HTML@musDoubleFlat} {}{\HTMLunicode{1D12B}}
101 \NewDocumentCommand{\LWR@HTML@musSharp}       {}{\HTMLunicode{266F}}
102 \NewDocumentCommand{\LWR@HTML@musDoubleSharp}{}{\HTMLunicode{1D12A}}
103 \NewDocumentCommand{\LWR@HTML@musNatural}     {}{\HTMLunicode{266E}}

104

105 \LWR@formatted{musFlat}
106 \LWR@formatted{musDoubleFlat}
107 \LWR@formatted{musSharp}
108 \LWR@formatted{musDoubleSharp}
109 \LWR@formatted{musNatural}

110

111 \NewDocumentCommand{\LWR@HTML@musWhole}        {}{\HTMLunicode{1D15D}}
112 \NewDocumentCommand{\LWR@HTML@musHalf}         {}{\HTMLunicode{1D15E}}
113 \NewDocumentCommand{\LWR@HTML@musQuarter}      {}{\HTMLunicode{1D15F}}
114 \NewDocumentCommand{\LWR@HTML@musEighth}       {}{\HTMLunicode{1D160}}
115 \NewDocumentCommand{\LWR@HTML@musSixteenth}    {}{\HTMLunicode{1D161}}
116 \NewDocumentCommand{\LWR@HTML@musThirtySecond} {}{\HTMLunicode{1D162}}
117 \NewDocumentCommand{\LWR@HTML@musSixtyFourth}  {}{\HTMLunicode{1D163}}

118

119 \LWR@formatted{musWhole}
120 \LWR@formatted{musHalf}
121 \LWR@formatted{musQuarter}
122 \LWR@formatted{musEighth}
123 \LWR@formatted{musSixteenth}
124 \LWR@formatted{musThirtySecond}
125 \LWR@formatted{musSixtyFourth}

126

127 \NewDocumentCommand{\LWR@HTML@musWholeDotted}{}
128     {\HTMLunicode{1D15D}\HTMLunicode{1D16D}}
129 \NewDocumentCommand{\LWR@HTML@musHalfDotted}{}
130     {\HTMLunicode{1D15E}\HTMLunicode{1D16D}}
131 \NewDocumentCommand{\LWR@HTML@musQuarterDotted}{}
132     {\HTMLunicode{1D15F}\HTMLunicode{1D16D}}
133 \NewDocumentCommand{\LWR@HTML@musEighthDotted}{}
134     {\HTMLunicode{1D160}\HTMLunicode{1D16D}}
135 \NewDocumentCommand{\LWR@HTML@musSixteenthDotted}{}
136     {\HTMLunicode{1D161}\HTMLunicode{1D16D}}
```

```
137 \NewDocumentCommand{\LWR@HTML@musThirtySecondDotted}{}
138     {\HTMLunicode{1D162}\HTMLunicode{1D16D}}
139 \NewDocumentCommand{\LWR@HTML@musSixtyFourthDotted}{}
140     {\HTMLunicode{1D163}\HTMLunicode{1D16D}}
141
142 \LWR@formatted{musWholeDotted}
143 \LWR@formatted{musHalfDotted}
144 \LWR@formatted{musQuarterDotted}
145 \LWR@formatted{musEighthDotted}
146 \LWR@formatted{musSixteenthDotted}
147 \LWR@formatted{musThirtySecondDotted}
148 \LWR@formatted{musSixtyFourthDotted}
```

File 323   **`lwarp-mwe.sty`**

§ 432   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 324   **`lwarp-nameauth.sty`**

§ 433   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}[2017/03/22]`

\@nameauth@Hook (*Hook*)   lwarp formatting is inserted.
[nameauth]

```
2 \renewcommand*\@nameauth@Hook[1]
3 {%
4     \if@nameauth@Lock
5         \@nameauth@InHooktrue%
6         \protected@edef\test{#1}%
7         \expandafter\@nameauth@TestDot\expandafter{\test}%
8         \if@nameauth@InAKA
9             \if@nameauth@AlwaysFormat
10                 \@nameauth@FirstFormattrue%
```

```
11        \else
12          \unless\if@nameauth@AKAFormat
13          \@nameauth@FirstFormatfalse\fi
14        \fi
15        \if@nameauth@MainFormat
16          \if@nameauth@FirstFormat
17            \bgroup\NamesFormat{%
18              \LWR@textcurrentcolor{\LWR@textcurrentfont{#1}}%      lwarp
19            }\egroup%
20          \else
21            \bgroup\MainNameHook{%
22              \LWR@textcurrentcolor{\LWR@textcurrentfont{#1}}%      lwarp
23            }\egroup%
24          \fi
25        \else
26          \if@nameauth@FirstFormat
27            \bgroup\FrontNamesFormat{%
28              \LWR@textcurrentcolor{\LWR@textcurrentfont{#1}}%      lwarp
29            }\egroup%
30          \else
31            \bgroup\FrontNameHook{%
32              \LWR@textcurrentcolor{\LWR@textcurrentfont{#1}}%      lwarp
33            }\egroup%
34          \fi
35        \fi
36      \else
37        \if@nameauth@AlwaysFormat
38          \@nameauth@FirstFormattrue%
39        \fi
40        \if@nameauth@MainFormat
41          \if@nameauth@FirstFormat
42            \bgroup\NamesFormat{%
43              \LWR@textcurrentcolor{\LWR@textcurrentfont{#1}}%      lwarp
44            }\egroup%
45          \else
46            \bgroup\MainNameHook{%
47              \LWR@textcurrentcolor{\LWR@textcurrentfont{#1}}%      lwarp
48            }\egroup%
49          \fi
50        \else
51          \if@nameauth@FirstFormat
52            \bgroup\FrontNamesFormat{%
53              \LWR@textcurrentcolor{\LWR@textcurrentfont{#1}}%      lwarp
54            }\egroup%
55          \else
56            \bgroup\FrontNameHook{%
57              \LWR@textcurrentcolor{\LWR@textcurrentfont{#1}}%      lwarp
58            }\egroup%
59          \fi
60        \fi
61      \fi
62      \@nameauth@FirstFormatfalse%
63      \@nameauth@InHookfalse%
64    \fi
65 }
```

File 325 **`lwarp-nameref.sty`**

§ 434   Package # nameref

nameref (*Pkg*)   nameref is emulated by lwarp.

for HTML output: Discard all options for lwarp-nameref:

```
1 \PackageInfo{lwarp}{%
2 Using the lwarp HTML version of package 'nameref',\MessageBreak
3 and discarding options.\MessageBreak
4 (Not using \protect\ProvidesPackage, so that other packages\MessageBreak
5 do not attempt to patch lwarp's version of 'nameref'.)\MessageBreak
6 }
7 \DeclareOption*{}
8 \ProcessOptions\relax
```

File 326   **`lwarp-natbib.sty`**

§ 435   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 327 **`lwarp-nccfancyhdr.sty`**

§ 436   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 328 **`lwarp-nccfoots.sty`**

§ 437 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 329 **`lwarp-nccmath.sty`**

§ 438 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 330   **`lwarp-needspace.sty`**

§ 439   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 331   **`lwarp-newpxmath.sty`**

§ 440   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 332   **`lwarp-newtxmath.sty`**

§ 441   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 333 **`lwarp-newtxsf.sty`**

§ 442   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 334   **`lwarp-nextpage.sty`**

§ 443   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 335 **`lwarp-nfssext-cfr.sty`**

§ 444 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 \newrobustcmd{\LWR@HTML@textsu}[1]{#1}
183 % \newrobustcmd{\LWR@HTML@textsi}[1]{#1}
184 \newrobustcmd{\LWR@HTML@textdf}[1]{#1}


185 \ifdef{\LWR@HTML@textsw}{}{% duplicated by fontaxes
186     \newrobustcmd{\LWR@HTML@textsw}[1]{#1}
187     \LWR@formatted{textsw}
188 }
189
190 \newrobustcmd{\LWR@HTML@textti}[1]{#1}
191 \newrobustcmd{\LWR@HTML@textlt}[1]{#1}
192 \newrobustcmd{\LWR@HTML@textof}[1]{#1}
193 \newrobustcmd{\LWR@HTML@textalt}[1]{#1}
194 \newrobustcmd{\LWR@HTML@textreg}[1]{#1}
195 \newrobustcmd{\LWR@HTML@emboss}[1]{#1}
196 \newrobustcmd{\LWR@HTML@textorn}[1]{#1}
197 \newrobustcmd{\LWR@HTML@textqt}[1]{#1}
198 \newrobustcmd{\LWR@HTML@textsh}[1]{#1}
199 \newrobustcmd{\LWR@HTML@texttm}[1]{\texttt{#1}}
200 \newrobustcmd{\LWR@HTML@texttv}[1]{\texttt{#1}}
201 \newrobustcmd{\LWR@HTML@textl}[1]{\InlineClass{textln}{#1}}
202 \newrobustcmd{\LWR@HTML@texto}[1]{\textsc{#1}}
203 \newrobustcmd{\LWR@HTML@textp}[1]{\InlineClass{textp}{#1}}
204 \newrobustcmd{\LWR@HTML@textt}[1]{\InlineClass{textt}{#1}}
205 \newrobustcmd{\LWR@HTML@textpl}[1]{#1}
206 \newrobustcmd{\LWR@HTML@textpo}[1]{\textsc{#1}}
207 \newrobustcmd{\LWR@HTML@texttl}[1]{\InlineClass{textln}{#1}}
208 \newrobustcmd{\LWR@HTML@textto}[1]{\textsc{#1}}
209 \newrobustcmd{\LWR@HTML@textol}[1]{#1}
210 \newrobustcmd{\LWR@HTML@textswash}[1]{#1}
211 \newrobustcmd{\LWR@HTML@textu}[1]{#1}
212 \newrobustcmd{\LWR@HTML@textscu}[1]{#1}
213 \newrobustcmd{\LWR@HTML@textui}[1]{\LWR@HTML@textit{#1}}
214 \newrobustcmd{\LWR@HTML@textri}[1]{#1}
215 \newrobustcmd{\LWR@HTML@textnw}[1]{#1}
216 \newrobustcmd{\LWR@HTML@textcd}[1]{#1}
217 \newrobustcmd{\LWR@HTML@textec}[1]{#1}
218 \newrobustcmd{\LWR@HTML@textuc}[1]{#1}
219 \newrobustcmd{\LWR@HTML@textet}[1]{#1}
220 \newrobustcmd{\LWR@HTML@textep}[1]{#1}
221 \newrobustcmd{\LWR@HTML@textex}[1]{#1}
```

```
222 \newrobustcmd{\LWR@HTML@textux}[1]{#1}
223 \newrobustcmd{\LWR@HTML@textrw}[1]{#1}
224 \newrobustcmd{\LWR@HTML@textmb}[1]{{\LWR@HTML@mbweight\InlineClass{textmb}{#1}}}
225 \newrobustcmd{\LWR@HTML@textdb}[1]{{\LWR@HTML@dbweight\InlineClass{textdb}{#1}}}
226 \newrobustcmd{\LWR@HTML@textsb}[1]{{\LWR@HTML@sbweight\InlineClass{textsb}{#1}}}
227 % \newrobustcmd{\LWR@HTML@texteb}[1]]{#1}
228 \newrobustcmd{\LWR@HTML@textub}[1]{{\LWR@HTML@ubweight\InlineClass{textub}{#1}}}
229 % \newrobustcmd{\LWR@HTML@textlg}[1]]{#1}
230 \newrobustcmd{\LWR@HTML@textel}[1]{{\LWR@HTML@elweight\InlineClass{textel}{#1}}}
231 \newrobustcmd{\LWR@HTML@textul}[1]{{\LWR@HTML@ulweight\InlineClass{textul}{#1}}}
232
233 \LWR@formatted{textln}
234 \LWR@formatted{textos}
235 \LWR@formatted{textin}
236 \LWR@formatted{textsu}
237 % \LWR@formatted{textsi}
238 \LWR@formatted{textdf}
239 \LWR@formatted{textti}
240 \LWR@formatted{textlt}
241 \LWR@formatted{textof}
242 \LWR@formatted{textalt}
243 \LWR@formatted{textreg}
244 \LWR@formatted{emboss}
245 \LWR@formatted{textorn}
246 \LWR@formatted{textqt}
247 \LWR@formatted{textsh}
248 \LWR@formatted{texttm}
249 \LWR@formatted{texttv}
250 \LWR@formatted{textl}
251 \LWR@formatted{texto}
252 \LWR@formatted{textp}
253 \LWR@formatted{textt}
254 \LWR@formatted{textpl}
255 \LWR@formatted{textpo}
256 \LWR@formatted{texttl}
257 \LWR@formatted{textto}
258 \LWR@formatted{textol}
259 \LWR@formatted{textswash}
260 \LWR@formatted{textu}
261 \LWR@formatted{textscu}
262 \LWR@formatted{textui}
263 \LWR@formatted{textri}
264 \LWR@formatted{textnw}
265 \LWR@formatted{textcd}
266 \LWR@formatted{textec}
267 \LWR@formatted{textuc}
268 \LWR@formatted{textet}
269 \LWR@formatted{textep}
270 \LWR@formatted{textex}
271 \LWR@formatted{textux}
272 \LWR@formatted{textrw}
273 \LWR@formatted{textmb}
274 \LWR@formatted{textdb}
275 \LWR@formatted{textsb}
276 % \LWR@formatted{texteb}
277 \LWR@formatted{textub}
278 % \LWR@formatted{textlg}
279 \LWR@formatted{textel}
280 \LWR@formatted{textul}
281
```

```
282 \FilenameNullify{%
283     \LetLtxMacro\textln\@firstofone%
284     \LetLtxMacro\textos\@firstofone%
285     \LetLtxMacro\textin\@firstofone%
286     \LetLtxMacro\textsu\@firstofone%
287 %   \LetLtxMacro\textsi\@firstofone%
288     \LetLtxMacro\textdf\@firstofone%
289     \LetLtxMacro\textsw\@firstofone%
290     \LetLtxMacro\textti\@firstofone%
291     \LetLtxMacro\textlt\@firstofone%
292     \LetLtxMacro\textof\@firstofone%
293     \LetLtxMacro\textalt\@firstofone%
294     \LetLtxMacro\textreg\@firstofone%
295     \LetLtxMacro\emboss\@firstofone%
296     \LetLtxMacro\textorn\@firstofone%
297     \LetLtxMacro\textqt\@firstofone%
298     \LetLtxMacro\textsh\@firstofone%
299     \LetLtxMacro\texttm\@firstofone%
300     \LetLtxMacro\texttv\@firstofone%
301     \LetLtxMacro\textl\@firstofone%
302     \LetLtxMacro\texto\@firstofone%
303     \LetLtxMacro\textp\@firstofone%
304     \LetLtxMacro\textt\@firstofone%
305     \LetLtxMacro\textpl\@firstofone%
306     \LetLtxMacro\textpo\@firstofone%
307     \LetLtxMacro\texttl\@firstofone%
308     \LetLtxMacro\textto\@firstofone%
309     \LetLtxMacro\textol\@firstofone%
310     \LetLtxMacro\textswash\@firstofone%
311     \LetLtxMacro\textu\@firstofone%
312     \LetLtxMacro\textscu\@firstofone%
313     \LetLtxMacro\textui\@firstofone%
314     \LetLtxMacro\textri\@firstofone%
315     \LetLtxMacro\textnw\@firstofone%
316     \LetLtxMacro\textcd\@firstofone%
317     \LetLtxMacro\textec\@firstofone%
318     \LetLtxMacro\textuc\@firstofone%
319     \LetLtxMacro\textet\@firstofone%
320     \LetLtxMacro\textep\@firstofone%
321     \LetLtxMacro\textex\@firstofone%
322     \LetLtxMacro\textux\@firstofone%
323     \LetLtxMacro\textrw\@firstofone%
324     \LetLtxMacro\textmb\@firstofone%
325     \LetLtxMacro\textdb\@firstofone%
326     \LetLtxMacro\textsb\@firstofone%
327 %   \LetLtxMacro\texteb\@firstofone%
328     \LetLtxMacro\textub\@firstofone%
329 %   \LetLtxMacro\textlg\@firstofone%
330     \LetLtxMacro\textel\@firstofone%
331     \LetLtxMacro\textul\@firstofone%
332 }
333
334 \providecommand*{\zeroslash}{0}
335 \newrobustcmd*{\LWR@HTML@zeroslash}{0}
336 \LWR@formatted{zeroslash}
```

File 336 **`lwarp-nicefrac.sty`**

§ 445 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 337 **`lwarp-niceframe.sty`**

§ 446 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 338 **`lwarp-nicematrix.sty`**

§ 447   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 339   **`lwarp-noitcrul.sty`**

§ 448   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 340   **`lwarp-nolbreaks.sty`**

§ 449   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 341   **`lwarp-nomencl.sty`**

§ 450   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}[2005/09/22]`

\BaseJobname is added to the label in case xr or xr-hyper are used.

```
2 \def\@@@nomenclature[#1]#2#3{%
3 \def\@tempa{#2}\def\@tempb{#3}%
4 \protected@write\@nomenclaturefile{}%
5  {\string\nomenclatureentry{#1\nom@verb\@tempa @[{\nom@verb\@tempa}]%
6       \begingroup\nom@verb\@tempb\protect\nomeqref{\theequation}%
```

```
7         |nompageref}{\theLWR@previousautopagelabel}}%   lwarp
8 \endgroup
9 \@esphack}
10
11 \renewcommand*{\pagedeclaration}[1]{, \nameref{\BaseJobname-autopage-#1}}%
```

File 342 **`lwarp-nonfloat.sty`**

§ 451 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 343 **`lwarp-nonumonpart.sty`**

§ 452 Package # nonumonpart

nonumonpart (*Pkg*) nonumonpart is ignored.

for HTML output: 1 `\LWR@ProvidesPackageDrop{nonumonpart}[2011/04/15]`

File 344 **`lwarp-nopageno.sty`**

§ 453 Package # nopageno

nopageno (*Pkg*) nopageno is ignored.

for HTML output: 1 `\LWR@ProvidesPackageDrop{nopageno}[1989/01/01]`

File 345 **`lwarp-notes.sty`**

§ 454 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 346   `lwarp-notespages.sty`

§ 455   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 347   `lwarp-nowidow.sty`

§ 456   Package   # nowidow

(*Emulates or patches code by* RAPHAËL PINSON.)

nowidow (*Pkg*)   nowidow is ignored.

for HTML output: Discard all options for lwarp-nowidow. \LWR@ProvidesPackageDrop{nowidow}[2011/09/20]

\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 348   **lwarp-ntheorem.sty**

§ 457   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.

§ 457.1 **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.

§ 457.2 **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]
```

§ 457.3   Remembering the theorem style

Storage for the style being used for new theorems.

```
36 \newcommand{\LWR@newtheoremstyle}{plain}
```

```
37 \AtBeginDocument{
38 \IfPackageLoadedTF{cleveref}{
39 \gdef\@thm#1#2#3{%
40    \if@thmmarks
41      \stepcounter{end\InTheoType ctr}%
42    \fi
43    \renewcommand{\InTheoType}{#1}%
44    \if@thmmarks
45      \stepcounter{curr#1ctr}%
46      \setcounter{end#1ctr}{0}%
47    \fi
48    \refstepcounter[#1]{#2}%  <<< cleveref modification
49    \theorem@prework
50     \LWR@forcenewpage% lwrap
51     \LWR@printpendingfootnotes%                        lwrap
52     \BlockClass{theorembody#1}%\LWR@thisthmstyle% lwrap
53    \trivlist % latex's \trivlist, calling latex's \@trivlist unchanged
54    \ifuse@newframeskips  % cf. latex.ltx for topsepadd: \@trivlist
55      \ifthm@inframe
```

```
56        \thm@topsep\theoreminframepreskipamount
57        \thm@topsepadd\theoreminframepostskipamount
58       \else
59        \thm@topsep\theorempreskipamount
60        \thm@topsepadd\theorempostskipamount
61       \fi
62     \else% oldframeskips
63        \thm@topsep\theorempreskipamount
64        \thm@topsepadd \theorempostskipamount
65        \ifvmode\advance\thm@topsepadd\partopsep\fi
66     \fi
67     \@topsep\thm@topsep
68     \@topsepadd\thm@topsepadd
69     \advance\linewidth -\theorem@indent
70     \advance\linewidth -\theorem@rightindent
71     \advance\@totalleftmargin \theorem@indent
72     \parshape \@ne \@totalleftmargin \linewidth
73     \@ifnextchar[{\@ythm{#1}{#2}{#3}}{\@xthm{#1}{#2}{#3}}
74 }
75 }{% not @ifpackageloaded{cleveref}
76 \gdef\@thm#1#2#3{%
77     \if@thmmarks
78       \stepcounter{end\InTheoType ctr}%
79     \fi
80     \renewcommand{\InTheoType}{#1}%
81     \if@thmmarks
82       \stepcounter{curr#1ctr}%
83       \setcounter{end#1ctr}{0}%
84     \fi
85     \refstepcounter{#2}%
86     \theorem@prework
87       \LWR@forcenewpage% lwarp


88       \LWR@printpendingfootnotes%                    lwarp


89      \BlockClass{theorembody#1}%\LWR@thisthmstyle% lwarp
90     \trivlist % latex's \trivlist, calling latex's \@trivlist unchanged
91     \ifuse@newframeskips  % cf. latex.ltx for topsepadd: \@trivlist
92       \ifthm@inframe
93         \thm@topsep\theoreminframepreskipamount
94         \thm@topsepadd\theoreminframepostskipamount
95       \else
96         \thm@topsep\theorempreskipamount
97         \thm@topsepadd\theorempostskipamount
98       \fi
99     \else% oldframeskips
100        \thm@topsep\theorempreskipamount
101        \thm@topsepadd \theorempostskipamount
102        \ifvmode\advance\thm@topsepadd\partopsep\fi
103     \fi
104     \@topsep\thm@topsep
105     \@topsepadd\thm@topsepadd
106     \advance\linewidth -\theorem@indent
107     \advance\linewidth -\theorem@rightindent
108     \advance\@totalleftmargin \theorem@indent
109     \parshape \@ne \@totalleftmargin \linewidth
110     \@ifnextchar[{\@ythm{#1}{#2}{#3}}{\@xthm{#1}{#2}{#3}}
111 }
112 }
```

```
113 }% AtBeginDocument
```

Patched to remember the style being used for new theorems:

```
114 \gdef\theoremstyle#1{%
115   \@ifundefined{th@#1}{\@warning
116       {Unknown theoremstyle '#1'. Using 'plain'}%
117       \theorem@style{plain}
118         \renewcommand{\LWR@newtheoremstyle}{plain}% lwarp
119         }%
120    {
121      \theorem@style{#1}
122      \renewcommand{\LWR@newtheoremstyle}{#1}% lwarp
123      }
124 }
```

Patched to remember the style for this theorem type, and set it later when the environment is started.

```
125
126 \gdef\@xnthm#1#2[#3]{%
127   \ifthm@tempif
128     \csedef{LWR@thmstyle#1}{\LWR@newtheoremstyle}% lwarp
129     \expandafter\@ifundefined{c@#1}%
130       {\@definecounter{#1}}{}%
131     \@newctr{#1}[#3]%
132     \expandafter\xdef\csname the#1\endcsname{%
133       \expandafter\noexpand\csname the#3\endcsname \@thmcountersep
134         {\noexpand\csname\the\theoremnumbering\endcsname{#1}}}%
135     \expandafter\gdef\csname mkheader@#1\endcsname
136       {\csname setparms@#1\endcsname
137       \@thm{#1}{#1}{#2}
138       }%
139     \global\@namedef{end#1}{\@endtheorem}
140    \AtBeginEnvironment{#1}{\edef\LWR@thisthmstyle{\@nameuse{LWR@thmstyle#1}}}% lwarp
141  \fi
142 }
143
144 \gdef\@ynthm#1#2{%
145   \ifthm@tempif
146     \csedef{LWR@thmstyle#1}{\LWR@newtheoremstyle}% lwarp
147     \expandafter\@ifundefined{c@#1}%
148       {\@definecounter{#1}}{}%
149     \expandafter\xdef\csname the#1\endcsname
150       {\noexpand\csname\the\theoremnumbering\endcsname{#1}}%
151     \expandafter\gdef\csname mkheader@#1\endcsname
152       {\csname setparms@#1\endcsname
153       \@thm{#1}{#1}{#2}
154       }%
155     \global\@namedef{end#1}{\@endtheorem}
156    \AtBeginEnvironment{#1}{\edef\LWR@thisthmstyle{\@nameuse{LWR@thmstyle#1}}}% lwarp
157  \fi
158 }
159
160 \gdef\@othm#1[#2]#3{%
161   \@ifundefined{c@#2}{\@nocounterr{#2}}%
162    {\ifthm@tempif
163      \csedef{LWR@thmstyle#1}{\LWR@newtheoremstyle}% lwarp
164      \global\@namedef{the#1}{\@nameuse{the#2}}%
165      \expandafter\protected@xdef\csname num@addtheoremline#1\endcsname{%
```

```
166              \noexpand\@num@addtheoremline{#1}{#3}}%
167    \expandafter\protected@xdef\csname nonum@addtheoremline#1\endcsname{%
168              \noexpand\@nonum@addtheoremline{#1}{#3}}%
169    \theoremkeyword{#3}%
170    \expandafter\protected@xdef\csname #1Keyword\endcsname
171          {\the\theoremkeyword}%
172    \expandafter\gdef\csname mkheader@#1\endcsname
173      {\csname setparms@#1\endcsname
174              \@thm{#1}{#2}{#3}
175        }%
176    \global\@namedef{end#1}{\@endtheorem}
177    \AtBeginEnvironment{#1}{\edef\LWR@thisthmstyle{\@nameuse{LWR@thmstyle#1}}}% lwarp
178  \fi}
179 }
```

§ 457.4 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.

```
180 \newcommand{\LWR@inctheorem}{%
181 \addtocounter{LWR@thisautoid}{1}%
182 \LWR@stoppars%
183 \LWR@htmltag{%
184    a id=\textquotedbl\LWR@print@mbox{autoid-\arabic{LWR@thisautoid}}\textquotedbl%
185 }%
186 \LWR@htmltag{/a}\LWR@orignewline%
187 \LWR@startpars%
188 }
```

§ 457.5 \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>.

```
189 \gdef\newtheoremstyle#1#2#3{%
190   \expandafter\@ifundefined{th@#1}%
191   {\expandafter\gdef\csname th@#1\endcsname{%
192     \def\@begintheorem####1####2{%
193     \LWR@inctheorem% lwarp
194     #2}%
195     \def\@opargbegintheorem####1####2####3{%
196     \LWR@inctheorem% lwarp
197     #3}%
198 }%
199 }%
200 {\PackageError{\basename}{Theorem style #1 already defined}\@eha}
201 }
```

§ 457.6 **Standard styles**

```
202 \renewtheoremstyle{plain}%
203  {\item[
204    \InlineClass{theoremheaderplain}{##1\ ##2\theorem@separator}]}%
205  {\item[
206    \InlineClass{theoremheaderplain}{##1\ ##2\ (##3)\theorem@separator}]}
207
208 \renewtheoremstyle{break}%
209  {\item[
210    \InlineClass{theoremheaderbreak}{##1\ ##2\theorem@separator}\newline
211    ]}%
212  {\item[
213    \InlineClass{theoremheaderbreak}%
214       {##1\ ##2\ (##3)\theorem@separator}\newline
215    ]}
216
217 \renewtheoremstyle{change}%
218  {\item[
219    \InlineClass{theoremheaderchange}{##2\ ##1\theorem@separator}]}%
220  {\item[
221    \InlineClass{theoremheaderchange}{##2\ ##1\ (##3)\theorem@separator}]}
222
223 \renewtheoremstyle{changebreak}%
224  {\item[
225       \InlineClass{theoremheaderchangebreak}%
226          {##2\ ##1\theorem@separator}\newline
227    ]}%
228  {\item[
229       \InlineClass{theoremheaderchangebreak}%
230          {##2\ ##1\ (##3)\theorem@separator}\newline
231    ]}
232
233 \renewtheoremstyle{margin}%
234  {\item[
235       \InlineClass{theoremheadermargin}{##2 \qquad ##1\theorem@separator}
236    ]}%
237  {\item[
238     \InlineClass{theoremheadermargin}{##2 \qquad ##1\ (##3)\theorem@separator}
239    ]}
240
241 \renewtheoremstyle{marginbreak}%
242  {\item[
243    \InlineClass{theoremheadermarginbreak}%
244       {##2 \qquad ##1\theorem@separator}\newline
245    ]}%
246  {\item[
247    \InlineClass{theoremheadermarginbreak}%
248       {##2 \qquad ##1\ (##3)\theorem@separator}\newline
249    ]}
250
251 \renewtheoremstyle{nonumberplain}%
252  {\item[
253    \InlineClass{theoremheaderplain}{##1\theorem@separator}]}%
254  {\item[
255    \InlineClass{theoremheaderplain}{##1\ (##3)\theorem@separator}]}
256
257 \renewtheoremstyle{nonumberbreak}%
258  {\item[
259    \InlineClass{theoremheaderbreak}{##1\theorem@separator}\newline
```

```
260    ]}%
261  {\item[
262    \InlineClass{theoremheaderbreak}{##1\ (##3)\theorem@separator}\newline
263    ]}
264
265 \renewtheoremstyle{empty}%
266  {\item[]}%
267  {\item[
268    \InlineClass{theoremheaderplain}{##3}]}
269
270 \renewtheoremstyle{emptybreak}%
271  {\item[]}%
272  {\item[
273    \InlineClass{theoremheaderplain}{##3}] \ \newline}
```

§ 457.7 Additional objects

The following manually adjust the css for the standard configuration objects which
are not a purely plain style:

```
274 \ifbool{LWR@ntheoremamsthm}{}{%
```

Upright text via CSS:

```
275    \newtheoremstyle{plainupright}%
276    {\item[
277      \InlineClass{theoremheaderplain}{##1\ ##2\theorem@separator}]}%
278    {\item[
279      \InlineClass{theoremheaderplain}{##1\ ##2\ (##3)\theorem@separator}]}
```

Upright text and small caps header via CSS:

```
280    \newtheoremstyle{nonumberplainuprightsc}%
281    {\item[
282      \InlineClass{theoremheadersc}{##1\theorem@separator}]}%
283    {\item[
284      \InlineClass{theoremheadersc}{##1\ (##3)\theorem@separator}]}
285 }% not amsthm
```

§ 457.8 Renewed standard configuration

The following standard configuration is renewed using the new css:

```
286 \ifbool{LWR@ntheoremamsthm}{}{%
```

```
287 \ifx\thm@usestd\@undefined
288 \else
289    \theoremnumbering{arabic}
290    \theoremstyle{plain}
291    \RequirePackage{latexsym}
292    \theoremsymbol{\Box}
293    \theorembodyfont{\itshape}
294    \theoremheaderfont{\normalfont\bfseries}
295    \theoremseparator{}
296    \renewtheorem{Theorem}{Theorem}
297    \renewtheorem{theorem}{Theorem}
298    \renewtheorem{Satz}{Satz}
299    \renewtheorem{satz}{Satz}
```

```
300     \renewtheorem{Proposition}{Proposition}
301     \renewtheorem{proposition}{Proposition}
302     \renewtheorem{Lemma}{Lemma}
303     \renewtheorem{lemma}{Lemma}
304     \renewtheorem{Korollar}{Korollar}
305     \renewtheorem{korollar}{Korollar}
306     \renewtheorem{Corollary}{Corollary}
307     \renewtheorem{corollary}{Corollary}
308
309     \theoremstyle{plainupright}
310     \theorembodyfont{\upshape}
311     \theoremsymbol{\HTMLunicode{25A1}}% UTF-8 white box
312     \renewtheorem{Example}{Example}
313     \renewtheorem{example}{Example}
314     \renewtheorem{Beispiel}{Beispiel}
315     \renewtheorem{beispiel}{Beispiel}
316     \renewtheorem{Bemerkung}{Bemerkung}
317     \renewtheorem{bemerkung}{Bemerkung}
318     \renewtheorem{Anmerkung}{Anmerkung}
319     \renewtheorem{anmerkung}{Anmerkung}
320     \renewtheorem{Remark}{Remark}
321     \renewtheorem{remark}{Remark}
322     \renewtheorem{Definition}{Definition}
323     \renewtheorem{definition}{Definition}
324
325     \theoremstyle{nonumberplainuprightsc}
326     \theoremsymbol{\HTMLunicode{220E}}% UTF-8 end-of-proof
327     \renewtheorem{Proof}{Proof}
328     \renewtheorem{proof}{Proof}
329     \renewtheorem{Beweis}{Beweis}
330     \renewtheorem{beweis}{Beweis}
331     \qedsymbol{\HTMLunicode{220E}}% UTF-8 end-of-proof
332
333     \theoremsymbol{}
334 \fi
335 }% not amsthm
```

§ 457.9   **amsthm option**

Only if the amsthm option was given:

```
336 \ifbool{LWR@ntheoremamsthm}{
337
338 \gdef\th@plain{%
339   \def\theorem@headerfont{\normalfont\bfseries}\itshape%
340   \def\@begintheorem##1##2{%
341       \LWR@inctheorem% lwarp
342     \item[
343 \InlineClass{theoremheaderplain}{##1\ ##2.}
344       ]}%
345   \def\@opargbegintheorem##1##2##3{%
346       \LWR@inctheorem% lwarp
347     \item[
348 \InlineClass{theoremheaderplain}{##1\ ##2\ (##3).}
349       ]}}
350
351 \gdef\th@nonumberplain{%
352   \def\theorem@headerfont{\normalfont\bfseries}\itshape%
353   \def\@begintheorem##1##2{%
354       \LWR@inctheorem% lwarp
```

```
355        \item[
356    \InlineClass{theoremheaderplain}{##1.}
357          ]}%
358    \def\@opargbegintheorem##1##2##3{%
359          \LWR@inctheorem% lwarp
360        \item[
361    \InlineClass{theoremheaderplain}{##1\ (##3).}
362          ]}}
363
364 \gdef\th@definition{%
365    \def\theorem@headerfont{\normalfont\bfseries}\normalfont%
366    \def\@begintheorem##1##2{%
367          \LWR@inctheorem% lwarp
368        \item[
369    \InlineClass{theoremheaderdefinition}{##1\ ##2.}
370          ]}%
371    \def\@opargbegintheorem##1##2##3{%
372          \LWR@inctheorem% lwarp
373        \item[
374    \InlineClass{theoremheaderdefinition}{##1\ ##2\ (##3).}
375          ]}}
376
377 \gdef\th@nonumberdefinition{%
378    \def\theorem@headerfont{\normalfont\bfseries}\normalfont%
379    \def\@begintheorem##1##2{%
380          \LWR@inctheorem% lwarp
381        \item[
382    \InlineClass{theoremheaderdefinition}{##1.}
383          ]}%
384    \def\@opargbegintheorem##1##2##3{%
385          \LWR@inctheorem% lwarp
386        \item[
387    \InlineClass{theoremheaderdefinition}{##1\ (##3).}
388          ]}}
389
390 \gdef\th@remark{%
391    \def\theorem@headerfont{\itshape}\normalfont%
392    \def\@begintheorem##1##2{%
393          \LWR@inctheorem% lwarp
394        \item[
395    \InlineClass{theoremheaderremark}{##1\ ##2.}
396          ]}%
397    \def\@opargbegintheorem##1##2##3{%
398          \LWR@inctheorem% lwarp
399        \item[
400    \InlineClass{theoremheaderremark}{##1\ ##2\ (##3).}
401          ]}}
402
403 \gdef\th@nonumberremark{%
404    \def\theorem@headerfont{\itshape}\normalfont%
405    \def\@begintheorem##1##2{%
406          \LWR@inctheorem% lwarp
407        \item[
408    \InlineClass{theoremheaderremark}{##1.}
409          ]}%
410    \def\@opargbegintheorem##1##2##3{%
411          \LWR@inctheorem% lwarp
412        \item[
413    \InlineClass{theoremheaderremark}{##1\ (##3).}
414          ]}}
```

```
415
416 \gdef\th@proof{%
417    \def\theorem@headerfont{\normalfont\bfseries}\itshape%
418    \def\@begintheorem##1##2{%
419        \LWR@inctheorem% lwarp
420      \item[
421 \InlineClass{theoremheaderproof}{##1.}
422        ]}%
423    \def\@opargbegintheorem##1##2##3{%
424        \LWR@inctheorem% lwarp
425      \item[
426 \InlineClass{theoremheaderproof}{##1\ (##3).}
427        ]}}
428
429
430
431 \newcounter{proof}%
432 \if@thmmarks
433    \newcounter{currproofctr}%
434    \newcounter{endproofctr}%
435 \fi
436
437 \gdef\proofSymbol{\openbox}
438
439 \newcommand{\proofname}{Proof}
440
441 \newenvironment{proof}[1][\proofname]{
442    \th@proof
443    \def\theorem@headerfont{\itshape}%
444    \normalfont
445    \theoremsymbol{\HTMLunicode{220E}}% UTF-8 end-of-proof
446    \@thm{proof}{proof}{#1}
447 }%
448 {\@endtheorem}
449
450 }{}% amsthm option
```

§ 457.10   **Ending a theorem**

Patched for css:

```
451 \let\LWR@origendtheorem\@endtheorem
452 \renewcommand{\@endtheorem}{%
453 \ifbool{LWR@ntheoremmarks}{%
454    \ifsetendmark%
455    \InlineClass{theoremendmark}{\csname\InTheoType Symbol\endcsname}%
456    \setendmarkfalse%
457    \fi%
458 }{}%
459 \LWR@origendtheorem% also does \@endtrivlist
460 \ifbool{LWR@ntheoremmarks}{\global\setendmarktrue}{}%


461    \LWR@printpendingfootnotes%                            lwarp


462 \endBlockClass%
463 }
```

§ 457.11 **\NoEndMark**

464 \gdef\NoEndMark{\global\setendmarkfalse}

§ 457.12 **List-of**

Redefined to reuse the float mechanism to add list-of-theorem links:

\thm@thmline {⟨*1: printed type*⟩} {⟨*2: #*⟩} {⟨*3: optional*⟩} {⟨*4: page*⟩}

465 \renewcommand{\thm@@thmline@noname}[4]{%
466 \hypertocfloat{1}{theorem}{thm}{#2 #3}{}%
467 }
468
469 \renewcommand{\thm@@thmline@name}[4]{%
470 \hypertocfloat{1}{theorem}{thm}{#1 #2 #3}{}%
471 }

This was redefined by ntheorem when loaded, so it is now redefined for lwarp:

472 \def\thm@@thmline{\thm@@thmline@name}

Patch for css:

473 \def\listtheorems#1{
474 \LWR@htmlelementclass{nav}{lothm}%
475 \begingroup
476 \c@tocdepth=-2%
477 \def\thm@list{#1}\thm@processlist
478 \endgroup
479 \LWR@htmlelementclassend{nav}{lothm}%
480 }

§ 457.13 **Symbols**

Proof QED symbol:

481 \newcommand{\qed}{\qquad\the\qedsymbol}
482
483 \AtBeginDocument{
484 \@ifundefined{LWR@orig@openbox}{
485 \LetLtxMacro\LWR@orig@openbox\openbox
486 \LetLtxMacro\LWR@orig@blacksquare\blacksquare
487 \LetLtxMacro\LWR@orig@Box\Box
488
489 \def\openbox{\text{\HTMLunicode{25A1}}}% UTF-8 white box
490 \def\blacksquare{\text{\HTMLunicode{220E}}}% UTF-8 end-of-proof
491 \def\Box{\text{\HTMLunicode{25A1}}}% UTF-8 white box
492
493 \appto\LWR@restoreorigformatting{%
494 \LetLtxMacro\openbox\LWR@orig@openbox%
495 \LetLtxMacro\blacksquare\LWR@orig@blacksquare%
496 \LetLtxMacro\Box\LWR@orig@Box%
497 }% appto
498 }{}% @ifundefined
499 }% AtBeginDocument

§ 457.14 **Cross-referencing**

`\thref {⟨label⟩}`

```
500 \newcommand*{\thref}[1]{\cref{#1}}%
```

File 349 **`lwarp-octave.sty`**

§ 458 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 \RenewDocumentCommand{\@PrintTicks}{ m }{%
3 \kern-1pt%  lwarp
4 \@TickNum = #1%
5 \loop
6 \@Tick{}%
7 \advance\@TickNum by -1
8 \ifnum\@TickNum > 0
9 \repeat
10 }
```

Use unicode for the prime character:

```
11 \RenewDocumentCommand{\@Tick}{}{\HTMLunicode{2032}}
```

Catch the inline font:

```
12 \RenewDocumentCommand{\pitch}{ m o m }{%
13 \if@OctaveNumber%
14 {%
15     \pitchfont{%
16         \LWR@textcurrentfont{%  lwarp
17             \MakeUppercase{#1}%
18             \IfValueTF{#2}{#2}{}\textsubscript{#3}%
19         }%
20     }%
21 }%
22 \else%
23 {%
24     \pitchfont{%
25         \LWR@textcurrentfont{%  lwarp
26             \@GetOctaveTick{#1}[#2]{#3}%
27         }%
28     }%
29 }%
30 \fi%
31 }
```

The original was hard to adapt to lwarp's handling of &.

```
32 \StartDefiningTabulars
33 \renewcommand{\octavetable}{%
34 \begin{tabular}{ll}
35 \octaveprimes \pitch{C}{0} & \octavenumbers \pitch{C}{0} \\
36 \octaveprimes \pitch{C}{1} & \octavenumbers \pitch{C}{1} \\
37 \octaveprimes \pitch{C}{2} & \octavenumbers \pitch{C}{2} \\
38 \octaveprimes \pitch{C}{3} & \octavenumbers \pitch{C}{3} \\
39 \octaveprimes \pitch{C}{4} & \octavenumbers \pitch{C}{4} \\
40 \octaveprimes \pitch{C}{5} & \octavenumbers \pitch{C}{5} \\
41 \octaveprimes \pitch{C}{6} & \octavenumbers \pitch{C}{6} \\
42 \octaveprimes \pitch{C}{7} & \octavenumbers \pitch{C}{7} \\
43 \end{tabular}
44 }
45 \StopDefiningTabulars
```

File 350 **`lwarp-orcidlink.sty`**

§ 459   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}[2020/11/21]

4 \renewcommand\orcidlink[1]{%
5     \texorpdfstring%
6         {%
7             \href%
8                 {https://orcid.org/#1}%
9                 {%
10                     \begin{lateximage}[orcid #1]%    lwarp
11                     \mbox{%
12                         \scalerel*{%
13                             \begin{tikzpicture}[yscale=-1,transform shape]
14                             \pic{orcidlogo};
15                             \end{tikzpicture}
16                         }{|}%
17                     }%
18                     \end{lateximage}%    lwarp
19                 }%
20         }%
21         {}%
22 }
23
24 \begin{warpMathJax}
25 \CustomizeMathJax{\newcommand{\orcidlink}[1]{}}
26 \end{warpMathJax}
```

File 351 **`lwarp-overpic.sty`**

§ 460   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 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.

See section 88.2 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 352   **`lwarp-pagegrid.sty`**

§ 461   Package   # pagegrid

pagegrid (*Pkg*)   pagegrid is ignored.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{pagegrid}[2016/05/16]
```

```
2 \newcommand*{\pagegridsetup}[1]{}
```

File 353   **`lwarp-pagenote.sty`**

§ 462   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 354 **lwarp-pagesel.sty**

§ 463   Package   **pagesel**

pagesel (*Pkg*)   **pagesel** is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{pagesel}[2016/05/16]`

File 355 **lwarp-paralist.sty**

§ 464   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 356 **`lwarp-parallel.sty`**

§ 465 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 357   **`lwarp-parcolumns.sty`**

§ 466   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
33         \colplacechunks%
34         \end{LWR@setvirtualpage}%
35     }
36
37 \newcommand{\LWR@parcolumns@onecol}[1]{%
38     \ifbool{LWR@parcolumns@started}%
39         {}%
40         {%
41             \LWR@htmldivclass{parcolumns}%
42             \booltrue{LWR@parcolumns@started}%
43         }%
44     \ifboolexpr{%
45         bool {LWR@parcolumns@rule} and
46         test {%
47             \ifnumgreater
48                 {\value{LWR@parcolumns@thiscol}}
49                 {1}
50         }%
51     }%
52         {\renewcommand{\LWR@tempone}{ ; border-left: 2px solid black}}%
53         {\renewcommand{\LWR@tempone}{}}%
54     \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 358 **`lwarp-parnotes.sty`**

§ 467   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 \long\def\PN@parnote@real#1#2{%
3     \parnotemark{#1}%
4     % Unless this is the first parnote in \PN@text, add a separator first
5     \unless\ifx\PN@text\@empty\g@addto@macro\PN@text{\parnoteintercmd}\fi
6     % Redefine \@currentlabel to the parnote label, so \label works
7     \g@addto@macro\PN@text{%
8 %         \phantomsection%
9         \def\@currentlabel{#1}%
10        \def\cref@currentlabel{%          lwarp
11            [parnotemark][\arabic{parnotemark}][]\theparnotemark%
12        }%
13    }%
14    \g@addto@macro\PN@text{%
15        \LWR@textcurrentfont{%          lwarp
```

```
16              \parnotemark{#1}\nolinebreak\thinspace#2%
17          }%
18      }%
19 }
20
21 \def\PN@parnotes@real{%

22 \ifPN@inparnotes
23 \else

24      \LWR@stoppars%
```

Avoid nested paragraphs:

```
25      \addtocounter{LWR@spandepth}{1}%

26      % We call \par later, so this avoids recursion with \PN@parnotes@auto
27      \PN@inparnotestrue
28 %        \unless\ifvmode\par\fi
29      % Avoid page breaks between a paragraph and its parnotes
30 %        \nopagebreak\addvspace{\parnotevskip}%
31      \begin{BlockClass}(note){footnotes}%    lwarp
32      \leavevmode\LWR@orignewline%
```

Typeset the parnote inside its own group to avoid global changes:

```
33      {%
34          \parnotefmt{\PN@text}%
35      }%
36      \leavevmode\LWR@orignewline%
37      \end{BlockClass}%                              lwarp

38      \leavevmode\LWR@orignewline%
39      \global\def\PN@text{}%
40      %
41      % These can be enabled or disabled by package options
42      %
43      \PN@disable@indent
44      \PN@reset@optional
45      \PN@inparnotesfalse
```

Reenable normal paragraph handling:

```
46      \addtocounter{LWR@spandepth}{-1}%
47 \fi
48 }

49 \newbool{LWR@parnotes@doingauto}
50 \boolfalse{LWR@parnotes@doingauto}

51 \def\PN@parnotes@auto{%
52      \ifbool{LWR@parnotes@doingauto}{
53          \ifx\@currenvir\@PN@autopn
54              \unless\ifPN@inparnotes
55                  \unless\ifx\PN@text\@empty
56                      \expandafter\PN@parnotes@real
```

```
57                \fi
58              \fi
59          \fi
60    }{}%
61 }
```

Replace original logic due to the use of new LaTeXparagraph hook handling:

```
62 \renewenvironment{autopn}%
63    {\booltrue{LWR@parnotes@doingauto}}
64    {\PN@parnotes@auto}%
```

If **cleveref** is in use, name the new notes:

```
65 \AtBeginDocument{
66    \ifdef{\crefname}{
67        \crefname{parnotemark}{paragraph note}{paragraph notes}
68        \Crefname{parnotemark}{Paragraph note}{Paragraph notes}
69    }{}
70 }
```

To nullify the footnotes where necessary:

```
71 \apptocmd{\LWR@nullifyfootnotes}{%
72    \renewcommand{\parnote}[2][]{}%
73    \renewcommand\parnotemark[1]{}%
74 }{}{}
```

For MATHJAX:

```
75 \begin{warpMathJax}
76 \providecommand{\parnotename}{parnote}
77 \appto\LWR@syncnotenumbers{%
78    \addtocounter{parnotemark}{-1}% specific to parnotes
79    \LWR@synconenotenumber{LWRparnote}{\theparnotemark}%
80    \addtocounter{parnotemark}{1}% specific to parnotes
81 }
82 \appto\LWR@syncnotenames{\LWR@synconenotename{LWRparnote}{\parnotename}}
83 \CustomizeMathJax{\def\LWRparnote{1}}
84 \CustomizeMathJax{\newcommand{\parnote}[2][\LWRparnote]{{}^{\mathrm{#1}}}}
85 \CustomizeMathJax{\newcommand{\parnotemark}[1][\LWRparnote]{{}^{\mathrm{#1}}}}
86 \end{warpMathJax}
```

File 359   **lwarp-parskip.sty**

§ 468   Package   **parskip**

parskip (*Pkg*)   parskip is ignored.

for HTML output: Discard all options for lwarp-parskip.

```
1 \LWR@ProvidesPackageDrop{parskip}[2001/04/09]
```

File 360 **`lwarp-pbalance.sty`**

§ 469 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 361 **`lwarp-pbox.sty`**

§ 470 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 362 **`lwarp-pdfcol.sty`**

§ 471 Package **pdfcol**

pdfcol (*Pkg*) pdfcol is ignored.

for HTML output:
```
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 363   **`lwarp-pdfcolfoot.sty`**

§ 472   Package   # pdfcolfoot

pdfcolfoot (*Pkg*)   pdfcolfoot is ignored.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{pdfcolfoot}[2016/05/16]
2
3 \newcommand*{\pdfcolfoot@switch}{}
4
5 \newcommand*{\pdfcolfoot@current}{}
```

File 364   **`lwarp-pdfcolmk.sty`**

§ 473   Package   # pdfcolmk

pdfcolmk (*Pkg*)   pdfcolmk is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{pdfcolmk}[2016/05/16]`

File 365   **`lwarp-pdfcolparallel.sty`**

§ 474   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 366   **`lwarp-pdfcolparcolumns.sty`**

§ 475   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 367   **`lwarp-pdfcomment.sty`**

§ 476   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 368   **lwarp-pdfcrypt.sty**

§ 477   Package   # pdfcrypt

pdfcrypt (*Pkg*)   pdfcrypt is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{pdfcrypt}[2016/05/16]`

`2 \newcommand*{\pdfcryptsetup}[1]{}`

File 369   **lwarp-pdflscape.sty**

§ 478   Package   # pdflscape

pdflscape (*Pkg*)   pdflscape is ignored.

for HTML output: 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 370   **lwarp-pdfmarginpar.sty**

§ 479   Package   # pdfmarginpar

pdfmarginpar (*Pkg*)   pdfmarginpar is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{pdfmarginpar}[2011/08/05]`

```
2 \newcommand{\pdfmarginpar}[2][]{}
3 \newcommand{\pdfmarginparset}[1]{}
```

File 371 **lwarp-pdfpages.sty**

§ 480 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.

for HTML output: 1 \LWR@ProvidesPackagePass{pdfpages}[2022-10-09]

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 \xpatchcmd{\AM@output@i}
49     {\newpage}
50     {\LWR@esopic@newpage}
51     {}
52     {\LWR@patcherror{pdfpages}{AM@output-1}}
53
54 \xpatchcmd{\AM@output@i}
55     {\newpage}
56     {\LWR@esopic@newpage}
57     {}
58     {\LWR@patcherror{pdfpages}{AM@output-2}}
59
60 \xpatchcmd{\AM@output@i}
61     {\newpage}
62     {\LWR@esopic@newpage}
63     {}
64     {\LWR@patcherror{pdfpages}{AM@output-3}}
```

\includepdf   Patched to set the user's paper size.

```
65 \xpretocmd{\includepdf}{%
66     \begingroup%
67     \setlength{\paperwidth}{\LWR@userspaperwidth}%
68     \setlength{\paperheight}{\LWR@userspaperheight}%
69 }{}{}
70
71 \xapptocmd{\includepdf}{%
```

```
72      \endgroup%
73 }{}{}
```

\includepdfmerge   Patched to set the user's paper size.

```
74 \xpretocmd{\includepdfmerge}{%
75      \begingroup%
76      \setlength{\paperwidth}{\LWR@userspaperwidth}%
77      \setlength{\paperheight}{\LWR@userspaperheight}%
78 }{}{}
79
80 \xapptocmd{\includepdfmerge}{%
81      \endgroup%
82 }{}{}
```

\AM@hyper@begin@i   Hyper links are created by \LWR@esopic@newpage, so don't create them here:

```
83 \renewcommand{\AM@hyper@begin@i}{}
```

File 372   **lwarp-pdfprivacy.sty**

§ 481   Package   # pdfprivacy

pdfprivacy (*Pkg*)   pdfprivacy is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{pdfprivacy}[2017/12/03]

File 373   **lwarp-pdfrender.sty**

§ 482   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 374   **lwarp-pdfsync.sty**

§ 483   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 375   `lwarp-pdftricks.sty`

§ 484   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 376   `lwarp-pdfx.sty`

§ 485   Package   **pdfx**

pdfx (*Pkg*)   pdfx is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{pdfx}[2017/05/18]

File 377   `lwarp-perpage.sty`

§ 486   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 378   **`lwarp-pfnote.sty`**

§ 487   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 379   **`lwarp-phfqit.sty`**

§ 488   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 380   **`lwarp-physics.sty`**

§ 489   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 381   **`lwarp-physunits.sty`**

§ 490   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 382   **`lwarp-picinpar.sty`**

§ 491   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 383   **lwarp-pifont.sty**

§ 492   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 384   **lwarp-pinlabel.sty**

§ 493   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 385   **`lwarp-placeins.sty`**

§ 494   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 386   **`lwarp-plarydshln.sty`**

§ 495   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 387   **`lwarp-plext.sty`**

§ 496   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 388   `lwarp-plextarydshln.sty`

§ 497   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 389   `lwarp-plextcolortbl.sty`

§ 498   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 390   `lwarp-plimsoll.sty`

§ 499   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 391   `lwarp-prelim2e.sty`

§ 500   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 392   `lwarp-prettyref.sty`

§ 501   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 393   **`lwarp-preview.sty`**

§ 502   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 394   **`lwarp-psfrag.sty`**

§ 503   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 395   **`lwarp-psfragx.sty`**

§ 504   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 \def\pfx@includegraphicx#1#2{%
3     \begin{lateximage}[-psfragx-~\PackageDiagramAltText]%
4     \mbox{\pfx@overpix{#1}{#2}\endpfx@overpix}%
5     \end{lateximage}%
6 }
7
8 \def\@@@overpix[#1]<#2>[#3]#4{%
9     \begin{lateximage}[-psfragx-~\PackageDiagramAltText]%
10    \pfx@overpix{#1,ovpfgd={#2},ovpbgd={#3}}{#4}%
11 }
12
13 \def\endoverpix{%
14    \endpfx@overpix%
15    \end{lateximage}%
16 }
```

File 396   **`lwarp-pst-eps.sty`**

§ 505   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 397   **`lwarp-pstool.sty`**

§ 506   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`**

.

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 398   **`lwarp-pstricks.sty`**

§ 507   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.

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 399   **`lwarp-pxatbegshi.sty`**

§ 508   Package   # pxatbegshi

pxatbegshi (*Pkg*)   pxatbegshi is ignored.

1 `\LWR@ProvidesPackageDrop{pxatbegshi}[2017/11/04]`

2 `\LWR@origRequirePackage{lwarp-atbegshi}`

File 400   `lwarp-pxeveryshi.sty`

§ 509   Package   **pxeveryshi**

pxeveryshi (*Pkg*)   pxeveryshi is ignored.

for HTML output: 1 `\LWR@ProvidesPackageDrop{pxeveryshi}[2012/05/19]`

2 `\LWR@origRequirePackage{lwarp-everyshi}`

File 401   `lwarp-pxfonts.sty`

§ 510   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 402   `lwarp-pxftnright.sty`

§ 511   Package   **pxftnright**

pxftnright (*Pkg*)   pxftnright is ignored.

for HTML output: 1 `\LWR@ProvidesPackageDrop{pxftnright}[2017/02/28]`

2 `\LWR@origRequirePackage{lwarp-ftnright}`

File 403   `lwarp-pxjahyper.sty`

§ 512   Package   **pxjahyper**

pxjahyper (*Pkg*)   pxjahyper is ignored.

for HTML output: 1 `\LWR@ProvidesPackageDrop{pxjahyper}[2018/07/15]`

File 404   **`lwarp-quotchap.sty`**

§ 513   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 405 **`lwarp-quoting.sty`**

§ 514   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 \xpatchcmd{\quoting}{\quo@begintext}
3    {\begin{LWR@blocktextcurrentfont}\quo@begintext}
4    {}
5    {\LWR@patcherror{quoting}{quoting}}
6
7 \xpatchcmd{\endquoting}{\quo@endtext}
8    {\quo@endtext\end{LWR@blocktextcurrentfont}\LWR@stoppars}
9    {}
10   {\LWR@patcherror{quoting}{endquoting}}
```

File 406   **`lwarp-ragged2e.sty`**

§ 515   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 407   **lwarp-realscripts.sty**

§ 516   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 408 **`lwarp-refcheck.sty`**

§ 517   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 409   **`lwarp-register.sty`**

§ 518   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]`

```
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 410 **`lwarp-relsize.sty`**

§ 519 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 411   **`lwarp-repeatindex.sty`**

§ 520   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 412   **`lwarp-repltext.sty`**

§ 521   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 413   **`lwarp-resizegather.sty`**

§ 522   Package   # resizegather

resizegather (*Pkg*)   resizegather is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{resizegather}[2016/05/16]

```
2 \newcommand*{\resizegathersetup}[1]{}
```

File 414 **`lwarp-returntogrid.sty`**

§ 523   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 415   **`lwarp-rlepsf.sty`**

§ 524   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 416   **`lwarp-rmathbr.sty`**

§ 525   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 417 **`lwarp-rmpage.sty`**

§ 526　Package **rmpage**

rmpage (*Pkg*)　rmpage is ignored.

for HTML output:　1 `\LWR@ProvidesPackageDrop{rmpage}[1997/09/29]`

File 418 **`lwarp-romanbar.sty`**

§ 527　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 419 **`lwarp-romanbarpagenumber.sty`**

§ 528　Package **romanbarpagenumber**

romanbarpagenumber (*Pkg*)　romanbarpagenumber is ignored.

for HTML output:　1 `\LWR@ProvidesPackageDrop{romanbarpagenumber}[2015/02/06]`

File 420 **`lwarp-rotating.sty`**

§ 529　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 421   **`lwarp-rotfloat.sty`**

§ 530   Package   **rotfloat**

(*Emulates or patches code by* AXEL SOMMERFELDT.)

rotfloat (*Pkg*)   rotfloat is emulated.

for HTML output:
```
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 422   **lwarp-rviewport.sty**

§ 531   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 423   **lwarp-savetrees.sty**

§ 532   Package   # savetrees

savetrees (*Pkg*)   savetrees is ignored.

for HTML output: Discard all options for lwarp-savetrees:

`1 \LWR@ProvidesPackageDrop{savetrees}[2016/04/13]`

File 424   **lwarp-scalefnt.sty**

§ 533   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 425   **`lwarp-scalerel.sty`**

§ 534   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 426   **`lwarp-schemata.sty`**

§ 535   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 427   **`lwarp-scrextend.sty`**

§ 536   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 428   **lwarp-scrhack.sty**

§ 537   Package   # scrhack

scrhack (*Pkg*)   scrhack is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{scrhack}[2018/03/30]

File 429   **`lwarp-scrlayer.sty`**

§ 538   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 430   **`lwarp-scrlayer-notecolumn.sty`**

§ 539   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 431   **`lwarp-scrlayer-scrpage.sty`**

§ 540   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 432 **lwarp-scrpage2.sty**

§ 541 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 433 **`lwarp-section.sty`**

§ 542 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 434 **`lwarp-sectionbreak.sty`**

§ 543 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 435   **`lwarp-sectsty.sty`**

§ 544   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 436   **`lwarp-selectp.sty`**

§ 545   Package   # selectp

selectp (*Pkg*)   selectp is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{selectp}% no date given

```
2 \newcommand*{\outputonly}[1]{}
```

File 437   **`lwarp-semantic-markup.sty`**

§ 546   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 438 **`lwarp-seqsplit.sty`**

§ 547 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 439   **`lwarp-setspace.sty`**

§ 548   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 440   `lwarp-shadethm.sty`

§ 549   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 441   `lwarp-shadow.sty`

§ 550   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 442   `lwarp-shapepar.sty`

§ 551   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 443   **`lwarp-showidx.sty`**

§ 552   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 444   **`lwarp-showkeys.sty`**

§ 553   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 445   **`lwarp-showlabels.sty`**

§ 554   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 446   `lwarp-showtags.sty`

§ 555   Package   # showtags

showtags (*Pkg*)   showtags is ignored.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{showtags}% no version is given
```

```
2 \newcommand{\thecitetag}[1]{}
```

File 447   `lwarp-shuffle.sty`

§ 556   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.

for HTML output:
```
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 448 **`lwarp-sidecap.sty`**

§ 557   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 449   **`lwarp-sidenotes.sty`**

§ 558   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 \RenewDocumentCommand \sidecaption {s o o m}
3 {
4     \LWR@stoppars
5     \begingroup
6   \captionsetup{style=sidecaption}%
7   \IfBooleanTF{#1}
8   { % starred
9     \begin{BlockClass}[border:none ; box-shadow:none]{marginblock}%
10    \caption*{#4}%
11    \end{BlockClass}
12  }
13  { % unstarred
14  \IfNoValueOrEmptyTF{#2}
15    {\def\@sidenotes@sidecaption@tof{#4}}
16    {\def\@sidenotes@sidecaption@tof{#2}}
17    \begin{BlockClass}[border:none ; box-shadow:none]{marginblock}%
18    \caption[\@sidenotes@sidecaption@tof]{#4}
19    \end{BlockClass}
20  }
21    \endgroup
22    \LWR@startpars
23 }
```

Borrowed from the lwarp version of keyfloat:

```
24 \NewDocumentEnvironment{KFLTsidenotes@marginfloat}{O{-1.2ex} m}
25 {% start
26     \LWR@BlockClassWP{float:right; width:2in; margin:10pt}{}{marginblock}%
27     \renewcommand*{\@captype}{#2}%
28 }
29 {%
30     \endLWR@BlockClassWP%
31 }
32
33 \RenewDocumentEnvironment{marginfigure}{o}
34   {\begin{KFLTsidenotes@marginfloat}{figure}}
35   {\end{KFLTsidenotes@marginfloat}}
36
37 \RenewDocumentEnvironment{margintable}{o}
38   {\begin{KFLTsidenotes@marginfloat}{table}}
39   {\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:

```
40 \renewenvironment{figure*}
```

```
41                    {\@dblfloat{figure}}
42                    {\end@dblfloat}
43
44 \renewenvironment{table*}
45                    {\@dblfloat{table}}
46                    {\end@dblfloat}
```

For MATHJAX:

⚠ Note that sidenotes does not support \sidenote inside math in print mode. Use
   \sidenotemark and \sidenotetext instead.

```
47 \begin{warpMathJax}
48 \providecommand{\sidenotename}{sidenote}
49 \appto\LWR@syncnotenumbers{\LWR@synconenotenumber{LWRsidenote}{\thesidenote}}
50 \appto\LWR@syncnotenames{\LWR@synconenotename{LWRsidenote}{\sidenotename}}
51 \CustomizeMathJax{\def\LWRsidenote{1}}
52 \CustomizeMathJax{\newcommand{\sidenotemark}[1][\LWRsidenote]{{}^{\mathrm{#1}}}}
53 \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:

   \CustomizeMathJax{\newcommand{\sidenote}[2][\LWRsidenote]{{}^{\mathrm{#1}}}}

File 450   **lwarp-simplebnf.sty**

§ 559   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}[2022/05/08]

The entire object is placed inside a lateximage whose alt text is the LATEX source
BNF expression.

```
2 \ExplSyntaxOn
3
4 \RenewDocumentEnvironment { bnfgrammar } { O{llcll} O{[^\|]\|[^\|]} O{\|\|} +b }
5 {
6     \regex_gset:Nn \g_simplebnf_rhs_newline_r { #2 }
7     \regex_gset:Nn \g_simplebnf_rhs_nb_r { #3 }
8
9     %% \l__input_seq is a list of term definitions.
10    \regex_split:nnN { ;; } { #4 } \l__input_seq
11    \begin{center}
12    \begin{lateximage}[#4]%       lwarp
13      \tl_set:Nn \l__table_tl
14        {
15           \begin{tabular}{#1}
16        }
17
18    \bool_set_true:N \l_tmp_first_term % Is this the first term in this grammar?
19    \seq_map_inline:Nn \l__input_seq
```

```
20        {
21          %% If not-first, add newline
22          \bool_if:NTF \l_tmp_first_term
23            {
24               \bool_set_false:N \l_tmp_first_term
25            }
26            {
27               \tl_put_right:Nn \l__table_tl { \\ }
28            }
29
30          \regex_split:nnNTF { ::= } { ##1 } \l__term_seq
31            % Parse a ::= definition
32            {
33               %% \l__term_seq   - (lhs, rhses)...
34               %% \l__term_tl    - lhs
35               %% \l__keypairs_tl - rhses
36               \seq_pop_left:NN \l__term_seq \l__term_tl
37               \seq_pop_left:NN \l__term_seq \l__keypairs_tl
38
39               \simplebnf_typeset_lhs:n{\l__term_tl}
40               \tl_put_right:Nn \l__table_tl
41                 {
42                    & \SimpleBNFDefEq &
43                 }
44               %% \l__keypairs_seq - (rhs:annot | rhs)...
45          \regex_split:NVN \g_simplebnf_rhs_newline_r \l__keypairs_tl \l__keypairs_seq
46
47               \bool_set_true:N \l__first_rhs
48               \seq_map_function:NN \l__keypairs_seq \simplebnf_typeset_rhs:n
49            }
50            {
51               % Else, parse a \in declaration
52               \regex_split:nnNTF { \c{in} } { ##1 } \l__term_seq
53                 {
54                    %% \l__term_seq - (lhs, rhs)
55                    \seq_pop_left:NN \l__term_seq \l_tmpa_tl
56
57                    \simplebnf_typeset_lhs:n{\l_tmpa_tl}
58                    \tl_put_right:Nn \l__table_tl
59                      {
60                         & $\in$ & $
61                      }
62                    \seq_pop_left:NN \l__term_seq \l_tmpa_tl
63                    \tl_put_right:NV \l__table_tl \l_tmpa_tl
64                    \tl_put_right:Nn \l__table_tl
65                      {
66                         $ &
67                      }
68                 }
69                 { \msg_error:nn {simplebnf} { Could not parser ##1 } }
70            }
71       }
72
73    \tl_put_right:Nn \l__table_tl { \end{tabular} }
74    \tl_use:N \l__table_tl
75    \end{lateximage}%        lwarp
76    \end{center}
77 }
78 { }
79
```

80 \ExplSyntaxOff

File 451 **`lwarp-SIunits.sty`**

§ 560   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 MATHJAXinstead 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         \LWR@subsingledollar*%  lwarp
5         {% alt tag
6             \textbackslash{}unit%
7             \{\LWR@HTMLsanitizedetokenized{\detokenize{#1}}\}%
8              \{ \LWR@HTMLsanitizedetokenized{\detokenize{#2}}\}% extra space
9         }%
10        {SIunits}% add'l hashing
11        {%
12            #1\,{#2}%
13        }% contents
14    }
15 }{% not MathJax
16     \DeclareRobustCommand{\LWR@HTML@unit}[2]{%
17         \@inunitcommandtrue%    original
18         \LWR@subsingledollar*%  lwarp
19         {% alt tag
20         \textbackslash{}unit\{\LWR@HTMLsanitizedetokenized{\detokenize{#1}}\}%
21             \{ \LWR@HTMLsanitizedetokenized{\detokenize{#2}}\}% extra space
22         }%
23         {SIunits}% add'l hashing
24         {%
25             \LWR@origensuredmath{%  lwarp modification
26                 \SI@fstyle{%
27                     {#1}\@qsk\period@active{#2}%
28                 }%  original
29             }%
30         }% contents
31         \@inunitcommandfalse%   original
32     }
```

```
33 }% not MathJax
34 \LWR@formatted{unit}
```

For MATHJAX:

```
35 \begin{warpMathJax}
36 \LWR@infoprocessingmathjax{SIunits}
37
38 \CustomizeMathJax{\newcommand{\one}{}}
39 \CustomizeMathJax{\newcommand{\meter}{\metre}}
40 \CustomizeMathJax{\newcommand{\deka}{\deca}}
41 \CustomizeMathJax{\newcommand{\dekad}{\decad}}
42 \CustomizeMathJax{\newcommand{\per}{/}}
43 \CustomizeMathJax{\newcommand{\usk}{\;}}
44 \CustomizeMathJax{\newcommand{\unit}[2]{#1\,{#2}}}
45 \CustomizeMathJax{\newcommand{\power}[2]{#1^{#2}}}
46
47 \AtBeginDocument{%
48 \if@redefsquare
49  \CustomizeMathJax{\renewcommand{\square}[1]{\power{#1}{2}}}
50 \else
51   \if@defsquaren
52    \CustomizeMathJax{\newcommand{\squaren}[1]{\power{#1}{2}}}
53   \else
54    \CustomizeMathJax{\renewcommand{\square}[1]{\power{#1}{2}}}
55   \fi %\if@defsquaren
56 \fi   %\if@redefsquare
57 }     %\AtBeginDocument
58
59 \CustomizeMathJax{\newcommand{\squared}{^{2}}}
60 \CustomizeMathJax{\newcommand{\cubic}[1]{\power{#1}{3}}}
61 \CustomizeMathJax{\newcommand{\cubed}{^{3}}}
62 \CustomizeMathJax{\newcommand{\fourth}[1]{\power{#1}{4}}}
63 \CustomizeMathJax{\newcommand{\reciprocal}[1]{\power{#1}{-1}}}
64 \CustomizeMathJax{\newcommand{\rp}{\reciprocal}}
65 \CustomizeMathJax{\newcommand{\rpsquare}[1]{\power{#1}{-2}}}
66 \CustomizeMathJax{\newcommand{\rpsquared}{^{-2}}}
67 \CustomizeMathJax{\newcommand{\rpcubic}[1]{\power{#1}{-3}}}
68 \CustomizeMathJax{\newcommand{\rpcubed}{^{-3}}}
69 \CustomizeMathJax{\newcommand{\rpfourth}[1]{\power{#1}{-4}}}
70 \CustomizeMathJax{\newcommand{\yocto}{\mathrm{y}}}
71 \CustomizeMathJax{\newcommand{\zepto}{\mathrm{z}}}
72 \CustomizeMathJax{\newcommand{\atto}{\mathrm{a}}}
73 \CustomizeMathJax{\newcommand{\femto}{\mathrm{f}}}
74 \CustomizeMathJax{\newcommand{\pico}{\mathrm{p}}}
75 \CustomizeMathJax{\newcommand{\nano}{\mathrm{n}}}
76 \CustomizeMathJax{\newcommand{\micro}{\mathrm{\unicode{x00B5}}}}
77 \CustomizeMathJax{\newcommand{\milli}{\mathrm{m}}}
78 \CustomizeMathJax{\newcommand{\centi}{\mathrm{c}}}
79 \CustomizeMathJax{\newcommand{\deci}{\mathrm{d}}}
80 \CustomizeMathJax{\newcommand{\deca}{\mathrm{da}}}
81 \CustomizeMathJax{\newcommand{\hecto}{\mathrm{h}}}
82 \CustomizeMathJax{\newcommand{\kilo}{\mathrm{k}}}
83 \CustomizeMathJax{\newcommand{\mega}{\mathrm{M}}}
84 \CustomizeMathJax{\newcommand{\giga}{\mathrm{G}}}
85 \CustomizeMathJax{\newcommand{\tera}{\mathrm{T}}}
86 \CustomizeMathJax{\newcommand{\peta}{\mathrm{P}}}
87 \CustomizeMathJax{\newcommand{\exa}{\mathrm{E}}}
88 \CustomizeMathJax{\newcommand{\zetta}{\mathrm{Z}}}
89 \CustomizeMathJax{\newcommand{\yotta}{\mathrm{Y}}}
```

```
 90 \CustomizeMathJax{\newcommand{\yoctod}{\power{10}{-24}}}
 91 \CustomizeMathJax{\newcommand{\zeptod}{\power{10}{-21}}}
 92 \CustomizeMathJax{\newcommand{\attod}{\power{10}{-18}}}
 93 \CustomizeMathJax{\newcommand{\femtod}{\power{10}{-15}}}
 94 \CustomizeMathJax{\newcommand{\picod}{\power{10}{-12}}}
 95 \CustomizeMathJax{\newcommand{\nanod}{\power{10}{-9}}}
 96 \CustomizeMathJax{\newcommand{\microd}{\power{10}{-6}}}
 97 \CustomizeMathJax{\newcommand{\millid}{\power{10}{-3}}}
 98 \CustomizeMathJax{\newcommand{\centid}{\power{10}{-2}}}
 99 \CustomizeMathJax{\newcommand{\decid}{\power{10}{-1}}}
100 \CustomizeMathJax{\newcommand{\decad}{\power{10}{1}}}
101 \CustomizeMathJax{\newcommand{\hectod}{\power{10}{2}}}
102 \CustomizeMathJax{\newcommand{\kilod}{\power{10}{3}}}
103 \CustomizeMathJax{\newcommand{\megad}{\power{10}{6}}}
104 \CustomizeMathJax{\newcommand{\gigad}{\power{10}{9}}}
105 \CustomizeMathJax{\newcommand{\terad}{\power{10}{12}}}
106 \CustomizeMathJax{\newcommand{\petad}{\power{10}{15}}}
107 \CustomizeMathJax{\newcommand{\exad}{\power{10}{18}}}
108 \CustomizeMathJax{\newcommand{\zettad}{\power{10}{21}}}
109 \CustomizeMathJax{\newcommand{\yottad}{\power{10}{24}}}
110 \CustomizeMathJax{\newcommand{\gram}{\mathrm{g}}}
111 \CustomizeMathJax{\newcommand{\metre}{\mathrm{m}}}
112 \CustomizeMathJax{\newcommand{\kilogram}{\kilo\gram}}
113 \CustomizeMathJax{\newcommand{\second}{\mathrm{s}}}
114 \CustomizeMathJax{\newcommand{\ampere}{\mathrm{A}}}
115 \CustomizeMathJax{\newcommand{\kelvin}{\mathrm{K}}}
116 \CustomizeMathJax{\newcommand{\mole}{\mathrm{mol}}}
117 \CustomizeMathJax{\newcommand{\candela}{\mathrm{cd}}}
118 \CustomizeMathJax{\newcommand{\radian}{\mathrm{rad}}}
119 \CustomizeMathJax{\newcommand{\steradian}{\mathrm{sr}}}
120 \CustomizeMathJax{\newcommand{\hertz}{\mathrm{Hz}}}
121 \CustomizeMathJax{\newcommand{\newton}{\mathrm{N}}}
122 \CustomizeMathJax{\newcommand{\pascal}{\mathrm{Pa}}}
123 \CustomizeMathJax{\newcommand{\joule}{\mathrm{J}}}
124 \CustomizeMathJax{\newcommand{\watt}{\mathrm{W}}}
125 \CustomizeMathJax{\newcommand{\coulomb}{\mathrm{C}}}
126 \CustomizeMathJax{\newcommand{\volt}{\mathrm{V}}}
127 \CustomizeMathJax{\newcommand{\farad}{\mathrm{F}}}
128 \CustomizeMathJax{\newcommand{\ohm}{\mathrm{\Omega}}}
129 \CustomizeMathJax{\newcommand{\siemens}{\mathrm{S}}}
130 \CustomizeMathJax{\newcommand{\weber}{\mathrm{Wb}}}
131 \CustomizeMathJax{\newcommand{\tesla}{\mathrm{T}}}
132 \CustomizeMathJax{\newcommand{\henry}{\mathrm{H}}}
133 \CustomizeMathJax{\newcommand{\degreecelsius}{\mathrm{\unicode{x2103}}}}
134 \CustomizeMathJax{\newcommand{\celsius}{\degreecelsius}}
135 \CustomizeMathJax{\newcommand{\lumen}{\mathrm{lm}}}
136 \CustomizeMathJax{\newcommand{\lux}{\mathrm{lx}}}
137 \CustomizeMathJax{\newcommand{\becquerel}{\mathrm{Bq}}}
138 \CustomizeMathJax{\newcommand{\sievert}{\mathrm{Sv}}}
139 \CustomizeMathJax{\newcommand{\katal}{\mathrm{kat}}}
140
141 \ifdef{\radianbase}{
142 \CustomizeMathJax{\newcommand{\radianbase}%
143         {\metre\usk\reciprocal\metre}}
144 \CustomizeMathJax{\newcommand{\steradianbase}%
145         {\squaremetre\usk\rpsquare\metre}}
146 \CustomizeMathJax{\newcommand{\hertzbase}%
147         {\reciprocal\second}}
148 \CustomizeMathJax{\newcommand{\newtonbase}%
149         {\metre\usk\kilogram\usk\second\rpsquared}}
```

```
150 \CustomizeMathJax{\newcommand{\pascalbase}%
151        {\reciprocal\metre\usk\kilogram\usk\second\rpsquared}}
152 \CustomizeMathJax{\newcommand{\joulebase}%
153        {\squaremetre\usk\kilogram\usk\second\rpsquared}}
154 \CustomizeMathJax{\newcommand{\wattbase}%
155        {\squaremetre\usk\kilogram\usk\rpcubic\second}}
156 \CustomizeMathJax{\newcommand{\coulombbase}%
157        {\ampere\usk\second}}
158 \CustomizeMathJax{\newcommand{\voltbase}%
159        {\squaremetre\usk\kilogram\usk\rpcubic\second\usk\reciprocal\ampere}}
160 \CustomizeMathJax{\newcommand{\faradbase}%
161       {\rpsquare\metre\usk\reciprocal\kilogram\usk\fourth\second\usk\ampere\squared}}
162 \CustomizeMathJax{\newcommand{\ohmbase}%
163        {\squaremetre\usk\kilogram\usk\rpcubic\second\usk\rpsquare\ampere}}
164 \CustomizeMathJax{\newcommand{\siemensbase}%
165       {\rpsquare\metre\usk\reciprocal\kilogram\usk\cubic\second\usk\ampere\squared}}
166 \CustomizeMathJax{\newcommand{\weberbase}%
167       {\squaremetre\usk\kilogram\usk\second\rpsquared\usk\reciprocal\ampere}}
168 \CustomizeMathJax{\newcommand{\teslabase}%
169        {\kilogram\usk\second\rpsquared\usk\reciprocal\ampere}}
170 \CustomizeMathJax{\newcommand{\henrybase}%
171       {\squaremetre\usk\kilogram\usk\second\rpsquared\usk\rpsquare\ampere}}
172 \CustomizeMathJax{\newcommand{\celsiusbase}%
173        {\kelvin}}
174 \CustomizeMathJax{\newcommand{\lumenbase}%
175        {\candela\usk\squaremetre\usk\rpsquare\metre}}
176 \CustomizeMathJax{\newcommand{\luxbase}%
177        {\candela\usk\squaremetre\usk\rpfourth\metre}}
178 \CustomizeMathJax{\newcommand{\becquerelbase}%
179        {\hertzbase}}
180 \CustomizeMathJax{\newcommand{\graybase}%
181        {\squaremetre\usk\second\rpsquared}}
182 \CustomizeMathJax{\newcommand{\sievertbase}%
183        {\graybase}}
184 \CustomizeMathJax{\newcommand{\katalbase}%
185        {\rp\second\usk\mole }}
186 }{}
187
188 \ifdef{\derradian}{
189 \CustomizeMathJax{\newcommand{\derradian}%
190        {\metre\usk\reciprocal\metre}}
191 \CustomizeMathJax{\newcommand{\dersteradian}%
192        {\squaremetre\usk\rpsquare\metre}}
193 \CustomizeMathJax{\newcommand{\derhertz}%
194        {\reciprocal\second}}
195 \CustomizeMathJax{\newcommand{\dernewton}%
196        {\metre\usk\kilogram\usk\second\rpsquared}}
197 \CustomizeMathJax{\newcommand{\derpascal}%
198        {\newton\usk\rpsquare\metre}}
199 \CustomizeMathJax{\newcommand{\derjoule}%
200        {\newton\usk\metre}}
201 \CustomizeMathJax{\newcommand{\derwatt}%
202        {\joule\usk\reciprocal\second}}
203 \CustomizeMathJax{\newcommand{\dercoulomb}%
204        {\ampere\usk\second}}
205 \CustomizeMathJax{\newcommand{\dervolt}%
206        {\watt\usk\reciprocal\ampere}}
207 \CustomizeMathJax{\newcommand{\derfarad}%
208        {\coulomb\usk\reciprocal\volt}}
209 \CustomizeMathJax{\newcommand{\derohm}%
```

```
210          {\volt\usk\reciprocal\ampere}}
211 \CustomizeMathJax{\newcommand{\dersiemens}%
212          {\ampere\usk\reciprocal\volt}}
213 \CustomizeMathJax{\newcommand{\derweber}%
214      {\squaremetre\usk\kilogram\usk\second\rpsquared\usk\reciprocal\ampere}}
215 \CustomizeMathJax{\newcommand{\dertesla}%
216          {\weber\usk\rpsquare\metre}}
217 \CustomizeMathJax{\newcommand{\derhenry}%
218          {\weber\usk\reciprocal\ampere}}
219 \CustomizeMathJax{\newcommand{\dercelsius}%
220          {\kelvin}}
221 \CustomizeMathJax{\newcommand{\derlumen}%
222          {\candela\usk\steradian}}
223 \CustomizeMathJax{\newcommand{\derlux}%
224          {\lumen\usk\rpsquare\metre}}
225 \CustomizeMathJax{\newcommand{\derbecquerel}%
226          {\derhertz}}
227 \CustomizeMathJax{\newcommand{\dergray}%
228          {\joule\usk\reciprocal\kilogram}}
229 \CustomizeMathJax{\newcommand{\dersievert}%
230          {\dergray}}
231 \CustomizeMathJax{\newcommand{\derkatal}%
232          {\katalbase}}
233 }{}
234
235 \CustomizeMathJax{\newcommand{\minute}{\mathrm{min}}}
236 \CustomizeMathJax{\newcommand{\hour}{\mathrm{h}}}
237 \CustomizeMathJax{\newcommand{\dday}{\mathrm{d}}}
238 \CustomizeMathJax{\newcommand{\degree}{\mathrm{^\circ}}}
239 \CustomizeMathJax{\newcommand{\paminute}{^\prime}}
240 \CustomizeMathJax{\newcommand{\arcminute}{^\prime}}
241 \CustomizeMathJax{\newcommand{\pasecond}{^{\prime\prime}}}
242 \CustomizeMathJax{\newcommand{\arcsecond}{^{\prime\prime}}}
243 \CustomizeMathJax{\newcommand{\ton}{\mathrm{t}}}
244 \CustomizeMathJax{\newcommand{\tonne}{\mathrm{t}}}
245 \CustomizeMathJax{\newcommand{\liter}{\mathrm{L}}}
246 \CustomizeMathJax{\newcommand{\litre}{\mathrm{l}}}
247 \CustomizeMathJax{\newcommand{\neper}{\mathrm{Np}}}
248 \CustomizeMathJax{\newcommand{\bel}{\mathrm{B}}}
249 \CustomizeMathJax{\newcommand{\curie}{\mathrm{Ci}}}
250 \CustomizeMathJax{\newcommand{\rad}{\mathrm{rad}}}
251 \CustomizeMathJax{\newcommand{\arad}{\mathrm{rd}}}
252 \CustomizeMathJax{\newcommand{\rem}{\mathrm{rem}}}
253 \CustomizeMathJax{\newcommand{\roentgen}{\mathrm{R}}}
254 \CustomizeMathJax{\newcommand{\electronvolt}{\mathrm{\mathrm{eV}}}}
255 \CustomizeMathJax{\newcommand{\atomicmass}{\mathrm{u}}}
256 \CustomizeMathJax{\newcommand{\atomicmassunit}{\mathrm{u}}}
257 \CustomizeMathJax{\newcommand{\dalton}{\mathrm{Da}}}
258 \CustomizeMathJax{\newcommand{\are}{\mathrm{a}}}
259 \CustomizeMathJax{\newcommand{\hectare}{\mathrm{\hecto\are}}}
260 \CustomizeMathJax{\newcommand{\barn}{\mathrm{b}}}
261 \CustomizeMathJax{\newcommand{\bbar}{\mathrm{bar}}}
262 \CustomizeMathJax{\newcommand{\gal}{\mathrm{Gal}}}
263 \CustomizeMathJax{\newcommand{\angstrom}{\mathrm{\unicode{x212B}}}}
264 \CustomizeMathJax{\newcommand{\rperminute}{\mathrm{r}\per\minute}}
265 \CustomizeMathJax{\newcommand{\rpersecond}{\mathrm{r}\per\second}}
266 \CustomizeMathJax{\newcommand{\squaremetre}{\power{\metre}{2}}}
267 \CustomizeMathJax{\newcommand{\cubicmetre}{\cubic\metre}}
268 \CustomizeMathJax{\newcommand{\graypersecond}{\gray\per\second}}
269 \CustomizeMathJax{\newcommand{\graypersecondnp}{\gray\usk\reciprocal\second}}
```

```
270 \CustomizeMathJax{\newcommand{\metrepersquaresecond}{\metre\per\second\squared}}
271 \CustomizeMathJax{\newcommand{\metrepersquaresecondnp}{\metre\usk\second\rpsquared}}
272 \CustomizeMathJax{\newcommand{\joulepermole}{\joule\per\mole}}
273 \CustomizeMathJax{\newcommand{\joulepermolenp}{\joule\usk\reciprocal\mole}}
274 \CustomizeMathJax{\newcommand{\molepercubicmetre}{\mole\per\cubic\metre}}
275 \CustomizeMathJax{\newcommand{\molepercubicmetrenp}{\mole\usk\rpcubic\metre}}
276 \CustomizeMathJax{\newcommand{\radianpersquaresecond}{\radian\per\second\squared}}
277 \CustomizeMathJax{\newcommand{\radianpersquaresecondnp}{\radian\usk\second\rpsquared}}
278 \CustomizeMathJax{\newcommand{\kilogramsquaremetrepersecond}{%
279     \kilogram\usk\squaremetre\per\second%
280 }}
281 \CustomizeMathJax{\newcommand{\kilogramsquaremetrepersecondnp}{%
282     \kilogram\usk\squaremetre\usk\reciprocal\second%
283 }}
284 \CustomizeMathJax{\newcommand{\radianpersecond}{\radian\per\second}}
285 \CustomizeMathJax{\newcommand{\radianpersecondnp}{\radian\usk\reciprocal\second}}
286 \CustomizeMathJax{\newcommand{\squaremetrepercubicmetre}{\squaremetre\per\cubic\metre}}
287 \CustomizeMathJax{\newcommand{\squaremetrepercubicmetrenp}{%
288     \squaremetre\usk\rpcubic\metre%
289 }}
290 \CustomizeMathJax{\newcommand{\katalpercubicmetre}{\katal\per\cubic\metre}}
291 \CustomizeMathJax{\newcommand{\katalpercubicmetrenp}{\katal\usk\rpcubic\metre}}
292 \CustomizeMathJax{\newcommand{\coulombpermol}{\coulomb\per\mole}}
293 \CustomizeMathJax{\newcommand{\coulombpermolnp}{\coulomb\usk\reciprocal\mole}}
294 \CustomizeMathJax{\newcommand{\amperepersquaremetre}{\ampere\per\squaremetre}}
295 \CustomizeMathJax{\newcommand{\amperepersquaremetrenp}{\ampere\usk\rpsquare\metre}}
296 \CustomizeMathJax{\newcommand{\kilogrampercubicmetre}{\kilogram\per\cubic\metre}}
297 \CustomizeMathJax{\newcommand{\kilogrampercubicmetrenp}{\kilogram\usk\rpcubic\metre}}
298 \CustomizeMathJax{\newcommand{\squaremetrepernewtonsecond}{%
299     \squaremetre\per\newton\usk\second%
300 }}
301 \CustomizeMathJax{\newcommand{\squaremetrepernewtonsecondnp}{%
302     \squaremetre\usk\reciprocal\newton\usk\reciprocal\second%
303 }}
304 \CustomizeMathJax{\newcommand{\pascalsecond}{\pascal\usk\second}}
305 \CustomizeMathJax{\newcommand{\coulombpercubicmetre}{\coulomb\per\cubic\metre}}
306 \CustomizeMathJax{\newcommand{\coulombpercubicmetrenp}{\coulomb\usk\rpcubic\metre}}
307 \CustomizeMathJax{\newcommand{\amperemetresecond}{\ampere\usk\metre\usk\second}}
308 \CustomizeMathJax{\newcommand{\voltpermetre}{\volt\per\metre}}
309 \CustomizeMathJax{\newcommand{\voltpermetrenp}{\volt\usk\reciprocal\metre}}
310 \CustomizeMathJax{\newcommand{\coulombpersquaremetre}{\coulomb\per\squaremetre}}
311 \CustomizeMathJax{\newcommand{\coulombpersquaremetrenp}{\coulomb\usk\rpsquare\metre}}
312 \CustomizeMathJax{\newcommand{\faradpermetre}{\farad\per\metre}}
313 \CustomizeMathJax{\newcommand{\faradpermetrenp}{\farad\usk\reciprocal\metre}}
314 \CustomizeMathJax{\newcommand{\ohmmetre}{\ohm\usk\metre}}
315 \CustomizeMathJax{\newcommand{\kilowatthour}{\kilo\watt\hour}}
316 \CustomizeMathJax{\newcommand{\wattpersquaremetre}{\watt\per\squaremetre}}
317 \CustomizeMathJax{\newcommand{\wattpersquaremetrenp}{\watt\usk\rpsquare\metre}}
318 \CustomizeMathJax{\newcommand{\joulepersquaremetre}{\joule\per\squaremetre}}
319 \CustomizeMathJax{\newcommand{\joulepersquaremetrenp}{\joule\usk\rpsquare\metre}}
320 \CustomizeMathJax{\newcommand{\newtonpercubicmetre}{\newton\per\cubic\metre}}
321 \CustomizeMathJax{\newcommand{\newtonpercubicmetrenp}{\newton\usk\rpcubic\metre}}
322 \CustomizeMathJax{\newcommand{\newtonperkilogram}{\newton\per\kilogram}}
323 \CustomizeMathJax{\newcommand{\newtonperkilogramnp}{\newton\usk\reciprocal\kilogram}}
324 \CustomizeMathJax{\newcommand{\jouleperkelvin}{\joule\per\kelvin}}
325 \CustomizeMathJax{\newcommand{\jouleperkelvinnp}{\joule\usk\reciprocal\kelvin}}
326 \CustomizeMathJax{\newcommand{\jouleperkilogram}{\joule\per\kilogram}}
327 \CustomizeMathJax{\newcommand{\jouleperkilogramnp}{\joule\usk\reciprocal\kilogram}}
328 \CustomizeMathJax{\newcommand{\coulombperkilogram}{\coulomb\per\kilogram}}
329 \CustomizeMathJax{\newcommand{\coulombperkilogramnp}{\coulomb\usk\reciprocal\kilogram}}
```

```
330 \CustomizeMathJax{\newcommand{\squaremetrepersecond}{\squaremetre\per\second}}
331 \CustomizeMathJax{\newcommand{\squaremetrepersecondnp}{%
332     \squaremetre\usk\reciprocal\second%
333 }}
334 \CustomizeMathJax{\newcommand{\squaremetrepersquaresecond}{%
335     \squaremetre\per\second\squared%
336 }}
337 \CustomizeMathJax{\newcommand{\squaremetrepersquaresecondnp}{%
338     \squaremetre\usk\second\rpsquared%
339 }}
340 \CustomizeMathJax{\newcommand{\kilogrammetrepersecond}{%
341     \kilogram\usk\metre\per\second%
342 }}
343 \CustomizeMathJax{\newcommand{\kilogrammetrepersecondnp}{%
344     \kilogram\usk\metre\usk\reciprocal\second%
345 }}
346 \CustomizeMathJax{\newcommand{\candelapersquaremetre}{\candela\per\squaremetre}}
347 \CustomizeMathJax{\newcommand{\candelapersquaremetrenp}{\candela\usk\rpsquare\metre}}
348 \CustomizeMathJax{\newcommand{\amperepermetre}{\ampere\per\metre}}
349 \CustomizeMathJax{\newcommand{\amperepermetrenp}{\ampere\usk\reciprocal\metre}}
350 \CustomizeMathJax{\newcommand{\joulepertesla}{\joule\per\tesla}}
351 \CustomizeMathJax{\newcommand{\jouleperteslanp}{\joule\usk\reciprocal\tesla}}
352 \CustomizeMathJax{\newcommand{\henrypermetre}{\henry\per\metre}}
353 \CustomizeMathJax{\newcommand{\henrypermetrenp}{\henry\usk\reciprocal\metre}}
354 \CustomizeMathJax{\newcommand{\kilogrampersecond}{\kilogram\per\second}}
355 \CustomizeMathJax{\newcommand{\kilogrampersecondnp}{\kilogram\usk\reciprocal\second}}
356 \CustomizeMathJax{\newcommand{\kilogrampersquaremetresecond}{%
357     \kilogram\per\squaremetre\usk\second%
358 }}
359 \CustomizeMathJax{\newcommand{\kilogrampersquaremetresecondnp}{%
360     \kilogram\usk\rpsquare\metre\usk\reciprocal\second%
361 }}
362 \CustomizeMathJax{\newcommand{\kilogrampersquaremetre}{\kilogram\per\squaremetre}}
363 \CustomizeMathJax{\newcommand{\kilogrampersquaremetrenp}{\kilogram\usk\rpsquare\metre}}
364 \CustomizeMathJax{\newcommand{\kilogrampermetre}{\kilogram\per\metre}}
365 \CustomizeMathJax{\newcommand{\kilogrampermetrenp}{\kilogram\usk\reciprocal\metre}}
366 \CustomizeMathJax{\newcommand{\joulepermolekelvin}{\joule\per\mole\usk\kelvin}}
367 \CustomizeMathJax{\newcommand{\joulepermolekelvinnp}{%
368     \joule\usk\reciprocal\mole\usk\reciprocal\kelvin%
369 }}
370 \CustomizeMathJax{\newcommand{\kilogramperkilomole}{\kilogram\per\kilo\mole}}
371 \CustomizeMathJax{\newcommand{\kilogramperkilomolenp}{%
372     \kilogram\usk\kilo\reciprocal\mole%
373 }}
374 \CustomizeMathJax{\newcommand{\kilogramsquaremetre}{\kilogram\usk\squaremetre}}
375 \CustomizeMathJax{\newcommand{\kilogramsquaremetrenp}{\kilogramsquaremetre}}
376 \CustomizeMathJax{\newcommand{\kilogrammetrepersquaresecond}{%
377     \kilogram\usk\metre\per\second\squared%
378 }}
379 \CustomizeMathJax{\newcommand{\kilogrammetrepersquaresecondnp}{%
380     \kilogram\usk\metre\usk\second\rpsquared%
381 }}
382 \CustomizeMathJax{\newcommand{\newtonpersquaremetre}{\newton\per\squaremetre}}
383 \CustomizeMathJax{\newcommand{\newtonpersquaremetrenp}{\newton\usk\rpsquare\metre}}
384 \CustomizeMathJax{\newcommand{\persquaremetresecond}{1\per\squaremetre\usk\second}}
385 \CustomizeMathJax{\newcommand{\persquaremetresecondnp}{%
386     \rpsquare\metre\usk\reciprocal\second%
387 }}
388 \CustomizeMathJax{\newcommand{\wattperkilogram}{\watt\per\kilogram}}
389 \CustomizeMathJax{\newcommand{\wattperkilogramnp}{\watt\usk\reciprocal\kilogram}}
```

390 \CustomizeMathJax{\newcommand{\wattpercubicmetre}{\watt\per\cubic\metre}}
391 \CustomizeMathJax{\newcommand{\wattpercubicmetrenp}{\watt\usk\rpcubic\metre}}
392 \CustomizeMathJax{\newcommand{\wattpersquaremetresteradian}{%
393   \watt\per\squaremetre\usk\steradian%
394 }}
395 \CustomizeMathJax{\newcommand{\wattpersquaremetresteradiannp}{%
396   \watt\usk\rpsquare\metre\usk\rp\steradian%
397 }}
398 \CustomizeMathJax{\newcommand{\jouleperkilogramkelvin}{\joule\per\kilogram\usk\kelvin}}
399 \CustomizeMathJax{\newcommand{\jouleperkilogramkelvinnp}{%
400   \joule\usk\reciprocal\kilogram\usk\reciprocal\kelvin%
401 }}
402 \CustomizeMathJax{\newcommand{\squaremetreperkilogram}{\squaremetre\per\kilogram}}
403 \CustomizeMathJax{\newcommand{\rpsquaremetreperkilogram}{%
404   \squaremetre\usk\reciprocal\kilogram%
405 }}
406 \CustomizeMathJax{\newcommand{\cubicmetreperkilogram}{\cubic\metre\per\kilogram}}
407 \CustomizeMathJax{\newcommand{\rpcubicmetreperkilogram}{%
408   \cubic\metre\usk\reciprocal\kilogram%
409 }}
410 \CustomizeMathJax{\newcommand{\newtonpermetre}{\newton\per\metre}}
411 \CustomizeMathJax{\newcommand{\newtonpermetrenp}{\newton\usk\reciprocal\metre}}
412 \CustomizeMathJax{\newcommand{\Celsius}{\unicode{x2103}}}
413 \CustomizeMathJax{\newcommand{\wattpermetrekelvin}{\watt\per\metre\usk\kelvin}}
414 \CustomizeMathJax{\newcommand{\wattpermetrekelvinnp}{%
415   \watt\usk\reciprocal\metre\usk\reciprocal\kelvin%
416 }}
417 \CustomizeMathJax{\newcommand{\newtonmetre}{\newton\usk\metre}
418 \CustomizeMathJax{\newcommand{\newtonmetrenp}{\newtonmetre}}}
419 \CustomizeMathJax{\newcommand{\squaremetrepercubicsecond}{%
420   \squaremetre\per\cubic\second%
421 }}
422 \CustomizeMathJax{\newcommand{\squaremetrepercubicsecondnp}{%
423   \squaremetre\usk\rpcubic\second%
424 }}
425 \CustomizeMathJax{\newcommand{\metrepersecond}{\metre\per\second}}
426 \CustomizeMathJax{\newcommand{\metrepersecondnp}{\metre\usk\reciprocal\second}}
427 \CustomizeMathJax{\newcommand{\joulepercubicmetre}{\joule\per\cubicmetre}}
428 \CustomizeMathJax{\newcommand{\joulepercubicmetrenp}{\joule\usk\rpcubic\metre}}
429 \CustomizeMathJax{\newcommand{\kilogrampercubicmetrecoulomb}{%
430   \kilogram\per\cubic\metre\usk\coulomb%
431 }}
432 \CustomizeMathJax{\newcommand{\kilogrampercubicmetrecoulombnp}{%
433   \kilogram\usk\rpcubic\metre\usk\reciprocal\coulomb%
434 }}
435 \CustomizeMathJax{\newcommand{\cubicmetrepersecond}{\cubicmetre\per\second}}
436 \CustomizeMathJax{\newcommand{\rpcubicmetrepersecond}{\cubicmetre\usk\reciprocal\second}}
437 \CustomizeMathJax{\newcommand{\kilogrampersecondcubicmetre}{%
438   \kilogram\per\second\usk\cubicmetre%
439 }}
440 \CustomizeMathJax{\newcommand{\kilogrampersecondcubicmetrenp}{%
441   \kilogram\usk\reciprocal\second\usk\rpcubic\metre%
442 }}
443 \end{warpMathJax}

File 452   **`lwarp-siunitx.sty`**

§ 561   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}[2022-02-15]
11
12 \ExplSyntaxOn
```

```
13 \cs_set_protected:Npn \siunitx_number_format:nN #1#2
14   {
15     \group_begin:
16       \bool_if:NTF \l_siunitx_number_parse_bool
17         {
18           \siunitx_number_parse:nN {#1} \l__siunitx_number_parsed_tl
19         \siunitx_number_process:NN \l__siunitx_number_parsed_tl \l__siunitx_number_parsed_tl
20           \tl_set:Nx \l__siunitx_number_outputted_tl
21             { \siunitx_number_output:N \l__siunitx_number_parsed_tl }
22         }
23         {
24           \tl_set:Nn \l__siunitx_number_outputted_tl
25           {
26             \LWR@subsingledollar{%                                lwarp
27                 \textbackslash( % space
28                 \LWR@HTMLsanitizedetokenized{%
29                     \detokenize{#1}%
30                 } \textbackslash)%                                lwarp
31             }%
32             {siunitx unparsed}%
33             {\ensuremath{#1}}%                                lwarp
34           }
35         }
36     \exp_args:NNNV \group_end:
37     \tl_set:Nn #2 \l__siunitx_number_outputted_tl
38   }
```

```
39 \cs_set_protected:Npn \__siunitx_compound_unparsed:n #1
40   {
41     \tl_if_blank:nF {#1}
42       { \seq_put_right:Nn \l__siunitx_compound_tmp_seq
43         {
44             \LWR@subsingledollar{%                          lwarp
45                 \textbackslash( % space
```

```
46                    \LWR@HTMLsanitizedetokenized{%
47                        \detokenize{#1}%
48                    } \textbackslash)%                      lwarp
49                }%
50                {siunitx unparsed}%
51                {\ensuremath{#1}}%                          lwarp
52            }
53        }
54    }
```

If not in a lateximage, always use text mode. Ignore current text font if resetting text family, series, and shape.

```
55 \cs_set_protected:Npn \__siunitx_print_aux:nn #1#2
56  {
57    \tl_if_empty:cTF { l__siunitx_print_ #1 _color_tl }
58      { \use:n }
59      { \exp_args:Nv \textcolor { l__siunitx_print_ #1 _color_tl } }
60        {
61            \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%       lwarp
62            {
63                \use:c
64                    {
65                    siunitx_print_
66                    \tl_use:c { l__siunitx_print_ #1 _mode_tl } :n
67                    }
68                    {#2}
69            }
70            {
71                \bool_lazy_all:nTF%      lwarp
72                    {
73                        {\l__siunitx_print_text_family_bool}
74                        {\l__siunitx_print_text_series_bool}
75                        {\l__siunitx_print_text_shape_bool}
76                    }
77                    {% No font control if reset-text-family/series/shape
78                        \use:c
79                            {
80                            siunitx_print_%                      lwarp
81                                text%                            lwarp
82                                :n%                              lwarp
83                            }%                                   lwarp
84                            {#2}%                                lwarp
85                        }
86                    {
87                        \LWR@textcurrentfont{%                   lwarp
88                            \use:c
89                                {
90                                siunitx_print_%                  lwarp
91                                    text%                        lwarp
92                                    :n%                          lwarp
93                                }%                               lwarp
94                                {#2}%                            lwarp
95                            }
96                        }
97                    }
98                }
99    }
```

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 is set appropri-
ately.

```
100 \newcommand*{\LWR@siunitx@complexrootstyle}{textrm}
101
102 \newcommand*{\LWR@siunitx@complexrmi}{\mathrm{i}}
103 \newcommand*{\LWR@siunitx@complexrmj}{\mathrm{j}}
104
105 \newcommand*{\LWR@siunitx@setcomplexroot}{%
106     \renewcommand*{\LWR@siunitx@complexrootstyle}{textit}%
107    \ifdefequal{\l__siunitx_complex_output_root_tl}{\LWR@siunitx@complexrmi}%
108         {\renewcommand*{\LWR@siunitx@complexrootstyle}{textrm}}%
109         {}%
110    \ifdefequal{\l__siunitx_complex_output_root_tl}{\LWR@siunitx@complexrmj}%
111         {\renewcommand*{\LWR@siunitx@complexrootstyle}{textrm}}%
112         {}%
113 }
```

```
114 \cs_set_protected:Npn \__siunitx_complex_format_auxii:n #1
115  {
116     \LWR@siunitx@setcomplexroot%                              lwarp
117     \__siunitx_complex_format_units:n {#1}
118     \tl_if_empty:NF \l__siunitx_complex_real_tl
119     { \exp_after:wN \__siunitx_complex_drop_exponent:nnnnnnn \l__siunitx_complex_real_tl }
120    \exp_after:wN \__siunitx_complex_format_sign:nnnnnnn \l__siunitx_complex_img_tl
121     \tl_set:Nx \l__siunitx_complex_tmp_tl
122       { \siunitx_number_output:NN \l__siunitx_complex_img_tl \q_nil }
123    \exp_after:wN \__siunitx_complex_extract_exponent:w \l__siunitx_complex_tmp_tl \q_stop
124     \tl_set:Nx \l__siunitx_complex_tmp_tl
125       {
126         \bool_lazy_or:nnTF
127           {
128             \bool_lazy_and_p:nn
129               { \l_siunitx_number_bracket_ambiguous_bool }
130               { ! \tl_if_empty_p:N \l__siunitx_complex_exp_tl }
131           }
132           {
133             ! \bool_lazy_any_p:n
134               {
135                 { \tl_if_blank_p:n {#1} }
136                 { \tl_if_empty_p:N \l__siunitx_complex_real_tl }
137                 { \tl_if_empty_p:N \l__siunitx_complex_img_tl }
138               }
139           }
140           { \__siunitx_complex_format_bracket:n }
141           { \use:n }
142           {
143             \siunitx_number_output:N \l__siunitx_complex_real_tl
144             \exp_not:V \l__siunitx_complex_sign_tl
145             \bool_if:NF \l__siunitx_complex_root_after_bool
146               {
147                 \InlineClass{\LWR@siunitx@complexrootstyle}%    lwarp
148                   {
149                       \exp_not:V \l__siunitx_complex_output_root_tl
150                   }
151               }
152             \exp_not:V \l__siunitx_complex_tmp_tl
153             \bool_if:NT \l__siunitx_complex_root_after_bool
154               {
```

```
155                    \InlineClass{\LWR@siunitx@complexrootstyle}%      lwarp
156                        {
157                            \exp_not:V \l__siunitx_complex_output_root_tl
158                        }
159                    }
160                }
161            \exp_not:V \l__siunitx_complex_exp_tl
162        }
163    }
```

{⟨*1: deg/min/sec character*⟩} {⟨*2: ?*⟩} {⟨*3: ?*⟩} {⟨*4: integer part of angle*⟩} {⟨*5: decimal point character*⟩} {⟨*6: decimal part of angle*⟩} {⟨*7: ?*⟩} {⟨*8: ?*⟩}

If not in a lateximage, print a simplified verison without the box measurement things which conflict with lwarp:

```
164 \cs_set_protected:Npn \__siunitx_angle_arc_print_auxii:nw
165   #1#2 \q_nil #3 \q_nil #4 \q_nil #5 \q_nil #6 \q_nil #7 \q_nil #8 \q_stop
166   {
167     \mode_if_math:TF
168       { \bool_set_true:N \l__siunitx_angle_tmp_bool }
169       { \bool_set_false:N \l__siunitx_angle_tmp_bool }
170     \siunitx_print_number:n {#2#3#4}
171     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%      lwarp
172     {%                                                  lwarp
173         \tl_if_blank:nTF {#6}
174         { \__siunitx_angle_arc_print_auxvi:n {#1} }
175         {
176             \hbox_set:Nn \l__siunitx_angle_marker_box
177             {
178                 \__siunitx_angle_arc_print_auxiii:n
179                 { \siunitx_print_number:n {#5} }
180             }
181             \hbox_set:Nn \l__siunitx_angle_unit_box
182             {
183                 \__siunitx_angle_arc_print_auxiii:n
184                 {
185                     \siunitx_unit_format:nN {#1} \l__siunitx_angle_tmp_tl
186                     \siunitx_print_unit:V \l__siunitx_angle_tmp_tl
187                     \skip_horizontal:n { -\scriptspace }
188                 }
189             }
190             \dim_compare:nNnTF { \box_wd:N \l__siunitx_angle_marker_box } >
191             { \box_wd:N \l__siunitx_angle_unit_box }
192             {
193                 \__siunitx_angle_arc_print_auxiv:NN
194                 \l__siunitx_angle_marker_box
195                 \l__siunitx_angle_unit_box
196             }
197             {
198                 \__siunitx_angle_arc_print_auxiv:NN
199                 \l__siunitx_angle_unit_box
200                 \l__siunitx_angle_marker_box
201             }
202             \hbox_set_to_wd:Nnn \l__siunitx_angle_marker_box
203             \l__siunitx_angle_tmp_dim
204             {
205                 \hbox_overlap_right:n
206                 { \box_use_drop:N \l__siunitx_angle_marker_box }
207                 \hbox_overlap_right:n
```

```
208                        { \box_use_drop:N \l__siunitx_angle_unit_box }
209                        \tex_hfil:D
210                    }
211                    \box_use:N \l__siunitx_angle_marker_box
212                    \skip_horizontal:N \scriptspace
213                    \siunitx_print_number:n {#6}
214                }
215        }%
```

{⟨*1: deg/min/sec character*⟩} {⟨*2: ?*⟩} {⟨*3: ?*⟩} {⟨*4: integer part of angle*⟩} {⟨*5: decimal point character*⟩} {⟨*6: decimal part of angle*⟩} {⟨*7: ?*⟩} {⟨*8: ?*⟩}

```
216    {%           lwarp: not in a lateximage, simplify for HTML
217        \tl_if_blank:nTF {#6}
218        { \__siunitx_angle_arc_print_auxvi:n {#1} }
219        {
220            \__siunitx_angle_arc_print_auxiii:n
221            {
222                \siunitx_print_number:n {#5}
223            }
224            \__siunitx_angle_arc_print_auxiii:n
225            {
226                \siunitx_unit_format:nN {#1} \l__siunitx_angle_tmp_tl
227                \siunitx_print_unit:V \l__siunitx_angle_tmp_tl
228            }
229            \siunitx_print_number:n {#6}
230        }
231    }%           lwarp
232  }
```

If not in a `lateximage`, print a simple inline fraction, avoiding the use of SVG math:

```
233 \cs_set_protected:Npn \__siunitx_print_text_fraction:Nnn #1#2#3
234  {
235    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}% lwarp
236    {%
237      \ensuremath
238      {
239        #1
240          { \mbox { \__siunitx_print_text_replace:n {#2} } }
241          { \mbox { \__siunitx_print_text_replace:n {#3} } }
242      }
243    }%
244    {%                                                       lwarp
245        { \mbox { \__siunitx_print_text_replace:n {#2} } }%   lwarp
246        /%                                                   lwarp
247        { \mbox { \__siunitx_print_text_replace:n {#3} } }%   lwarp
248    }%     lwarp
249  }
```

If not in a `lateximage`, print a `\textsubscript`:

```
250 \cs_set_protected:Npn \__siunitx_unit_format_qualifier_subscript:
251  {
252    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%        lwarp
253    {%
254        \__siunitx_unit_format_font:
255        \tl_set:Nx \l__siunitx_unit_part_tl
256        {
```

```
257             \c__siunitx_unit_math_subscript_tl
258             {
259                 \exp_not:V \l_siunitx_unit_font_tl
260                 { \exp_not:V \l__siunitx_unit_part_tl }
261             }
262         }
263     }
264     {%      lwarp simplified for HTML:
265         \__siunitx_unit_format_font:
266         \tl_set:Nx \l__siunitx_unit_part_tl
267         {
268             \textsubscript
269             {
270                 \exp_not:V \l_siunitx_unit_font_tl
271                 { \exp_not:V \l__siunitx_unit_part_tl }
272             }
273         }
274     }
275 }

276 \cs_set_protected:Npn \siunitx_quantity:nn #1#2
277 {
278     \group_begin:
279         \siunitx_unit_options_apply:n {#2}
280         \tl_if_blank:nTF {#1}
281         {
282             \siunitx_unit_format:nN {#2} \l__siunitx_quantity_unit_tl
283             \siunitx_print_unit:V \l__siunitx_quantity_unit_tl
284         }
285         {
286             \bool_if:NTF \l_siunitx_number_parse_bool
287             { \__siunitx_quantity_parsed:nn {#1} {#2} }
288             {
289                 \tl_set:Nn \l__siunitx_quantity_number_tl {
290                     \LWR@subsingledollar{%                       lwarp
291                         \textbackslash( % space
292                         \LWR@HTMLsanitizedetokenized{%
293                             \detokenize{#1}%
294                         } \textbackslash)%                       lwarp
295                     }%
296                     {siunitx unparsed}%
297                     {\ensuremath{#1}}%                           lwarp
298                 }
299                 \siunitx_unit_format:nN {#2} \l__siunitx_quantity_unit_tl
300                 \siunitx_quantity_print:VV
301                     \l__siunitx_quantity_number_tl \l__siunitx_quantity_unit_tl
302             }
303         }
304     \group_end:
305 }
```

\cancel for HTML does not work yet.

```
306 \newcommand*{\LWR@siunitx@nocancel}[1]{%
307     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
308         {\cancel{#1}}%  SVG
309         {#1}%           HTML
310 }
311
312 \AtBeginDocument{
```

```
313 \__siunitx_unit_set_symbolic:Npnn \cancel
314   { }
315 %   { \__siunitx_unit_parse_special:n { \cancel } }
316   { \__siunitx_unit_parse_special:n { \LWR@siunitx@nocancel } }%    lwarp
317 }
```

For HTML, use a simple unaligned \num:

```
318 \newcommand{\LWR@HTML@tablenum}[2][]{\num[#1]{#2}}
319 \LWR@formatted{tablenum}
```

For HTML, the S column is simplified to a c column. Keys are set locally, allowing drop-exponent, etc.

```
320 \AtBeginDocument{
321 \HTMLnewcolumntype{S}[1][]{>{\begingroup\sisetup{#1}}c<{\endgroup}}
322 }
```

To define simplified units for HTML:

\HTMLDeclareSIUnit [⟨*options*⟩] {⟨*name*⟩} {⟨*definition*⟩}

```
323 \NewDocumentCommand{\HTMLDeclareSIUnit}{o +m m}
324 {
325     \ifcsdef{ __siunitx_unit_ \token_to_str:N #2 :w }
326         {}
327         {
328             \PackageError{lwarp}
329                 {%
330                     First~use\MessageBreak
331                         \space\space\protect\DeclareSIUnit{
332                             \token_to_str:N#2}{...}\MessageBreak
333                     before~using\MessageBreak
334                         \space\space\protect\HTMLDeclareSIUnit{
335                             \token_to_str:N#2}{...}%
336                 }
337                 {%
338                     See~the~Lwarp~manual~section~about~special~cases,~
339                     regarding~siunitx.%
340                 }
341         }
342     \csNewCommandCopycs
343         { __orig_siunitx_unit_ \token_to_str:N #2 :w }
344         { __siunitx_unit_ \token_to_str:N #2 :w }
345     \DeclareSIUnit[#1]{#2}
346         {
347             \ifnumcomp{\value{LWR@leateximagedepth}}{>}{0}
348                 {\csuse{ __orig_siunitx_unit_ \token_to_str:N #2 :w }}
349                 {#3}
350         }
351 }
```

```
352 \ExplSyntaxOff
```

HTML versions for existing units:

```
353 \AtBeginDocument{
354 \HTMLDeclareSIUnit\celsius{\LWR@siunitx@textcelsius}
```

```
355 \HTMLDeclareSIUnit\arcminute{\LWR@siunitx@textprime}
356 \HTMLDeclareSIUnit\arcsecond{\LWR@siunitx@textdblprime}
357 \HTMLDeclareSIUnit\elementarycharge{\textit{e}}
358 %
359 \HTMLDeclareSIUnit\clight{\text{\textit{c}\textsubscript{0}}}
360 \HTMLDeclareSIUnit\bohr{\text{\textit{a}\textsubscript{0}}}
361 \HTMLDeclareSIUnit\electronmass{\text{\textit{m}\textsubscript{e}}}
362 \HTMLDeclareSIUnit\hartree{\text{\textit{E}\textsubscript{h}}}
363 \HTMLDeclareSIUnit\planckbar{\LWR@siunitx@textplanckbar}
364 }% \AtBeginDocument
```

Initial options:

```
365 \AtBeginDocument{
366 \sisetup{
367     per-mode=symbol,     % fraction is not seen by pdftotext
368     angle-symbol-degree = {\LWR@siunitx@textdegree},
369     angle-symbol-minute = {\LWR@siunitx@textprime} ,
370     angle-symbol-second = {\LWR@siunitx@textdblprime} ,
371 }
372 }
```

Load late paches for lltjp-siunitx:

```
373 \AtBeginDocument{
374 \ifdef{\ltj@allalchar}
375     {\LWR@origRequirePackage{lwarp-lltjp-siunitx}}
376     {}
377 }
```

For MathJax:

```
378 \LWR@origRequirePackage{lwarp-common-mathjax-siunitx}
379
380 \CustomizeMathJax{\let\unit\si}
381 \CustomizeMathJax{\let\qty\SI}
382 \CustomizeMathJax{\let\qtylist\SIlist}
383 \CustomizeMathJax{\let\qtyrange\SIrange}
384 \CustomizeMathJax{\let\numproduct\num}
385 \CustomizeMathJax{\let\qtyproduct\SI}
386 \CustomizeMathJax{\let\complexnum\num}
387 \CustomizeMathJax{\newcommand{\complexqty}[3][]{(\complexnum{#2})\si{#3}}}
```

Pass range-phrase to common-mathjax-siunitx:

```
388 \ExplSyntaxOn
389 \AtBeginDocument{
390 \edef\LWR@siunitx@rangephrase{\l_siunitx_range_phrase_tl}
391 \expandafter\CustomizeMathJax\expandafter{%
392     \expandafter\def\expandafter\LWRsiunitxrangephrase%
393     \expandafter{\LWR@siunitx@rangephrase}%
394 }
395 }
396 \ExplSyntaxOff
```

File 453   **lwarp-siunitx-v2.sty**

§ 562   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 S and s columns are rendered as simple c columns, although key settings
⚠ tabular   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 re-
stored to its original. Similar for \mbox. svg math is created explicitly when neces-
sary, 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@lateximagedepth}}{>}{0}% lwarp
```

```
220 {% lateximage
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 }

File 454   **`lwarp-common-mathjax-siunitx.sty`**

§ 563   Package   **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 455   **`lwarp-skmath.sty`**

§ 564   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 \CustomizeMathJax{\renewcommand{\ln}{\LWRskmathlog{ln}}}
144 \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 \CustomizeMathJax{\newcommand{\LWRskmathminnostar}[2][]{%
166     \ifblank{#1}%
167         {\operatorname{\LWRskmathminname}}%
168         {%
169             \underset%
170                 {\mathchoice{\mathclap{#1}}{#1}{#1}{#1}}%
171                 {\operatorname{\LWRskmathminname}}%
172         }%
```

```
173     \ifblank{#2}{}{\left\{{#2}\right\}}%
174 }}
```

\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 456   `lwarp-slantsc.sty`

§ 565   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 457   `lwarp-slashed.sty`

§ 566   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 458   `lwarp-soul.sty`

§ 567   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 459   **`lwarp-soulpos.sty`**

§ 568   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 460   **`lwarp-soulutf8.sty`**

§ 569   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 461   **`lwarp-splitbib.sty`**

§ 570   Package   # splitbib

(*Emulates or patches code by* Nicolas Markey.)

splitbib (*Pkg*)   splitbib is patched for use by lwarp.

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 462   **`lwarp-splitidx.sty`**

§ 571   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 695.17 in the lwarp-patch-memoir package for the \@@wrspindexhyp macro as an example.

1 \LWR@ProvidesPackagePass{splitidx}[2016/02/18]

```
2 \catcode`\_=12%
3 \xpatchcmd{\newindex}
4     {\jobname-#2.idx}
5     {\jobname-#2_html.idx}
6     {}
7     {\LWR@patcherror{splitidx}{@newindex}}
8 \catcode`\_=8%
```

Patched to use lwarp's automatic indexing counter instead of \thepage:

```
9 \renewcommand*{\@wrsindex}[2][]{%
10   \ifx\relax#1\relax
11     \if@splitidx
12       \@wrsindex[idx]{#2}%
13     \else
14       \def\@tempa{#2}%
15       \if@verbindex\@onelevel@sanitize\@tempa\fi
16       \@wrindex{\@tempa}%
17     \fi
18   \else
19     \def\@tempa{#2}%
20     \csname index@#1@hook\endcsname
21 %     \expandafter\ifx\csname @@wrsindex\endcsname\relax
22     \addtocounter{LWR@autoindex}{1}%                        lwarp
23 %       \@@@wrsindex{#1}{{\@tempa}{\thepage}}%
24     \@@@wrsindex{#1}{{\@tempa}{\arabic{LWR@autoindex}}}%
25 %     \else
26 %       \def\@tempb{\@@wrsindex{#1}}%
27 %       \expandafter\@tempb\@tempa||\\%
28 %     \fi
```

The label is assigned after the file write to avoid conflict with cleveref.

```
29     \label{LWRindex-\arabic{LWR@autoindex}}%    lwarp
30     \endgroup
31     \@esphack
32   \fi
33 }
```

lwarp defines sectioning commands with xparse, so the below patches are done as temporary redefinitions instead of being \let.

```
34 \xpatchcmd{\printsubindex}
35     {\let\section\subsection}
36     {\renewcommand*{\section}{\subsection}}
37     {}
38     {\LWR@patcherror{splitidx}{printsubindex-section}}
39
40 \xpatchcmd{\printsubindex}
41     {\let\chapter\section}
42     {\renewcommand*{\chapter}{\section}}
```

```
43    {}
44    {\LWR@patcherror{splitidx}{printsubindex-chapter}}
45
46 \xpatchcmd{\printsubindex}
47    {\let\@makechapterhead\section}
48    {\def\@makechapterhead{\section}}
49    {}
50    {\LWR@patcherror{splitidx}{printsubindex-chapter}}
```

File 463   **lwarp-srcltx.sty**

§ 572   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 464   **lwarp-srctex.sty**

§ 573   Package   # srctex

srctex (*Pkg*)   srctex is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{srctex}[2006/11/12]
2 \LWR@origRequirePackage{lwarp-srcltx}

File 465   **lwarp-stabular.sty**

§ 574   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 466   `lwarp-stackengine.sty`

§ 575   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]`

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 \renewcommand*\savestack[2]{%
50   \xdef\sv@name{\stack@macro@name{#1}}%
51   \@ifundefined{\sv@name content}{%
52     \expandafter\newsavebox\expandafter{\csname\sv@name content\endcsname}%
53   }{}%
54   \begingroup%      lwarp
55   \LWR@restoreorigformatting%      lwarp
56   \RenewDocumentEnvironment{lateximage}{s o s o o d()}{}{}% lwarp: inside group
57   \expandafter\LWR@gsavebox\csname\sv@name content\endcsname{#2}%
58   \expandafter\gdef\expandafter#1\expandafter{%
59       \expandafter\begin\expandafter{lateximage\expandafter}%      lwarp
60       \expandafter\usebox\expandafter%
61       {\csname\sv@name content\endcsname}%
62       \expandafter\end\expandafter{lateximage\expandafter}%        lwarp
63   }%
64   \endgroup%      lwarp
65 }
```

File 467 **`lwarp-stackrel.sty`**

§ 576   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 468   **`lwarp-statex2.sty`**

§ 577   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%
}
```

for HTML output: 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 469   **`lwarp-statistics.sty`**

§ 578   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 \StartDefiningTabulars%      lwarp
4 \cs_set_protected_nopar:Nn \__statistics_table_make:nn {
5     \int_compare:nT
6           { 0 < \l__statistics_table_maxcols_int
7               = \l__statistics_nbvals_int } {
8        \__statistics_table_end:
9        \tl_use:N \l__statistics_table_sep_tl
10       \__statistics_table_start:
11    }
12    \int_incr:N \l__statistics_nbvals_int
13    \int_incr:N \l__statistics_currange_int
14    \fp_add:Nn \l__statistics_curtotal_fp { #2 }
15    \__statistics_set_if_shown:N \l_tmpa_bool
16    \tl_set:Nx \l_tmpa_tl {
17        \exp_not:n { & \tl_set:Nn \currentcolumn } {
18            \int_use:N \l__statistics_currange_int
19        }
20    }
21    \bool_if:NTF \l_tmpa_bool {
22        \tl_put_right:Nn \l_tmpa_tl
23            {\__statistics_table_shown_format:n}
24    }{
25        \tl_put_right:Nn \l_tmpa_tl
```

```
26                  {\__statistics_table_hidden_format:n}
27          }
28          \seq_put_right:Nn \l__statistics_store_values_seq { #1 }
29          \bool_if:NT \l__statistics_table_values_bool {
30              \tl_put_right:Nx \l__statistics_table_values_tl {
31                  \exp_not:V \l_tmpa_tl {
32                      \exp_not:n {
33                          \__statistics_table_values_format:n { #1 }
34                      }
35                  }
36              }
37          }
38          \seq_put_right:Nx \l__statistics_store_counts_seq { \fp_eval:n {#2} }
39          \bool_if:NT \l__statistics_table_counts_bool {
40              \tl_put_right:Nx \l__statistics_table_counts_tl {
41                  \exp_not:V \l_tmpa_tl {
42                      \exp_not:n {
43                          \__statistics_table_counts_format:n {
44                              { \__statistics_table_allcounts_format:n { #2 } }
45                          }
46                      }
47                  }
48              }
49          }
50          \bool_if:NT \l__statistics_table_icc_bool {
51              \tl_put_right:Nx \l__statistics_table_icc_tl {
52                  \exp_not:V \l_tmpa_tl {
53                      \exp_not:n { \__statistics_table_icc_format:n }
54                      {
55                          \exp_not:n{ \__statistics_table_allcounts_format:n }
56                          { \fp_use:N \l__statistics_curtotal_fp }
57                      }
58                  }
59              }
60          }
61          \bool_if:NT \l__statistics_table_dcc_bool {
62              \tl_put_right:Nx \l__statistics_table_dcc_tl {
63                  \exp_not:V \l_tmpa_tl {
64                      \exp_not:n { \__statistics_table_dcc_format:n }
65                      {
66                          \exp_not:n{ \__statistics_table_allcounts_format:n }
67                          {
68                              \fp_eval:n {
69                                  \l__statistics_total_fp
70                                      - \l__statistics_curtotal_fp
71                                      + #2
72                              }
73                          }
74                      }
75                  }
76              }
77          }
78          \fp_set:Nn \l__statistics_table_curICF_fp {
79              round(\l__statistics_curtotal_fp
80                      / \l__statistics_total_fp,
81                  \l__statistics_table_round_int)
82          }
83          \bool_if:NT \l__statistics_table_frequencies_bool {
84              \tl_put_right:Nx \l__statistics_table_frequencies_tl {
85                  \exp_not:V \l_tmpa_tl {
```

```
 86                    \exp_not:n { \__statistics_table_frequencies_format:n }
 87                    {
 88                        \exp_not:n{ \__statistics_table_allfreqs_format:n }
 89                        {
 90                            \fp_eval:n {
 91                                \l__statistics_table_curICF_fp
 92                                    - \l__statistics_table_prevICF_fp
 93                            }
 94                        }
 95                    }
 96                }
 97            }
 98        }
 99        \bool_if:NT \l__statistics_table_icf_bool {
100            \tl_put_right:Nx \l__statistics_table_icf_tl {
101                \exp_not:V \l_tmpa_tl {
102                    \exp_not:n { \__statistics_table_icf_format:n }
103                    {
104                        \exp_not:n{ \__statistics_table_allfreqs_format:n }
105                            { \fp_to_decimal:N \l__statistics_table_curICF_fp }
106                    }
107                }
108            }
109        }
110        \bool_if:NT \l__statistics_table_dcf_bool {
111            \tl_put_right:Nx \l__statistics_table_dcf_tl {
112                \exp_not:V \l_tmpa_tl {
113                    \exp_not:n { \__statistics_table_dcf_format:n }
114                    {
115                        \exp_not:n{ \__statistics_table_allfreqs_format:n }
116                        {
117                            \fp_eval:n {
118                                1 - \l__statistics_table_prevICF_fp
119                            }
120                        }
121                    }
122                }
123            }
124        }
125        \fp_set_eq:NN
126            \l__statistics_table_prevICF_fp
127            \l__statistics_table_curICF_fp
128 }
129 \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.

```
130 \cs_set_protected_nopar:Nn \__statistics_table_end: {
131    \tl_set:Nx \l__statistics_table_preamble_tl {
132 %        \exp_not:n { \begin{array}[ }
133        \exp_not:n {\begin{tabular}[ }%       lwarp
134            \exp_not:V \l__statistics_table_valign_tl
135        \exp_not:n { ] }
136                { \exp_not:V \l__statistics_table_headcoltype_tl
137                    \prg_replicate:nn { \l__statistics_nbvals_int }
138                    { \exp_not:V \l__statistics_table_coltype_tl } }
139    }
140    \seq_clear:N \l__statistics_table_contents_seq
141    \clist_map_inline:nn { values, counts, icc, dcc, frequencies, icf, dcf } {
```

```
142          \bool_if:cT { l__statistics_table_##1_bool } {
143               \seq_put_right:Nv
144                         \l__statistics_table_contents_seq
145                         { l__statistics_table_##1_tl }
146          }
147      }
148 %     $
149     \tl_use:N \l__statistics_table_preamble_tl
150          \hline%        lwarp
151 %          \l__statistics_table_preline_tl
152          \seq_use:Nn
153                  \l__statistics_table_contents_seq
154                  { \l__statistics_table_newline_tl }
155          \\
156 %          \l__statistics_table_postline_tl
157          \hline%        lwarp
158 %     \end{array}$
159     \end{tabular}%         lwarp
160 }
```

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:

```
161 \cs_set_protected_nopar:Nn \__statistics_table_start: {
162     \int_zero:N \l__statistics_nbvals_int
163     \clist_pop:NNT \l__statistics_table_maxcols_clist \l_tmpa_tl {
164         \int_set:Nn \l__statistics_table_maxcols_int { \l_tmpa_tl }
165     }
166     \clist_map_inline:nn { values, counts, frequencies, icc, icf, dcc, dcf } {
167         \tl_set:cx { l__statistics_table_##1_tl } {
168 %            \exp_not:N \ensuremath {
169                 \exp_not:N \hbox {
170                     \exp_not:c { l__statistics_table_##1_name_tl }
171                 }
172 %            }
173         }
174     }
175 }
```

For the first row. Redefined to use text output:

```
176 \RenewDocumentCommand \__statistics_IN:w { m u{;} u{;} m } {
177 %     \ensuremath{ \left#1 \num{#2} \mathbin{;} \num{#3} \right#4 }
178     #1 #2 ; #3 #4%         lwarp
179 }
180
181 \__statistics_setup:nn { table } {
182 %     values/format = \ensuremath{#1},
183     values/format = {#1},%         lwarp
184 }
```

Added \ExplSyntaxOn/Off to avoid errors. (In once instance, a double subscript error appeared.)

```
185 \RenewDocumentCommand \StatsGraph { +O{} +m +O{} } {
```

```
186    \group_begin:
187    \int_gincr:N \g__statistics_graph_last_int
188    \tl_set:Nx \l_tmpa_tl {
189        \exp_not:n { g__statistics_graph_xstep_ }
190        \int_use:N \g__statistics_graph_last_int
191        \exp_not:n { _tl }
192    }
193    \tl_if_exist:cTF { \l_tmpa_tl } {
194        \fp_gset:Nn \g__statistics_graph_xstep_fp
195            { \tl_use:c {\l_tmpa_tl} }
196    }{
197        \fp_gset:Nn \g__statistics_graph_xstep_fp { \c_one_int }
198    }
199    \__statistics_setup:nn { graph } { #1, #3 }
200    \tl_if_single:nTF { #2 } {
201        \cs_if_exist:NF #2 { #2 }
202        \tl_set_eq:NN \l__statistics_data_tl #2
203    }{
204        \tl_set:Nn \l__statistics_data_tl { #2 }
205    }
206    \fp_zero:N \l__statistics_graph_maxheight_fp
207    \fp_set:Nn \l__statistics_graph_minvalue_fp {inf}
208    \fp_set:Nn \l__statistics_graph_maxvalue_fp {-inf}
209    \fp_zero:N \l__statistics_total_fp
210    \int_zero:N \l__statistics_nbvals_int
211    \bool_set_true:N \l__statistics_graph_allranges_bool
212    \keyval_parse:NNV
213            \__statistics_graph_prepare:n
214            \__statistics_graph_prepare:nn
215            \l__statistics_data_tl
216    \tl_clear:N \l__statistics_graph_tikzdata_tl
217    \tl_clear:N \l__statistics_graph_tikzinfo_tl
218    \int_zero:N \l__statistics_currange_int
219    \bool_if:NTF \l__statistics_graph_allranges_bool {
220        \bool_if:NTF \l__statistics_graph_cumulative_bool {
221 \ExplSyntaxOn%     lwarp
222            \__statistics_graph_dopicture_cumulative:
223 \ExplSyntaxOff%    lwarp
224        }{
225 \ExplSyntaxOn%     lwarp
226            \__statistics_graph_dopicture_hist:
227 \ExplSyntaxOff%    lwarp
228        }
229    }{
230 \ExplSyntaxOn%     lwarp
231        \__statistics_graph_dopicture_comb:
232 \ExplSyntaxOff%    lwarp
233    }
234    \iow_now:Nx \@auxout {
235        \exp_not:n {
236            \ExplSyntaxOn
237            \tl_gset:cn
238        }
239        {
240            \exp_not:n {g__statistics_graph_xstep_}
241            \int_use:N \g__statistics_graph_last_int
242            \exp_not:n {_tl}
243        }
244        {
245            \fp_to_decimal:N \g__statistics_graph_xstep_fp
```

```
246        }
247        \exp_not:n {
248            \ExplSyntaxOff
249        }
250    }
251    \group_end:
252 }
253
254 \ExplSyntaxOff
```

File 470 **lwarp-statmath.sty**

§ 579   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 471   `lwarp-steinmetz.sty`

§ 580   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 472   **`lwarp-stfloats.sty`**

§ 581   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 473   **`lwarp-struktex.sty`**

§ 582   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 474 **`lwarp-subcaption.sty`**

§ 583   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 475   **`lwarp-subfig.sty`**

§ 584   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 paragraph tags to be generated, resulting in subfigures appearing on the following lines instead of all on a single line.

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 \long\def\sf@@@subfloat#1[#2][#3]#4{%
3 \begin{minipage}{\linewidth}% lwarp

4 \IfValueTF{#2}{%
5   \LWR@setlatestname{#2}%
6 }{%
7   \IfValueTF{#3}{%
8   \LWR@setlatestname{#3}%
9   }{}%
10 }%
11 \LWR@stoppars% lwarp
12   \@ifundefined{FBsc@max}{}%
13   {\FB@readaux{\let\FBsuboheight\relax}}%
14   \@tempcnta=\@ne
15   \if@minipage
16   \@tempcnta=\z@
17   \else\ifdim \lastskip=\z@ \else
18   \@tempcnta=\tw@
19   \fi\fi
20   \ifmaincaptiontop
21   \sf@top=\sf@nearskip
22   \sf@bottom=\sf@farskip
23   \else
24   \sf@top=\sf@farskip
25   \sf@bottom=\sf@nearskip
26   \fi
27   \leavevmode

28 %   \setbox\@tempboxa \hbox{#4}%
29 %   \@tempdima=\wd\@tempboxa
30 %   \@ifundefined{FBsc@max}{}%
31 %   {\global\advance\Xhsize-\wd\@tempboxa
32 %   \dimen@=\ht\@tempboxa
33 %   \advance\dimen@\dp\@tempboxa
34 %   \ifdim\dimen@>\FBso@max
35 %   \global\FBso@max\dimen@
36 %   \fi}%

Do not use boxes, which interfere with lateximages:

37 %   \vtop%
38   \bgroup
39 %   \vbox%
40   \bgroup
41   \ifcase\@tempcnta
42   \@minipagefalse
43   \or
44 %   \vskip\sf@top
45   \or
46   \ifdim \lastskip=\z@ \else
47 %   \@tempskipb\sf@top\relax\@xaddvskip
48   \fi
49   \fi

```
50            \sf@ifpositiontop{%
51              \ifx \@empty#3\relax \else
52                \sf@subcaption{#1}{#2}{#3}%
53 %              \vskip\sf@capskip
54 %              \vskip\sf@captopadj
55            \fi\egroup
56 %          \hrule width0pt height0pt depth0pt
57            \LWR@startpars% lwarp
58 %  \box\@tempboxa
59              #4
60            \LWR@stoppars% lwarp
61          }{%
62          \LWR@startpars% lwarp
63          \@ifundefined{FBsc@max}%
64              {
65 %  \box\@tempboxa
66              #4
67            }%
68            {\ifx\FBsuboheight\relax
69 %              \box\@tempboxa
70              #4
71            \else
72 %              \vbox to \FBsuboheight{\FBafil\box\@tempboxa\FBbfil}%
73              #4
74            \fi}%
75          \LWR@stoppars% lwarp
76            \egroup
77            \ifx \@empty#3\relax \else
78 %              \vskip\sf@capskip
79 %              \hrule width0pt height0pt depth0pt
80            \sf@subcaption{#1}{#2}{#3}%
81          \fi
82          }%
83 %          \vskip\sf@bottom
84        \egroup
85        \@ifundefined{FBsc@max}{}%
86            {\addtocounter{FRobj}{-1}%
87            \ifnum\c@FRobj=0\else
88              \subfloatrowsep
89            \fi}%
90        \ifmaincaptiontop\else
91          \global\advance\@nameuse{c@\@captype}\m@ne
92        \fi
93 \end{minipage}% lwarp
94 \LWR@startpars% lwarp
95    \endgroup\ignorespaces%
96 }%
```

\sf@subcaption $\{\langle 1\ type\rangle\}\ \{\langle 2\ lof\ entry\rangle\}\ \{\langle 3\ caption\rangle\}$

```
97 \long\def\sf@subcaption#1#2#3{%
98 \LWR@stoppars% lwarp
99    \ifx \relax#2\relax \else
100      \bgroup
101        \let\label=\@gobble
102        \let\protect=\string
103        \def\@subcaplabel{%
104          \caption@lstfmt{\@nameuse{p@#1}}{\@nameuse{the#1}}}%
105        \sf@updatecaptionlist{#1}{#2}{\the\value{\@captype}}{\the\value{#1}}%
106      \egroup
```

```
107    \fi
108    \bgroup
109      \ifx \relax#3\relax
110        \let\captionlabelsep=\relax
111      \fi
112 %    \setbox0\vbox{%
113 %        \hb@xt@\the\@tempdima{%
114 %
115 % %            \hss
116 % %          \parbox[t]{\the\@tempdima}{%
117 %          \caption@make
118 %              {\@nameuse{sub\@captype name}}%
119 %              {\@nameuse{thesub\@captype}}%
120 %              {#3}
121 % %    }%
122 % %            \hss
123 %    }
124 %    }%
125    \@ifundefined{FBsc@max}%
126 %        {\box0}%
127        {
128 %  \parbox[t]{\the\@tempdima}{%
129 \LWR@traceinfo{sfsubcap B1}%    lwarp
130            \LWR@figcaption%    lwarp
131            \caption@make
132                {\@nameuse{sub\@captype name}}%
133                {\@nameuse{thesub\@captype}}%
134                {\LWR@isolate{#3}}%

135            \endLWR@figcaption%    lwarp
136 \LWR@traceinfo{sfsubcap B2}%        lwarp
137 %    }%
138          }%
139        {\dimen@\ht0%
140          \advance\dimen@\dp0%
141          \ifdim\dimen@>\FBsc@max
142            \global\FBsc@max\dimen@
143          \fi
144          \FB@readaux{\let\FBsubcheight\relax}%
145          \ifx\FBsubcheight\relax
146            \def\next{
147 %  \parbox[t]{\the\@tempdima}
148              }%
149          \else
150            \def\next{
151 %  \parbox[t][\FBsubcheight][t]{\the\@tempdima}
152              }%
153          \fi
154 %        \vbox{%
155 %            \hb@xt@\the\@tempdima{%
156
157 %              \hss
158 %            \next{%
159 \LWR@traceinfo{sfsubcap C1}% lwarp
160                \caption@make
161                    {\@nameuse{sub\@captype name}}%
162                    {\@nameuse{thesub\@captype}}%
163                    {#3}
164 \LWR@traceinfo{sfsubcap C1}% lwarp
165 %    }%
```

```
166 %                    \hss
167
168 %   }
169 %            }
170         }%
171   \egroup
172 \LWR@startpars% lwarp
173 }
```

\subfloat@label   Patches for \sf@sub@label:

```
174 \xpretocmd{\subfloat@label}
175     {\LWR@ensuredoingapar}
176     {}
177     {\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>:

```
178 \renewcommand{\sf@subref}[1]{%
179     \LWR@subnewref{#1}{sub@#1}%
180 }
```

\sf@@subref   {⟨*label*⟩}

The starred version uses the printed sub@<label> which is stored as if it were a page number:

```
181 \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*⟩}

```
182 \LetLtxMacro\LWR@orig@newsubfloat\@newsubfloat
183
184 \def\@newsubfloat[#1]#2{%
185 \LWR@orig@newsubfloat[#1]{#2}%
186 \renewcommand{\l@sub#2}[2]{\hypertocfloat{2}{sub#2}{\ext@sub#2}{##1}{##2}}%
187 }
```

Pre-defined for figures and tables:

\l@subfigure   {⟨*text*⟩} {⟨*pagenum*⟩}

```
188 \renewcommand{\l@subfigure}[2]{\hypertocfloat{2}{subfigure}{lof}{#1}{#2}}
```

\l@subtable   {⟨*text*⟩} {⟨*pagenum*⟩}

```
189 \renewcommand{\l@subtable}[2]{\hypertocfloat{2}{subtable}{lot}{#1}{#2}}
```

File 476 **`lwarp-subfigure.sty`**

§ 585 Package **subfigure**

subfigure (*Pkg*)   subfigure is emulated by subfig.

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 477 **`lwarp-subsupscripts.sty`**

§ 586 Package **subsupscripts**

(*Emulates or patches code by* RICCARDO BRESCIANI.)

subsupscripts (*Pkg*)   subsupscripts is used as-is for SVG math, and is emulated for MATHJAX.

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 478   **lwarp-supertabular.sty**

§ 587   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 479   `lwarp-svg.sty`

§ 588   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 480   `lwarp-swfigure.sty`

§ 589   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 481 **`lwarp-sympytex.sty`**

§ 590   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@lateximagedepth}}{>}{0}%
6         {}%
7         {%
8             \LWR@forcenewpage%
9             \LWR@atbeginverbatim{verbatim}%
10        }%
11 }
12
13 \AfterEndEnvironment{sympyblock}{%
14     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
15        {}%
16        {%
17            \LWR@afterendverbatim%
18        }%
19 }
20
21 }
```

File 482   **`lwarp-syntonly.sty`**

§ 591   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 483 **`lwarp-tabfigures.sty`**

§ 592 Package **tabfigures**

tabfigures (*Pkg*)   tabfigures is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{tabfigures}[2012/01/24]

File 484 **`lwarp-tablefootnote.sty`**

§ 593 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 485 **`lwarp-tabls.sty`**

§ 594 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 486 **`lwarp-tabularx.sty`**

§ 595 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 487   **`lwarp-tabulary.sty`**

§ 596   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 488   **`lwarp-tagpdf.sty`**

§ 597   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.

```
1 \LWR@ProvidesPackagePass{tagpdf}[2022-08-24]
```

```
2 \ExplSyntaxOn
3
4 \keys_define:nn { __tag / struct }
5   {
6     alt .code:n      = % Alt property
7       {
8 %        \str_set_convert:Noon
9 %          \l__tag_tmpa_str
10 %          { #1 }
11 %          { default }
12 %          { utf16/hex }
13 %        \__tag_prop_gput:cnx
14 %          { g__tag_struct_\int_eval:n {\c@g__tag_struct_abs_int}_prop }
15 %          { Alt }
16 %          { <\l__tag_tmpa_str> }
17         \gdef\LWR@ThisAltText{\detokenize\expandafter{#1}}%      lwarp
18       },
19   }
20
21 \ExplSyntaxOff
```

The package is deactivated on load, and also each time \tagpdfsetup is used.

```
22 \LWR@tagpdf@deactivate
```

File 489   **lwarp-tagpdf-base.sty**

§ 598   Package   **tagpdf-base**

(*Emulates or patches code by* ULRIKE FISCHER.)

tagpdf-base (*Pkg*)   **tagpdf-base** is patched for use by lwarp.

```
1 \LWR@ProvidesPackagePass{tagpdf-base}[2022-08-24]
```

```
2 \ExplSyntaxOn
3
4 \newcommand*{\LWR@tagpdf@deactivate}{
5    \keys_set:nn { __tag / setup } {
6        activate-space = false ,
7        activate-mc = false ,
8        activate-tree = false ,
9        activate-struct = false
10    }
11 }
12
13 \RenewDocumentCommand \tagpdfsetup { m }{
14    \keys_set:nn { __tag / setup } { #1 }
15    \LWR@tagpdf@deactivate
16 }
```

```
17
18 \RenewDocumentCommand \tagmcbegin { m }
19   {
20 %    \tag_mc_begin:n {#1}
21     \keys_set:nn { __tag / mc } {#1}
22   }
23
24 \RenewDocumentCommand \tagmcend {   }
25   {
26 %    \tag_mc_end:
27     \ThisAltText{}%     lwarp
28   }
29
30 \RenewDocumentCommand \tagmcuse { m }
31   {
32 %    \tag_mc_use:n {#1}
33   }
34
35 \RenewDocumentCommand \tagstructbegin { m }
36   {
37     \keys_set:nn { __tag / struct} { #1 }%     lwarp
38 %    \tag_struct_begin:n {#1}
39   }
40
41 \RenewDocumentCommand \tagstructend {   }
42   {
43 %  \tag_struct_end:
44   \ThisAltText{}%        lwarp
45   }
46
47 \RenewDocumentCommand \tagstructuse { m }
48   {
49 %    \tag_struct_use:n {#1}
50   }
51
52 \ExplSyntaxOff
```

File 490   **`lwarp-tagpdf-mc-code-generic.sty`**

§ 599   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     alt .code:n      = % Alt property
8       {
9 %        \str_set_convert:Noon
10 %          \l__tag_tmpa_str
11 %          { #1 }
```

```
12 %          { default }
13 %          { utf16/hex }
14 %       \tl_put_right:Nn \l__tag_mc_key_properties_tl { /Alt~< }
15 %       \tl_put_right:No \l__tag_mc_key_properties_tl { \l__tag_tmpa_str>~ }
16       \gdef\LWR@ThisAltText{\detokenize\expandafter{#1}}%       lwarp
17     },
18   }
19
20 \ExplSyntaxOff
```

File 491   **`lwarp-tagpdf-mc-code-lua.sty`**

§ 600   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     alt .code:n      = % Alt property
7       {
8 %        \str_set_convert:Noon
9 %          \l__tag_tmpa_str
10 %          { #1 }
11 %          { default }
12 %          { utf16/hex }
13 %       \tl_put_right:Nn \l__tag_mc_key_properties_tl { /Alt~< }
14 %       \tl_put_right:No \l__tag_mc_key_properties_tl { \l__tag_tmpa_str>~ }
15 %       \lua_now:e
16 %         {
17 %           ltx.__tag.func.store_mc_data
18 %             (
19 %           \__tag_get_mc_abs_cnt:,"alt","/Alt~<\str_use:N \l__tag_tmpa_str>"
20 %             )
21 %         }
22       \gdef\LWR@ThisAltText{\detokenize\expandafter{#1}}%       lwarp
23     },
24   }
25
26 \ExplSyntaxOff
```

File 492   **`lwarp-tascmac.sty`**

§ 601   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 493   **lwarp-tcolorbox.sty**

§ 602   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}[2020/04/28]

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.

```
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 only temporarily defined where appropriate, so the HTML version is defined globally via \newcommand instead of \renewcommand.

```
121 \newcommand{\tcblower}{
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 }
```

Starred and unstarred \tcbline are simple \hrules.

```
130 \AtBeginDocument{
131 \ifdef{\tcbline}{
132     \newcommand*{\LWR@sub@tcbline}{%
133         \begin{BlockClass}{hrule}
134         \end{BlockClass}
135     }
136     \newcommand{\LWR@HTML@tcbline}{\@ifstar\LWR@sub@tcbline\LWR@sub@tcbline}
137     \LWR@formatted{tcbline}
138 }{}
139 }
140
141 \newcommand{\LWR@HTML@tcbox}[2][]{
142     \LWR@printpendingfootnotes
143     \LWR@tcolorbox@dostartfloat%
144     \begingroup
145     \tcb@layer@inc
146     \tcb@apply@box@options{#1}
147 %     \tcbset{title=,#1}
148     \boolfalse{LWR@havetcblower}
149     \LWR@tcolorboxstart{tcolorbox inlineminipage}
150     \tcb@insert@before@upper%
151     #2
152     \ifbool{LWR@havetcblower}{%
153         \tcb@insert@after@lower%
154     }{%
155         \tcb@insert@after@upper%
156     }%
157     \end{BlockClass}
158     \LWR@printpendingfootnotes
159     \end{BlockClass}
160     \tcb@layer@dec%
161     \endgroup%
162     \LWR@tcolorbox@dostartfloat%
163     \global\booltrue{LWR@minipagethispar}%
164 }
165 \LWR@formatted{tcbox}
166
167 \appto\LWR@restoreMathJaxformatting{%
168     \renewcommand{\tcbox}[2][]{#2}%
169 }
```

Patches for the subtitle, which is placed inside a <div> of class tcolorboxsubtitle.

```
170 \xpatchcmd{\tcbsubtitle}
171     {\begingroup}
172     {\begingroup\let\kvtcb@title\relax\begin{BlockClass}{tcolorboxsubtitle}}
173     {}
174     {\LWR@patcherror{tcolorbox}{tcbsubtitle}}
```

```
175
176 \xpatchcmd{\tcbsubtitle}
177     {\endgroup}
178     {\end{BlockClass}\endgroup}
179     {}
180     {\LWR@patcherror{tcolorbox}{tcbsubtitleB}}
```

\tcboxfit is the same as \tcbox.

```
181 \AtBeginDocument{
182     \ifdef{\tcboxfit}{%
183         \let\LWR@HTML@tcboxfit\tcbox%
184         \LWR@formatted{tcboxfit}
185     }{}
186 }
```

\tcbtitle is patched to support the text font.

```
187 \LetLtxMacro\LWR@HTML@tcbtitle\tcbtitle
188 \xpatchcmd{\LWR@HTML@tcbtitle}
189     {\kvtcb@before@title\tcbtitletext}
190     {\kvtcb@before@title\LWR@textcurrentfont{\LWR@textcurrentcolor{\tcbtitletext}}}
191     {}
192     {\LWR@patcherror{tcolorbox}{LWR@HTML@tcbtitle}}
193 \LWR@formatted{tcbtitle}
```

List-of:

```
194 \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.

```
195 \AtBeginDocument{
196 \pgfkeysifdefined{/tcb/libload/theorems}{
197
198     \def\LWR@HTML@tcb@hack@amsmath{%
199         \PackageError{lwarp}
200             {%
201                 tcolorbox ``math'', ``ams equation'', and related\MessageBreak
202                 are not supported.\MessageBreak
203                 \protect\tcboxmath\space and
204                 \protect\tcbhighmath\space are emulated.\MessageBreak
205                 Enter 'H' for possible solutions%
206             }
207             {%
208                 Remove tcolorbox math-related options, and instead\MessageBreak
209                 use the usual math environments inside each tcolorbox.%
210             }
211     }
212     \LWR@formatted{tcb@hack@amsmath}
213
214     % Cause an error if using math:
215     \tcbset{%
216         math upper/.style={before upper*=\tcb@hack@amsmath,after upper*=$},%
217         math lower/.style={before lower*=\tcb@hack@amsmath,after lower*=$},%
218     }
219
220     \appto\LWR@restoreorigformatting{%
```

```
221     \tcbset{%
222         math upper/.style={before upper*=$\displaystyle,after upper*=$},%
223         math lower/.style={before lower*=$\displaystyle,after lower*=$},%
224     }%
225     }
226
227     \newcommand{\LWR@HTML@tcboxmath}[2][]{#2}
228     \LWR@formatted{tcboxmath}
229     \newcommand{\LWR@HTML@tcbhighmath}[2][]{#2}
230     \LWR@formatted{tcbhighmath}
231     \appto\LWR@restoreMathJaxformatting{%
232         \renewcommand{\tcboxmath}[2][]{#2}%
233         \renewcommand{\tcbhighmath}[2][]{#2}%
234     }
235 }{}% theorems loaded
236 }% AtBeginDocument
```

For MATHJAX:

```
237 \CustomizeMathJax{\newcommand{\tcbset}[1]{}}
238 \CustomizeMathJax{\newcommand{\tcbsetforeverylayer}[1]{}}
239 \CustomizeMathJax{\newcommand{\tcbox}[2][]{\boxed{\text{#2}}}}
240 \CustomizeMathJax{\newcommand{\tcboxfit}[2][]{\boxed{#2}}}
241 \CustomizeMathJax{\newcommand{\tcblower}{}}
242 \CustomizeMathJax{\newcommand{\tcbline}{}}
243 \CustomizeMathJax{\newcommand{\tcbtitle}{}}
244 \CustomizeMathJax{\newcommand{\tcbsubtitle}[2][]{\mathrm{#2}}}
245 \CustomizeMathJax{\newcommand{\tcboxmath}[2][]{\boxed{#2}}}
246 \CustomizeMathJax{\newcommand{\tcbhighmath}[2][]{\boxed{#2}}}
```

File 494   **`lwarp-tensor.sty`**

§ 603   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 495 **`lwarp-termcal.sty`**

§ 604   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 496   **`lwarp-textarea.sty`**

§ 605   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 497   **`lwarp-textcomp.sty`**

§ 606   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.

§ 606.1 **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.

§ 606.2 **Package loading**

for HTML output: 1 \LWR@ProvidesPackagePass{textcomp}[2017/04/05]

§ 606.3 **HTML symbols**

For HTML, use HTML entities or direct Unicode, depending on the engine.

\AtBeginDocument improves support for LuaLATEX and XƎLATEX.

§ 606.3.1 **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
```

§ 606.3.2 **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}
```

§606.4 **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 498 **`lwarp-textfit.sty`**

§ 607   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 499   **`lwarp-textpos.sty`**

§ 608   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 500   `lwarp-theorem.sty`

§ 609   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.

---

1 \LWR@ProvidesPackagePass{theorem}[2014/10/28]

§ 609.1   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 \gdef\theoremstyle#1{%
4     \@ifundefined{th@#1}{\@warning
5         {Unknown theoremstyle '#1'. Using 'plain'}%
6         \theorem@style{plain}%
7          \renewcommand{\LWR@newtheoremstyle}{plain}% lwarp
8        }%
9     {%
10        \theorem@style{#1}%
11        \renewcommand{\LWR@newtheoremstyle}{#1}% lwarp
12      }%
13      \begingroup
```

```
14        \csname th@\the\theorem@style \endcsname
15      \endgroup}
```

Patched to remember the style for this theorem type, and set it later when the environment is started.

```
16 \gdef\@xnthm#1#2[#3]{%
17    \expandafter\@ifdefinable\csname #1\endcsname
18    {%
19    \csedef{LWR@thmstyle#1}{\LWR@newtheoremstyle}% lwarp
20    \@definecounter{#1}\@newctr{#1}[#3]%
21    \expandafter\xdef\csname the#1\endcsname
22      {\expandafter \noexpand \csname the#3\endcsname
23       \@thmcountersep \@thmcounter{#1}}%
24    \def\@tempa{\global\@namedef{#1}}%
25    \expandafter \@tempa \expandafter{%
26      \csname th@\the \theorem@style
27           \expandafter \endcsname \the \theorem@bodyfont
28      \@thm{#1}{#2}}%
29    \global \expandafter \let \csname end#1\endcsname \@endtheorem
30    \AtBeginEnvironment{#1}{\edef\LWR@thisthmstyle{\@nameuse{LWR@thmstyle#1}}}% lwarp
31    }}
32
33 \gdef\@ynthm#1#2{%
34    \expandafter\@ifdefinable\csname #1\endcsname
35    {
36    \csedef{LWR@thmstyle#1}{\LWR@newtheoremstyle}% lwarp
37    \@definecounter{#1}%
38    \expandafter\xdef\csname the#1\endcsname{\@thmcounter{#1}}%
39    \def\@tempa{\global\@namedef{#1}}\expandafter \@tempa
40     \expandafter{\csname th@\the \theorem@style \expandafter
41     \endcsname \the\theorem@bodyfont \@thm{#1}{#2}}%
42    \global \expandafter \let \csname end#1\endcsname \@endtheorem
43    \AtBeginEnvironment{#1}{\edef\LWR@thisthmstyle{\@nameuse{LWR@thmstyle#1}}}% lwarp
44    }}
45
46 \gdef\@othm#1[#2]#3{%
47  \expandafter\ifx\csname c@#2\endcsname\relax
48    \@nocounterr{#2}%
49  \else
50    \expandafter\@ifdefinable\csname #1\endcsname
51    {
52    \csedef{LWR@thmstyle#1}{\LWR@newtheoremstyle}% lwarp
53    \expandafter \xdef \csname the#1\endcsname
54      {\expandafter \noexpand \csname the#2\endcsname}%
55    \def\@tempa{\global\@namedef{#1}}\expandafter \@tempa
56     \expandafter{\csname th@\the \theorem@style \expandafter
57     \endcsname \the\theorem@bodyfont \@thm{#2}{#3}}%
58    \global \expandafter \let \csname end#1\endcsname \@endtheorem
59    \AtBeginEnvironment{#1}{\edef\LWR@thisthmstyle{\@nameuse{LWR@thmstyle#1}}}% lwarp
60    }%
61  \fi}
```

§ 609.2   **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.

```
62 \gdef\th@plain{%
63   \def\@begintheorem##1##2{%
64       \item[
65           \InlineClass{theoremheader}{##1\ ##2}
66       ]
67     }%
68 \def\@opargbegintheorem##1##2##3{%
69   \item[
70       \InlineClass{theoremheader}{##1\ ##2\ (##3)}
71   ]
72   }
73 }
74
75 \gdef\th@break{%
76   \def\@begintheorem##1##2{%
77     \item[
78         \InlineClass{theoremheader}{##1\ ##2}\newline%
79     ]
80     }%
81 \def\@opargbegintheorem##1##2##3{%
82   \item[
83       \InlineClass{theoremheader}{##1\ ##2\ (##3)}\newline
84   ]
85     }
86 }
87
88 \gdef\th@marginbreak{%
89   \def\@begintheorem##1##2{
90     \item[
91         \InlineClass{theoremheader}{##2 \qquad ##1}\newline
92     ]
93     }%
94 \def\@opargbegintheorem##1##2##3{%
95   \item[
96       \InlineClass{theoremheader}{##2 \qquad ##1\ %
97       (##3)}\newline
98     ]
99     }
100 }
101
102 \gdef\th@changebreak{%
103   \def\@begintheorem##1##2{
104     \item[
105       \InlineClass{theoremheader}{##2\ ##1}\newline
106     ]
107     }%
108 \def\@opargbegintheorem##1##2##3{%
109     \item[
110       \InlineClass{theoremheader}{ ##2\ ##1\ %
```

```
111        (##3)}\newline
112     ]
113     }
114 }
115
116 \gdef\th@change{%
117    \def\@begintheorem##1##2{
118      \item[
119        \InlineClass{theoremheader}{##2\ ##1}
120      ]
121      }%
122 \def\@opargbegintheorem##1##2##3{%
123      \item[
124        \InlineClass{theoremheader}{##2\ ##1\ (##3)}
125      ]
126      }
127 }
128
129 \gdef\th@margin{%
130    \def\@begintheorem##1##2{
131      \item[
132        \InlineClass{theoremheader}{##2 \qquad ##1}
133      ]
134      }%
135 \def\@opargbegintheorem##1##2##3{%
136      \item[
137            \InlineClass{theoremheader}{##2 \qquad ##1\ (##3)}
138      ]
139      }
140 }
```

Patched for css:

```
141 \gdef\@thm#1#2{\refstepcounter{#1}%
142 \LWR@forcenewpage% lwarp


143     \LWR@printpendingfootnotes%                        lwarp


144     \BlockClass{theorembody\LWR@thisthmstyle}% lwarp
145     \trivlist
146     \@topsep \theorempreskipamount              % used by first \item
147     \@topsepadd \theorempostskipamount          % used by \@endparenv
148     \@ifnextchar [%
149     {\@ythm{#1}{#2}}%
150     {\@begintheorem{#2}{\csname the#1\endcsname}\ignorespaces}}
151
152 \gdef\@endtheorem{%
153 \endtrivlist


154     \LWR@printpendingfootnotes%                     lwarp


155 \endBlockClass
156 }
```

File 501   `lwarp-thinsp.sty`

§ 610   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 502   `lwarp-thm-listof.sty`

§ 611   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 `  \def\thmtlo@newentry{%`
4 `    \csdef{l@\thmt@envname}##1##2{\hypertocfloat{1}{figure}{lof}{##1}{##2}}%`
5 `  }`
6 `}{% earlier than v0.72`
7 `    \xpatchcmd{\listoftheorems}`
8 `        {%`
9 `            \@xa\protected@edef\csname l@\thmt@envname\endcsname{%`
10 `                \@nx\@dottedtocline{1}{1.5em}{\@nx\thmt@listnumwidth}%`
11 `            }%`
12 `        }`
13 `        {%`
14 `            \csdef{l@\thmt@envname}##1##2{\hypertocfloat{1}{figure}{lof}{##1}{##2}}%`
15 `        }`
16 `        {}`
17 `        {\LWR@patcherror{thm-listof}{listoftheorems}}`
18
19 `    \xpatchcmd{\thmt@mklistcmd}`
20 `        {%`
21 `            \@xa\protected@edef\csname l@\thmt@envname\endcsname{%`
22 `                \@nx\@dottedtocline{1}{1.5em}{\@nx\thmt@listnumwidth}%`

```
23              }%
24          }
25          {%
26          \csdef{l@\thmt@envname}##1##2{\hypertocfloat{1}{figure}{lof}{##1}{##2}}%
27          }
28          {}
29          {\LWR@patcherror{thm-listof}{thmt@mklistcmd}}
30 }
```

File 503   **`lwarp-thm-restate.sty`**

§ 612   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 \xpatchcmd\thmt@restatable
3     {\@ifstar}
4     {\edef\LWR@thisthmstyle{#2}\@ifstar}
5     {}
6     {\LWR@patcherror{thm-restate}{thmt@restatable}}
```

File 504   **`lwarp-thmbox.sty`**

§ 613   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 505   **`lwarp-thmtools.sty`**

§ 614   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 506   **`lwarp-threadcol.sty`**

§ 615   Package   # threadcol

threadcol (*Pkg*)   threadcol is ignored.

for HTML output: `1 \LWR@ProvidesPackageDrop{threadcol}[2013/01/06]`

`2 \newcommand{\setthreadname}[1]{}`

File 507   **`lwarp-threeparttable.sty`**

§ 616   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 508   **lwarp-threeparttablex.sty**

§ 617   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 depend-
ing 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.

for HTML output: 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 509   **lwarp-thumb.sty**

§ 618   Package   # thumb

thumb (*Pkg*)   thumb is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{thumb}[1997/12/24]

2 \newcommand*{\Overviewpage}{}

```
3 \newlength{\thumbheight}
4 \newlength{\thumbwidth}
```

File 510   **`lwarp-thumbs.sty`**

§ 619   Package   **thumbs**

thumbs (*Pkg*)   thumbs is ignored.

for HTML output: `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 511   **`lwarp-tikz.sty`**

§ 620   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 TikZ babel library temporarily
changes catcodes back to normal for TikZ'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}}
```

TikZ 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 512   **lwarp-tikz-imagelabels.sty**

§ 621   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 513 **`lwarp-titleps.sty`**

§ 622 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 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 514   **`lwarp-titleref.sty`**

§ 623   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 515   **`lwarp-titlesec.sty`**

§ 624   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 \PackageInfo{lwarp}{Using the lwarp version of package 'titlesec'.}%
2 \ProvidesPackage{lwarp-titlesec}[2016/03/21]
3
4 \newbool{LWR@loadtitleps}
5 \boolfalse{LWR@loadtitleps}
6
7 \DeclareOption{pagestyles}{
8      \booltrue{LWR@loadtitleps}
9 }
10
11 \DeclareOption*{}
12
13 \ProcessOptions\relax
14
15 \ifbool{LWR@loadtitleps}{
16     \RequirePackage{lwarp-titleps}
17 }{}
```

\titlelabel   {⟨*label-format*⟩}

```
18 \newcommand*{\titlelabel}[1]{}
```

\titleformat*   {⟨*command*⟩} {⟨*format*⟩}

\titleformat   {⟨*command*⟩} [⟨*shape*⟩] {⟨*format*⟩} {⟨*label*⟩} {⟨*sep*⟩} {⟨*begfore*⟩} [⟨*after*⟩]

```
19 \newcommand\titleformat{%
20   \@ifstar{\ttl@format@s}%
21        {\ttl@format@i}}
22 \newcommand{\ttl@format@s}[1]{}
23 \NewDocumentCommand{\ttl@format@i}{m o m m m m o}{}
```

\chaptertitlename

```
24 \@ifundefined{@chapapp}{\let\@chapapp\chaptername}{}
25 \newcommand\chaptertitlename{\@chapapp}
```

\titlespacing   * {⟨*command*⟩} {⟨*left*⟩} {⟨*before*⟩} {⟨*after*⟩} [⟨*right*⟩]

```
26 \NewDocumentCommand{\titlespacing}{s m m m m o}{}
```

\filright

```
27 \newcommand*{\filright}{}
```

\filcenter

```
28 \newcommand*{\filcenter}{}
```

\filleft

```
29 \newcommand*{\filleft}{}
```

\fillast

```
30 \newcommand*{\fillast}{}
```

\filinner

```
31 \newcommand*{\filinner}{}
```

\filouter

```
32 \newcommand*{\filouter}{}
```

\wordsep

```
33 \newcommand\wordsep{\fontdimen\tw@\font \@plus
34   \fontdimen\thr@@\font \@minus \fontdimen4\font}
```

\titleline   * [⟨*align*⟩] {⟨*material*⟩}

```
35 \NewDocumentCommand{\titleline}{s o m}{}
```

\titlerule   [⟨*height*⟩]

```
36 \providecommand*\titlerule{\@ifstar{\ttl@row}{\ttl@rule}}
37 \newcommand*{\ttl@rule}[1][]{}
38 \newcommand*{\ttl@row}[2][]{}
```

\iftitlemeasuring   {⟨*true*⟩} {⟨*false*⟩}

```
39 \newcommand{\iftitlemeasuring}[2]{#2}
```

\assignpagestyle   {⟨*command*⟩} {⟨*pagestyle*⟩}

```
40 \newcommand{\assignpagestyle}[2]{#2}
```

\titleclass   {⟨*name*⟩} [⟨*startlevel*⟩] {⟨*class*⟩} [⟨*cmd*⟩]

```
41 \NewDocumentCommand{\titleclass}{m o m o}{}
```

File 516 **`lwarp-titletoc.sty`**

§ 625   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 517 **`lwarp-titling.sty`**

§ 626   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 69.8.

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}}%
```

$$\text{\textbackslash thefootnote} \Rightarrow \text{\textbackslash nameuse\{arabic\}\{footnote\}, or}$$
$$\text{\textbackslash thefootnote} \Rightarrow \text{\textbackslash nameuse\{fnsymbol\}\{footnote\}}$$

Redefine the footnote text:

```
36    \long\def\@makefntext##1{%
```

Make the footnote mark and some extra horizontal space for the tags:

```
37        \makethanksmark~%
```

$$\text{\textbackslash makethanksmark} \Rightarrow \text{\textbackslash thanksfootmark} \Rightarrow \text{\textbackslash tamark} \Rightarrow$$
$$\text{\textbackslash @thefnmark} \Rightarrow \text{\textbackslash itshape a (or similar)}$$

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 69.7.

```
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 518 **`lwarp-tocbasic.sty`**

§ 627   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 519   **`lwarp-tocbibind.sty`**

§ 628   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]   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.

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 520   **`lwarp-tocdata.sty`**

§ 629   Package   # tocdata

(*Emulates or patches code by* BRIAN DUNN.)

tocdata (*Pkg*)   **tocdata** is patched for use by lwarp.

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 521 **`lwarp-tocenter.sty`**

§ 630 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 522 **`lwarp-tocloft.sty`**

§ 631 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 523   **lwarp-tocstyle.sty**

§ 632   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 524   **`lwarp-todo.sty`**

§ 633   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 525   **lwarp-todonotes.sty**

§ 634   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 526 **lwarp-topcapt.sty**

§ 635   Package   **topcapt**

topcapt (*Pkg*)   topcapt is emulated.

for HTML output: 1 \LWR@ProvidesPackageDrop{topcapt}[2004/12/11]

2 \LetLtxMacro\topcaption\caption

File 527 **lwarp-tram.sty**

§ 636   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 528 **`lwarp-transparent.sty`**

§ 637 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 Ǝ LᴬTᴇX! Note that transparent does not work with X Ǝ LᴬTᴇX.

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 529 **`lwarp-trimclip.sty`**

§ 638 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 530   **`lwarp-trivfloat.sty`**

§ 639   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}{}}
```

§ 639.1 **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 531   **`lwarp-truncate.sty`**

§ 640   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 532   **`lwarp-turnthepage.sty`**

§ 641   Package   **turnthepage**

turnthepage (*Pkg*)   turnthepage is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{turnthepage}[2011/03/24]

2 \newcommand{\turnthepage}{}

File 533   **`lwarp-twoup.sty`**

§ 642   Package   **twoup**

twoup (*Pkg*)   twoup is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{twoup}[2007/02/26]

```
2 \newcommand{\cleartolastpage}{}
```

File 534 **`lwarp-txfonts.sty`**

§ 643 Package # txfonts

(*Emulates or patches code by* YOUNG RYU.)

txfonts (*Pkg*) txfonts is used as-is for svg math, and is emulated for MATHJAX.

```
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 535 **`lwarp-txgreeks.sty`**

§ 644 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.

```
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 536   **`lwarp-typearea.sty`**

§ 645   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 537   **`lwarp-typicons.sty`**

§ 646   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 538   **`lwarp-ulem.sty`**

§ 647   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 539 **`lwarp-umoline.sty`**

§ 648   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 540   `lwarp-underscore.sty`

§ 649   Package   **underscore**

underscore (*Pkg*)   underscore is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{underscore}[2006/09/13]

File 541   `lwarp-unicode-math.sty`

§ 650   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 542 **`lwarp-units.sty`**

§ 651   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 543   `lwarp-unitsdef.sty`

§ 652   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 544   **`lwarp-upgreek.sty`**

§ 653   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 545   **`lwarp-upref.sty`**

§ 654   Package   # upref

upref (*Pkg*)   upref is ignored.

for HTML output: Discard all options for lwarp-upref:

`1 \LWR@ProvidesPackageDrop{upref}[2007/03/14]`

File 546   **`lwarp-url.sty`**

§ 655   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 547   **`lwarp-ushort.sty`**

§ 656   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 548   **`lwarp-uspace.sty`**

§ 657   Package   # uspace

uspace (*Pkg*)   uspace is ignored.

for HTML output: 1 `\LWR@ProvidesPackageDrop{uspace}[2016/11/06]`

File 549   **`lwarp-varioref.sty`**

§ 658   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 550 **`lwarp-verse.sty`**

§ 659   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
may already be set by the user for print output. New lengths   \HTMLvleftskip
\vleftmargini (*Len*)   and \HTMLleftmargini are provided to control the margins in HTML output. These
\HTMLvleftskip (*Len*)   new lengths may be set by the user before any verse environment, and persist
\HTMLleftmargini (*Len*)   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 op-
tion) 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 format-
ting.

```
6 \LWR@restoreoriglists%
```

verse (*Pkg*)
memoir (*Cls*)
\flagverse

The **verse** or **memoir** packages can place stanza numbers to the left with their \flagverse command. The following does not allow them to go into the left 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 551   **lwarp-versonotes.sty**

§ 660   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 552 **`lwarp-vertbars.sty`**

§ 661   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 553   **`lwarp-vmargin.sty`**

§ 662   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 554   `lwarp-vowel.sty`

§ 663   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 555   `lwarp-vpe.sty`

§ 664   Package   **vpe**

vpe (*Pkg*)   vpe is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{vpe}[2012/04/18]

File 556   `lwarp-vwcol.sty`

§ 665   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 557 **`lwarp-wallpaper.sty`**

§ 666   Package   # wallpaper

(*Emulates or patches code by* Michael H.F. Wilkinson.)

wallpaper (*Pkg*)   wallpaper is ignored.

for HTML output: 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 558   **lwarp-watermark.sty**

§ 667   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 559   **lwarp-widetable.sty**

§ 668   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 560   **lwarp-widows-and-orphans.sty**

§ 669   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 561 **`lwarp-witharrows.sty`**

§ 670 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}[2019/12/27]`

```
2 \ifbool{mathjax}{
3     % For the hidden print version in the HTML:
4     \newcommand{\Arrow}[2][]{}
5     \newcommand{\unicode}[1]{}
6     \NewDocumentEnvironment { DispWithArrows } { ! d < > ! O { } +b}
7         {
8             \IfValueTF{#1}{
9                 \begin{displaymath}
10                 #1 \left\lbrace
11                 \begin{align}
12                 #3
13                 \end{align}
14                 \right .
15                 \end{displaymath}
16             }{
17                 \begin{displaymath}
18                 \begin{align}
19                 #3
20                 \end{align}
21                 \end{displaymath}
22             }
23         }
24         {}
25     \NewDocumentEnvironment { 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 }{
45     % If not MathJax, use SVG images.
46     \BeforeBeginEnvironment{WithArrows}{\global\booltrue{LWR@unknownmathsize}}
47     \BeforeBeginEnvironment{DispWithArrows}{%
```

```
48          \begin{BlockClass}{displaymathnumbered}%
49          \begin{lateximage}%
50      }
51      \AfterEndEnvironment{DispWithArrows}{\end{lateximage}\end{BlockClass}}
52      \BeforeBeginEnvironment{DispWithArrows*}{%
53          \begin{BlockClass}{displaymath}%
54          \begin{lateximage}%
55      }
56      \AfterEndEnvironment{DispWithArrows*}{\end{lateximage}\end{BlockClass}}
57 }
58
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 562 **`lwarp-wrapfig.sty`**

§ 671   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 563   **`lwarp-wrapfig2.sty`**

§ 672   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
25 \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
43 \IfPackageLoadedWithOptionsTF{wrapfig2}{WFold}
44 {% v4.0:
45     \NewDocumentEnvironment{wraptext}%
46         {O{l}    D||{0.5\columnwidth} D<>{0}   D(){figure}}%
47     {%
48         \wrapfloat{#4}[]{#1}[]{#2}%
49         \tcolorbox%
50     }
51     {%
52         \endtcolorbox%
53         \endwrapfloat%
54         \ignorespaces%
55     }
56 }{}
57
58 \IfPackageLoadedWithOptionsTF{wrapfig2}{WFfive}
59 {% 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 564   **lwarp-xbmks.sty**

§ 673   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 565   **lwarp-xcolor.sty**

§ 674   Package   # xcolor

(*Emulates or patches code by* DR. UWE KERN.)

xcolor (*Pkg*)   xcolor is supported by lwarp.

§ 674.1 **Limitations**

\colorboxBlock and \fcolorboxBlock \colorboxBlock and \fcolorboxBlock are provided for increased HTML compatibility, 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 tables \rowcolors is supported, except that the optional argument is ignored so far.

colored text and boxes \textcolor, \colorbox, and \fcolorbox are supported.

\color and \pagecolor \color and \pagecolor are ignored. Use css or \textcolor where possible.

§ 674.2 **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 89. 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 532.

\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 1223.
>
> **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 1223.
>
> **HTML lateximage:** Remembers and reuses the print version.

\colorboxBlock:

> **Print:** Becomes \colorbox.
>
> **HTML:** Newly defined by lwarp-xcolor to use a <div>, page 1224.
>
> **HTML lateximage:** Remembers and reuses the print version \colorbox.

\fcolorbox:

> **Print:** Modified to allow a background of none.
>   \LWR@print@fcolorbox at section 89
>
> **HTML:** Redefined by lwarp-xcolor, page 1224.
>
> **HTML lateximage:** Remembers and reuses the print version.

\fcolorboxBlock:

> **Print:** Becomes \fcolorbox. Section 89
>
> **HTML:** Newly defined by lwarp-xcolor to use a <div>, page 1225.
>
> **HTML lateximage:** Remembers and reuses the print version \fcolorbox.

fcolorminipage:

> **Print:** Newly defined in the lwarp core.
>   LWR@print@fcolorminipage at section 89
>
> **HTML:** Newly defined by lwarp-xcolor, page 1225.
>
> **HTML lateximage:** Uses the print version.

\boxframe:

> **Print:** Used as-is.
>
> **HTML:** Redefined by lwarp-xcolor, page 1226.
>
> **HTML lateximage:** Remembers and reuses the print version.

§ 674.3 **Package loading**

1 `\LWR@ProvidesPackagePass{xcolor}[2016/05/11]`

\color@endgroup's \endgraf was conflicting with lwarp's paragraph handling.

2 `\let\color@endgroup\endgroup`

§ 674.4 **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 `}`

§ 674.5 `\normalcolor`

`\normalcolor`

9 `\DeclareRobustCommand{\LWR@HTML@normalcolor}{\color{black}}%`
10
11 `\LWR@formatted{normalcolor}`

§ 674.6 **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 `}`

Prints a color style for the current color.

`\LWR@currenttextcolorstyle` 19 `\newcommand*{\LWR@currenttextcolorstyle}{%`
20 `    \LWR@findcurrenttextcolor%`
21 `    \ifdefstring{\LWR@tempcolor}{000000}%`
22 `    {}%`
23 `    {color: \LWR@origpound\LWR@tempcolor ; }%`
24 `}`

`\LWR@textcurrentcolor` {⟨*text*⟩} Like \textcolor but uses the current \color instead.

25 `\DeclareDocumentCommand{\LWR@textcurrentcolor}{m}{%`
26 `    \begingroup%`
27 `    \LWR@hook@processingtags%`
28 `    \LWR@findcurrenttextcolor%`

```
29      \InlineClass[color:\LWR@origpound\LWR@tempcolor]{textcolor}{%
30          \renewcommand*{\LWR@currenttextcolor}{\LWR@origpound\LWR@tempcolor}%
31          #1%
32      }%
33      \endgroup%
34 }
```

\LWR@colorstyle   {⟨1: model⟩} {⟨2: color⟩}

For a color style, prints the color converted to HTML colors.

```
35 \NewDocumentCommand{\LWR@colorstyle}{m m}{%
36      \begingroup%
37      \LWR@hook@processingtags%
```

Use the xcolor package to convert to an HTML color space:

```
38      \convertcolorspec{#1}{#2}{HTML}\LWR@tempcolor%
```

Print the converted color:

```
39      \LWR@origpound\LWR@tempcolor%
40      \endgroup%
41 }
```

\LWR@backgroundcolor [⟨model⟩] {⟨color⟩} {⟨text⟩}

Similar to \textcolor, but prints black text against a color background.

Converted into an HTML hex color span.

```
42 \NewDocumentCommand{\LWR@backgroundcolor}{O{named} m m}{%
43      \begingroup%
44      \LWR@hook@processingtags%
45      \InlineClass[background:\LWR@colorstyle{#1}{#2}]{backgroundcolor}{%
46          #3%
47      }%
48      \endgroup%
49 }
```

§ 674.7   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.

```
50 \newcommand*{\LWR@borderpadding}[2]{%
51   border:\LWR@printlength{\LWR@atleastonept} solid \LWR@colorstyle{#1}{#2} ; %
52      padding:\LWR@printlength{\fboxsep}%
53 }
```

§ 674.8 **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.

```
54 \NewDocumentCommand{\LWR@HTML@color}{o m}{%
55     \IfValueTF{#1}{%
56         \LWR@print@color[#1]{#2}%
57         \convertcolorspec{#1}{#2}{HTML}\LWR@tempcolor%
58     }{%
59         \LWR@print@color{#2}%
60         \convertcolorspec{named}{#2}{HTML}\LWR@tempcolor%
61     }%
62     \edef\LWR@currenttextcolor{\LWR@origpound\LWR@tempcolor}%
63 }
64
65 \LWR@formatted{color}
```

\textcolor [⟨*model*⟩] {⟨*color*⟩} {⟨*text*⟩}

Converted into an HTML hex color span.

```
66 \NewDocumentCommand{\LWR@HTML@textcolor}{o m m}{%
67     \begingroup%
68     \LWR@hook@processingtags%
69     \IfValueTF{#1}{%
70         \color[#1]{#2}%
71     }{%
72         \color{#2}%
73     }%
74     \InlineClass[color:\LWR@currenttextcolor]{textcolor}{#3}%
75     \endgroup%
76 }%
77
78 \LWR@formatted{textcolor}
```

\pagecolor [⟨*model*⟩] {⟨*color*⟩}

Ignored. Use CSS instead.

```
79 \renewcommand*{\pagecolor}[2][named]{}
```

\nopagecolor Ignored.

```
80 \renewcommand*{\nopagecolor}{}
```

\colorbox [⟨*model*⟩] {⟨*color*⟩} {⟨*text*⟩}

Converted into an HTML hex background color <span>.

```
81 \NewDocumentCommand{\LWR@HTML@colorbox}{O{named} m +m}{%
82     \begingroup%
```

```
83      \LWR@hook@processingtags%
84      \InlineClass[%
85      background:\LWR@colorstyle{#1}{#2} ; %
86      padding:\LWR@printlength{\fboxsep}%
87      ]{colorbox}{#3}%
88      \endgroup%
89 }
```

\colorboxBlock [⟨*model*⟩] {⟨*color*⟩} {⟨*text*⟩}

Converted into an HTML hex background color <div>.

```
90 \NewDocumentCommand{\LWR@HTML@colorboxBlock}{O{named} m +m}{%
91      \begingroup%
92      \LWR@hook@processingtags%


93      \LWR@stoppars%


94      \begin{BlockClass}[%
95          background:\LWR@colorstyle{#1}{#2} ; %
96          padding:\LWR@printlength{\fboxsep}%
97      ]{colorboxBlock}
98      #3
99      \end{BlockClass}%
100     \endgroup%
```

Prevent paragraph tags around horizontal white space until the start of the next paragraph:

```
101     \global\booltrue{LWR@minipagethispar}%
102 }
```

\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.

```
103 \NewDocumentCommand{\LWR@HTML@fcolorbox}{O{named} m O{#1} m +m}{%
104     \LWR@traceinfo{HTML fcolorbox #2 #4}%
105     \begingroup%
106     \LWR@hook@processingtags%
107     \LWR@forceminwidth{\fboxrule}%
108     \ifthenelse{\equal{#4}{none}}%
109         {% no background color
110             \InlineClass[%
111             \LWR@borderpadding{#1}{#2}%
112             ]{fcolorbox}{#5}%
113         }%
114         {% yes background color
115             \InlineClass[%
116             \LWR@borderpadding{#1}{#2} ; %
117             background:\LWR@colorstyle{#3}{#4}%
118             ]{fcolorbox}{#5}%
119         }%
120     \endgroup%
121 }
```

\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.

```
122 \NewDocumentCommand{\LWR@HTML@fcolorboxBlock}{O{named} m O{#1} m +m d()}{%
123     \LWR@traceinfo{HTML fcolorboxBlock #2 #4}%
124     \begingroup%
125     \LWR@hook@processingtags%
126     \LWR@forceminwidth{\fboxrule}%


127     \LWR@stoppars%


128     \ifthenelse{\equal{#4}{none}}%
129         {% no background color
130             \begin{BlockClass}[%
131                 \LWR@borderpadding{#1}{#2}%
132                 \IfValueT{#6}{ ; #6}%
133             ]{fcolorboxBlock}
134             #5
135             \end{BlockClass}%
136         }%
137         {% yes background color
138             \convertcolorspec{#3}{#4}{HTML}\LWR@tempcolortwo%
139             \begin{BlockClass}[%
140                 background:\LWR@origpound\LWR@tempcolortwo\ ; %
141                 \LWR@borderpadding{#1}{#2}%
142                 \IfValueT{#6}{ ; #6}%
143             ]{fcolorboxBlock}
144             #5
145             \end{BlockClass}%
146         }%
147     \endgroup%
```

Prevent paragraph tags around horizontal white space until the start of the next paragraph:

```
148     \global\booltrue{LWR@minipagethispar}%
149     \LWR@traceinfo{HTML fcolorboxBlock done}%
150 }
```

Creates a framed HTML <div> around its contents.

A print-output version is defined in the lwarp core: section 89

\LWR@subfcolorminipage   {⟨*framemodel*⟩} {⟨*framecolor*⟩} {⟨*background tag*⟩} {⟨*height*⟩}

```
151 \NewDocumentCommand{\LWR@subfcolorminipage}{m m m m}{%


152     \LWR@stoppars%


153     \begin{BlockClass}[%
154         #3%
155         \LWR@borderpadding{#1}{#2} ; %
156         \IfValueT{#4}{height:\LWR@printlength{\LWR@tempheight} ; }%
157         width:\LWR@printlength{\LWR@tempwidth}%
```

```
158     ]{fcolorminipage}%
159 }
```

fcolorminipage (*env.*) [⟨*1:framemodel*⟩] {⟨*2:framecolor*⟩} [⟨*3:boxmodel*⟩] {⟨*4:boxcolor*⟩} [⟨*5:align*⟩]
[⟨*6:height*⟩] [⟨*7:inner-align*⟩] {⟨*8:width*⟩}

```
160 \NewDocumentEnvironment{LWR@HTML@fcolorminipage}{O{named} m O{#1} m O{c} o o m}
161 {%
162     \LWR@hook@processingtags%
163     \setlength{\LWR@tempwidth}{#8}%
164     \IfValueT{#6}{\setlength{\LWR@tempheight}{#6}}%
165     \LWR@forceminwidth{\fboxrule}%
166     \convertcolorspec{#1}{#2}{HTML}\LWR@tempcolor%
167     \ifthenelse{\equal{#4}{none}}%
168         {\LWR@subfcolorminipage{#1}{#2}{}{#6}}%
169         {%
170             \convertcolorspec{#3}{#4}{HTML}\LWR@tempcolortwo%
171             \LWR@subfcolorminipage{#1}{#2}%
172                 {background:\LWR@origpound\LWR@tempcolortwo\ ; }%
173                 {#6}%
174         }%
175 }%
176 {%
177     \end{BlockClass}%
```

Prevent paragraph tags around horizontal white space until the start of the next
paragraph:

```
178     \global\booltrue{LWR@minipagethispar}%
179 }
```

\boxframe {⟨*width*⟩} {⟨*height*⟩} {⟨*depth*⟩}

The depth is added to the height, but the box is not decended below by the depth.
\textcolor is honored.

```
180 \newcommand*{\LWR@HTML@boxframe}[3]{%
181     {%
182         \setlength{\LWR@tempwidth}{#1}%
183         \setlength{\LWR@tempheight}{#2}%
184         \addtolength{\LWR@tempheight}{#3}%
185         \LWR@forceminwidth{\fboxrule}%
186         \LWR@findcurrenttextcolor%
187         \InlineClass[%
188             display:inline-block ; %
189             border:%
190                 \LWR@printlength{\LWR@atleastonept} % space
191                 solid % space
192                 \LWR@currenttextcolor{} ; % space
193             width:\LWR@printlength{\LWR@tempwidth} ; %
194             height:\LWR@printlength{\LWR@tempheight}%
195         ]{boxframe}{}%
196     }%
197 }
198
199 \LWR@formatted{boxframe}
```

§ 674.9 **Row colors**

\rowc@l@rs   [⟨*cmds*⟩] {⟨*startrow*⟩} {⟨*odd color*⟩} {⟨*even color*⟩}

```
200 \newcommand*{\LWR@xcolortempcolor}{}
201
202 \def\rowc@l@rs[#1]#2#3#4%
203 {%
204     \rownum=1%
205     \@rowcolorstrue%
206     \@ifxempty{#3}%
207         {\def\@oddrowcolor{\@norowcolor}}%
208         {%
209             \convertcolorspec{named}{#3}{HTML}\LWR@xcolortempcolor%
210             \edef\@oddrowcolor{%
211                 \csdef{LWR@xcolorrowHTMLcolor}{\LWR@xcolortempcolor}%
212             }%
213         }%
214     \@ifxempty{#4}%
215         {\def\@evenrowcolor{\@norowcolor}}%
216         {%
217             \convertcolorspec{named}{#4}{HTML}\LWR@xcolortempcolor%
218             \edef\@evenrowcolor{%
219                 \csdef{LWR@xcolorrowHTMLcolor}{\LWR@xcolortempcolor}%
220             }%
221         }%
222     \if@rowcmd
223         \def\@rowcolors
224         {%
225 %           #1%
226             \if@rowcolors
227 %             \noalign{%
228                 \relax\ifnum\rownum<#2\@norowcolor\else
229                 \ifodd\rownum\@oddrowcolor\else\@evenrowcolor\fi\fi%
230 %             }%
231             \fi%
232         }%
233     \else
234         \def\@rowcolors
235         {%
236             \if@rowcolors
237                 \ifnum\rownum<#2%
238 %                 \noalign{%
239                     \@norowcolor
240 %                 }
241                 \else
242 %                 #1%
243 %                 \noalign{%
244                     \ifodd\rownum\@oddrowcolor\else\@evenrowcolor\fi%
245 %                 }%
246                 \fi
247             \fi%
248         }%
249     \fi
250     \ignorespaces%
251 }
```

\@norowcolor   Turns off color for this row.

```
252 \def\@norowcolor{%
253     \renewcommand{\LWR@xcolorrowHTMLcolor}{}%
```

254 }

\@rowc@lors   Executed at the end of each row.

255 \def\@rowc@lors{%
256 %   \noalign{%
257   \advance\rownum\@ne%
258 %   }%
259   \@rowcolors%
260 }

File 566   **`lwarp-xechangebar.sty`**

§ 675   Package   # xechangebar

xechangebar (*Pkg*)   xechangebar is ignored

for HTML output: 1 \LWR@ProvidesPackageDrop{xechangebar}[2017/08/03]
2 \LWR@origRequirePackage{lwarp-changebar}

File 567   **`lwarp-xellipsis.sty`**

§ 676   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 568   **`lwarp-xetexko.sty`**

§ 677   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 \protected\def\typesetvertical{}
5 \protected\def\typesethorizontal{}
6
7 \def\verticaltypesetting{\BlockClass{verticalrl}}
8 \def\beginverticaltypesetting{\BlockClass{verticalrl}}
9 \def\endverticaltypesetting{\endBlockClass}
10
11 \protected\def\vertical#1{\BlockClass{verticalrl}}
12 \protected\def\endvertical{\endBlockClass}
13 \protected\def\horizontal#1{\BlockClass{horizontaltb}}
14 \protected\def\endhorizontal{\endBlockClass}
15 \DeclareDocumentCommand{\vertlatin}{m}{#1}
```

File 569   **`lwarp-xevlna.sty`**

§ 678   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 570   `lwarp-xfakebold.sty`

§ 679   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 571   `lwarp-xfrac.sty`

§ 680   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 572 **`lwarp-xltabular.sty`**

§ 681   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 573   **`lwarp-xltxtra.sty`**

§ 682   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 574   **`lwarp-xmpincl.sty`**

§ 683   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 575   **`lwarp-xpiano.sty`**

§ 684   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 576   **`lwarp-xpinyin.sty`**

§ 685   Package   **xpinyin**

(*Emulates or patches code by* SOBEN LEE.)

xpinyin (*Pkg*)   xpinyin is supported.

Pinyin is disabled for file names, the sideTOC, 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 577   **lwarp-xr.sty**

§ 686   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. \externaldocument is modified to also accept the optional arguments for xr-hyper, which currently uses xr for HTML output.

See section 5.17.

for HTML output:
```
1 \LWR@ProvidesPackagePass{xr}[2019/07/22]%
```
```
2 \LetLtxMacro\LWR@orig@externaldocument\externaldocument
3
4 \RenewDocumentCommand{\externaldocument}{O{} O{} m O{}}{%
5    \ifblank{#1}{%
6        \LWR@orig@externaldocument{#3_html}%
7    }{%
8        \LWR@orig@externaldocument[#1]{#3_html}%
9    }%
10 }
```

File 578   **lwarp-xr-hyper.sty**

§ 687   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 5.17.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{xr-hyper}[2019/10/03]%
2
3 \LWR@origRequirePackage{lwarp-xr}
```

File 579   **lwarp-xtab.sty**

§ 688   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 580 **lwarp-xunicode.sty**

§ 689   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 581   **`lwarp-xurl.sty`**

§ 690   Package   # xurl

xurl (*Pkg*)   xurl is ignored.

for HTML output: 1 `\LWR@ProvidesPackageDrop{xurl}[2020/01/14]`
2
3 `\def\useOriginalUrlSetting{}`

File 582   **`lwarp-xy.sty`**

§ 691   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 583 **lwarp-zhlineskip.sty**

§ 692   Package   # zhlineskip

zhlineskip (*Pkg*)   zhlineskip is ignored.

for HTML output: 1 \LWR@ProvidesPackageDrop{zhlineskip}[2019/05/15]

```
2 \newcommand*\SetTextEnvironmentSinglespace[1]{}
3 \newcommand*\RestoreTextEnvironmentLeading[1]{}
4 \newcommand*\SetMathEnvironmentSinglespace[1]{}
5 \newcommand*\RestoreMathEnvironmentLeading[1]{}
```

File 584   **lwarp-zwpagelayout.sty**

§ 693   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 585   **`lwarp-patch-komascript.sty`**

§ 694   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 586   **`lwarp-patch-memoir.sty`**

§ 695   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 695.1). 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}

§ 695.1   **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
```

§ 695.2 **Label handling**

Insert the lwarp label mechanism into the memoir package mechanism:

- \@mem@old@label is the LaTeX definition of \label.

- \LWR@orig@label becomes the memoir definition.

- lwarp's \LWR@new@label uses \LWR@orig@label.

- Want memoir's \label to use lwarp's \label, which then would use LaTeX's \label.

- So:

  – \@mem@old@label is set to \LWR@new@label.

  – \LWR@orig@label is set to \@mem@old@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.

```
36 \LetLtxMacro\LWR@orig@label\@mem@old@label
37 \LetLtxMacro\@mem@old@label\LWR@new@label
```

Patches for subfloats to support additional lwarp labels. This is the non-hyperref version from memoir.

```
38 \AtBeginDocument{
39     \renewcommand*{\sf@@memsub@label}[1]{%
40         \@bsphack
41         \sf@@memsub@label@hook{#1}%
42 %        \@memoldlabel{#1}%
43         \cref@label{#1}%                        lwarp
44         \LWR@label@createtag{sub@#1}%            lwarp
45         \protected@write\@auxout{}{%
46             \string\newlabel{sub@#1}%
47             {{\@nameuse{@@thesub\@captype}}%
48             {\thepage}}}%
49         \LWR@write@lwarplabel{sub@#1}%          lwarp
50         \@esphack
51     }
52 }
```

§ 695.3 **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.

```
53 \setstocksize{190in}{20in}
54 \setlrmarginsandblock{2in}{2in}{*}
55 \setulmarginsandblock{1in}{1in}{*}

56 \renewcommand*{\stockavi}{}
57 \renewcommand*{\stockav}{}
58 \renewcommand*{\stockaiv}{}
59 \renewcommand*{\stockaiii}{}
```

```
60 \renewcommand*{\stockavii}{}
61 \renewcommand*{\stockbvi}{}
62 \renewcommand*{\stockbv}{}
63 \renewcommand*{\stockbiv}{}
64 \renewcommand*{\stockbiii}{}
65 \renewcommand*{\stockbvii}{}
66 % \renewcommand*{\stockmetriccrownvo}{}% in docs but not in the package
67 \renewcommand*{\stockmlargecrownvo}{}
68 \renewcommand*{\stockmdemyvo}{}
69 \renewcommand*{\stockmsmallroyalvo}{}
70 \renewcommand*{\pageavi}{}
71 \renewcommand*{\pageavii}{}
72 \renewcommand*{\pageav}{}
73 \renewcommand*{\pageaiv}{}
74 \renewcommand*{\pageaiii}{}
75 \renewcommand*{\pagebvi}{}
76 \renewcommand*{\pagebvii}{}
77 \renewcommand*{\pagebv}{}
78 \renewcommand*{\pagebiv}{}
79 \renewcommand*{\pagebiii}{}
80 % \renewcommand*{\pagemetriccrownvo}{}% in docs but not in the package
81 \renewcommand*{\pagemlargecrownvo}{}
82 \renewcommand*{\pagemdemyvo}{}
83 \renewcommand*{\pagemsmallroyalvo}{}
84
85 \renewcommand*{\stockdbill}{}
86 \renewcommand*{\stockstatement}{}
87 \renewcommand*{\stockexecutive}{}
88 \renewcommand*{\stockletter}{}
89 \renewcommand*{\stockold}{}
90 \renewcommand*{\stocklegal}{}
91 \renewcommand*{\stockledger}{}
92 \renewcommand*{\stockbroadsheet}{}
93 \renewcommand*{\pagedbill}{}
94 \renewcommand*{\pagestatement}{}
95 \renewcommand*{\pageexecutive}{}
96 \renewcommand*{\pageletter}{}
97 \renewcommand*{\pageold}{}
98 \renewcommand*{\pagelegal}{}
99 \renewcommand*{\pageledger}{}
100 \renewcommand*{\pagebroadsheet}{}
101
102 \renewcommand*{\stockpottvo}{}
103 \renewcommand*{\stockfoolscapvo}{}
104 \renewcommand*{\stockcrownvo}{}
105 \renewcommand*{\stockpostvo}{}
106 \renewcommand*{\stocklargecrownvo}{}
107 \renewcommand*{\stocklargepostvo}{}
108 \renewcommand*{\stocksmalldemyvo}{}
109 \renewcommand*{\stockdemyvo}{}
110 \renewcommand*{\stockmediumvo}{}
111 \renewcommand*{\stocksmallroyalvo}{}
112 \renewcommand*{\stockroyalvo}{}
113 \renewcommand*{\stocksuperroyalvo}{}
114 \renewcommand*{\stockimperialvo}{}
115 \renewcommand*{\pagepottvo}{}
116 \renewcommand*{\pagefoolscapvo}{}
117 \renewcommand*{\pagecrownvo}{}
118 \renewcommand*{\pagepostvo}{}
119 \renewcommand*{\pagelargecrownvo}{}
```

```
120 \renewcommand*{\pagelargepostvo}{}
121 \renewcommand*{\pagesmalldemyvo}{}
122 \renewcommand*{\pagedemyvo}{}
123 \renewcommand*{\pagemediumvo}{}
124 \renewcommand*{\pagesmallroyalvo}{}
125 \renewcommand*{\pageroyalvo}{}
126 \renewcommand*{\pagesuperroyalvo}{}
127 \renewcommand*{\pageimperialvo}{}
128
129 \renewcommand*{\memfontfamily}{}
130 \renewcommand*{\memfontenc}{}
131 \renewcommand*{\memfontpack}{}
132
133 \renewcommand*{\anyptfilebase}{}
134 \renewcommand*{\anyptsize}{10}
135
136 \renewcommand*{\setstocksize}[2]{}
137 \renewcommand*{\settrimmedsize}[3]{}
138 \renewcommand*{\settrims}[2]{}
139
140 % \newlength{\lxvchars}
141 % \setlength{\lxvchars}{305pt}
142 % \newlength{\xlvchars}
143 % \setlength{\xlvchars}{190pt}
144 \renewcommand*{\setxlvchars}[1]{}
145 \renewcommand*{\setlxvchars}[1]{}
146
147 \renewcommand*{\settypeblocksize}[3]{}
148 \renewcommand*{\setlrmargins}[3]{}
149 \renewcommand*{\setlrmarginsandblock}[3]{}
150 \renewcommand*{\setbinding}[1]{}
151 \renewcommand*{\setulmargins}[3]{}
152 \renewcommand*{\setulmarginsandblock}[3]{}
153 \renewcommand*{\setcolsepandrule}[2]{}
154
155 \renewcommand*{\setheadfoot}[2]{}
156 \renewcommand*{\setheaderspaces}[3]{}
157 \renewcommand*{\setmarginnotes}[3]{}
158 \renewcommand*{\setfootins}[2]{}
159 \renewcommand*{\checkandfixthelayout}[1][]{}
160 \renewcommand*{\checkthelayout}[1]{}
161 \renewcommand*{\fixthelayout}{}
162 %
163 % \newlength{\stockheight}
164 % \newlength{\trimtop}
165 % \newlength{\trimedge}
166 % \newlength{\stockwidth}
167 % \newlength{\spinemargin}
168 % \newlength{\foremargin}
169 % \newlength{\uppermargin}
170 % \newlength{\headmargin}
171 %
172 \renewcommand*{\typeoutlayout}{}
173 \renewcommand*{\typeoutstandardlayout}{}
174 \renewcommand*{\settypeoutlayoutunit}[1]{}
175 \renewcommand*{\fixpdflayout}{}
176 \renewcommand*{\fixdvipslayout}{}
177
178 \renewcommand*{\medievalpage}[1][]{}
179 \renewcommand*{\isopage}[1][]{}
```

```
180 \renewcommand*{\semiisopage}[1][]{}
181
182 \renewcommand{\setpagebl}[3]{}
183 \renewcommand{\setpageml}[3]{}
184 \renewcommand{\setpagetl}[3]{}
185 \renewcommand{\setpagetm}[3]{}
186 \renewcommand{\setpagetr}[3]{}
187 \renewcommand{\setpagemr}[3]{}
188 \renewcommand{\setpagebr}[3]{}
189 \renewcommand{\setpagebm}[3]{}
190 \renewcommand{\setpagecc}[3]{}
```

§ 695.4 Text and fonts

```
191 \let\miniscule\tiny
192 \let\HUGE\Huge
193
194 \renewcommand*{\abnormalparskip}[1]{}
195 \renewcommand*{\nonzeroparskip}{}
196 \renewcommand*{\traditionalparskip}{}
197
198 \let\onelineskip\baselineskip
199
200 \let\OnehalfSpacing\onehalfspacing
201 \let\DoubleSpacing\doublespacing
202 \renewcommand*{\setPagenoteSpacing}[1]{}
203 \renewcommand*{\setFloatSpacing}[1]{}

204 \renewcommand{\SingleSpacing}{\@ifstar\singlespacing\singlespacing}

205 \let\setSingleSpace\SetSinglespace
206 \let\SingleSpace\singlespace
207 \let\endSingleSpace\endsinglespace
208 \let\Spacing\spacing
209 \let\endSpacing\endspacing
210 \let\OnehalfSpace\onehalfspace
211 \let\endOnehalfSpace\endonehalfspace
212 \csletcs{OnehalfSpace*}{onehalfspace}
213 \csletcs{endOnehalfSpace*}{endonehalfspace}
214 \let\DoubleSpace\doublespace
215 \let\endDoubleSpace\enddoublespace
216 \csletcs{DoubleSpace*}{doublespace}
217 \csletcs{endDoubleSpace*}{enddoublespace}
218 \renewcommand*{\setDisplayskipStretch}[1]{}
219 \renewcommand*{\memdskipstretch}{}
220 \renewcommand*{\noDisplayskipStretch}{}
221 \renewcommand*{\memdskips}{}
222
223 \renewcommand*{\midsloppy}{}
224 \renewenvironment*{midsloppypar}{}{}
225
226 \renewcommand*{\sloppybottom}{}
```

§ 695.5 Titles

```
227 \csletcs{titlingpage*}{titlingpage}
228 \csletcs{endtitlingpage*}{endtitlingpage}
229 \let\titlingpageend\relax
230 \newcommand{\titlingpageend}[2]{}
231 \let\andnext\and
```

```
232 \renewcommand*{\thanksmarkstyle}[1]{}
233
234 \renewcommand{\thanksfootmark}{%
235     \thanksscript{\tamark}%
236 }
237
238 % \newlength{\thanksmarksep}% already provided by memoir

239 \renewcommand\titlingpageend[2]{}
```

§ 695.6 **Abstracts**

```
240 % \newlength{\absindent}
241 % \newlength{\absparsep}
242 \renewcommand*{\abstractcol}{}
243 \renewcommand*{\abstractintoc}{}
244 \renewcommand*{\abstractnum}{}
245 \renewcommand*{\abstractrunin}{}
```

§ 695.7 **Docment divisions**

\book   * (⟨*2:PDF name*⟩) [⟨*3:TOC name*⟩] [⟨*4:PDF name*⟩] (⟨*5:PDF name*⟩) {⟨*6:name*⟩}

```
246 \DeclareDocumentCommand{\book}{s d() o o d() m}{%
247     \LWR@section{#1}{#3}{#6}{book}%
248 }

249 \def\@apppage{%
250     \part*{\appendixpagename}
251 }
252 \renewcommand\mempreaddapppagetotochook{}
253 \renewcommand\mempostaddapppagetotochook{}
254
255 \def\@sapppage{%
256     \part*{\appendixpagename}
257 }

258 \DeclareDocumentCommand{\mainmatter}{s}{%
259     \booltrue{LWR@mainmatter}%
260 }
261
262 \DeclareDocumentCommand{\frontmatter}{s}{%
263     \boolfalse{LWR@mainmatter}%
264 }

265 \renewcommand*{\raggedbottomsection}{}
266 \renewcommand*{\normalbottomsection}{}
267 \renewcommand*{\bottomsectionskip}{}
268 \renewcommand*{\bottomsectionpenalty}{}
269 \csletcs{appendixpage*}{appendixpage}
270 \renewcommand*{\namedsubappendices}{}
271 \renewcommand*{\unnamedsubappendices}{}
272 \renewcommand*{\beforebookskip}{}
273 \renewcommand*{\afterbookskip}{}
274 \renewcommand*{\beforepartskip}{}
275 \renewcommand*{\afterpartskip}{}
276 \renewcommand*{\midbookskip}{}
```

```
277 \renewcommand*{\midpartskip}{}
278 \renewcommand*{\printbookname}{}
279 \renewcommand*{\booknamefont}{}
280 \renewcommand*{\booknamenum}{}
281 \renewcommand*{\printbooknum}{}
282 \renewcommand*{\booknumfont}{}
283 \renewcommand*{\printpartname}{}
284 \renewcommand*{\partnamefont}{}
285 \renewcommand*{\partnamenum}{}
286 \renewcommand*{\printpartnum}{}
287 \renewcommand*{\partnumfont}{}
288 \renewcommand*{\printbooktitle}[1]{}
289 \renewcommand*{\booktitlefont}{}
290 \renewcommand{\printparttitle}[1]{}
291 \renewcommand*{\parttitlefont}{}
292 \renewcommand*{\bookpageend}{}
293 \renewcommand*{\bookblankpage}{}
294 \renewcommand*{\nobookblankpage}{}
295 \renewcommand*{\partpageend}{}
296 \renewcommand*{\partblankpage}{}
297 \renewcommand*{\nopartblankpage}{}
298 \RenewDocumentCommand{\newleadpage}{s o m m}{}% todo
299 \RenewDocumentCommand{\renewleadpage}{s o m m}{}% todo
300 \renewcommand*{\leadpagetoclevel}{chapter}
301
302 \renewcommand*{\openright}{}
303 \renewcommand*{\openleft}{}
304 \renewcommand*{\openany}{}
305 \renewcommand*{\clearforchapter}{}
306 \renewcommand*{\memendofchapterhook}{}
307 \renewcommand*{\chapterheadstart}{}
308 % \newlength{\beforechapskip}
309 \renewcommand*{\afterchapternum}{}
310 % \newlength{\midchapskip}
311 \renewcommand*{\afterchaptertitle}{}
312 % \newlength{\afterchapskip}
313 \renewcommand*{\printchaptername}{}
314 \renewcommand*{\chapnamefont}{}
315 \renewcommand*{\chapternamenum}{}
316 \renewcommand*{\printchapternum}{}
317 \renewcommand*{\chapnumfont}{}
318 \renewcommand{\printchaptertitle}[1]{}
319 \renewcommand*{\chaptitlefont}{}
320 \renewcommand*{\printchapternonum}{}
321 \renewcommand*{\indentafterchapter}{}
322 \renewcommand*{\noindentafterchapter}{}
323 \renewcommand*{\insertchapterspace}{}
324
325 \renewcommand*{\chapterstyle}[1]{}
326 \renewcommand{\makechapterstyle}[2]{}
327 \renewcommand*{\chapindent}{}
328 \let\chapterprecis\cftchapterprecis
329 \let\chapterprecishere\cftchapterprecishere
330 \let\chapterprecistoc\cftchapterprecistoc
331 \renewcommand*{\precisfont}{}
332 \renewcommand*{\prechapterprecis}{}
333 \renewcommand*{\postchapterprecis}{}
334 \renewcommand{\precistoctext}[1]{}
335 \renewcommand*{\precistocfont}{}
336 \renewcommand*{\precistocformat}{}
```

```
337 % \newlength{\prechapterprecisshift}
338
339 \renewcommand*{\setbeforesecskip}[1]{}
340 \renewcommand*{\setaftersecskip}[1]{}
341 \renewcommand*{\setsecindent}[1]{}
342 \renewcommand*{\setsecheadstyle}[1]{}
343 \renewcommand*{\setbeforesubsecskip}[1]{}
344 \renewcommand*{\setaftersubsecskip}[1]{}
345 \renewcommand*{\setsubsecindent}[1]{}
346 \renewcommand*{\setsubsecheadstyle}[1]{}
347 \renewcommand*{\setbeforesubsubsecskip}[1]{}
348 \renewcommand*{\setaftersubsubsecskip}[1]{}
349 \renewcommand*{\setsubsubsecindent}[1]{}
350 \renewcommand*{\setsubsubsecheadstyle}[1]{}
351 \renewcommand*{\setbeforeparaskip}[1]{}
352 \renewcommand*{\setafterparaskip}[1]{}
353 \renewcommand*{\setparaindent}[1]{}
354 \renewcommand*{\setparaheadstyle}[1]{}
355 \renewcommand*{\setbeforesubparaskip}[1]{}
356 \renewcommand*{\setaftersubparaskip}[1]{}
357 \renewcommand*{\setsubparaindent}[1]{}
358 \renewcommand*{\setsubparaheadstyle}[1]{}
359 \renewcommand{\@hangfrom}[1]{#1}
360 \renewcommand{\sethangfrom}[1]{}
361 \renewcommand{\setsecnumformat}[1]{}
362
363 \renewcommand*{\hangsecnum}{}
364 \renewcommand*{\defaultsecnum}{}
365
366 \renewcommand*{\sechook}{}
367 \renewcommand{\setsechook}[1]{}
368 \renewcommand*{\subsechook}{}
369 \renewcommand{\setsubsechook}[1]{}
370 \renewcommand*{\subsubsechook}{}
371 \renewcommand{\setsubsubsechook}[1]{}
372 \renewcommand*{\parahook}{}
373 \renewcommand{\setparahook}[1]{}
374 \renewcommand*{\subparahook}{}
375 \renewcommand{\setsubparahook}[1]{}
376
377 \RenewDocumentCommand{\plainbreak}{s m}{\begin{center}~\end{center}}
378
379 \RenewDocumentCommand{\fancybreak}{s +m}{%
380     \begin{center}#2\end{center}%
381 }
382
383 \RenewDocumentCommand{\plainfancybreak}{s m m +m}{%
384     \begin{center}#4\end{center}%
385 }
386
387 \RenewDocumentCommand{\pfbreak}{s}{%
388     \begin{center}
389     \pfbreakdisplay
390     \end{center}
391 }
392
393 % \newlength{\pfbreakskip}
394 \renewcommand{\pfbreakdisplay}{*\quad*\quad*}
395
396 \renewcommand{\makeheadstyles}[2]{}
```

```
397 \renewcommand*{\headstyles}[1]{}
```

§695.8 **Pagination and headers**

```
398 \renewcommand*{\savepagenumber}{}
399 \renewcommand*{\restorepagenumber}{}
400 \renewcommand*{\uppercaseheads}{}
401 \renewcommand*{\nouppercaseheads}{}
402
403 \renewcommand*{\bookpagemark}[1]{}
404 \renewcommand*{\partmark}[1]{}
405 \renewcommand*{\bibmark}{}
406 \renewcommand*{\indexmark}{}
407 \renewcommand*{\glossarymark}{}
408
409 \LWR@origpagestyle{empty}
410 \renewcommand*{\ps@empty}{}
411 \renewcommand*{\makepagestyle}[1]{}
412 \renewcommand*{\emptypshook}{}%
413 % \renewcommand*{\empty@oddhead}{}
414 % \renewcommand*{\empty@oddfoot}{}
415 % \renewcommand*{\empty@evenhead}{}
416 % \renewcommand*{\empty@evenfoot}{}
417 \renewcommand*{\@oddhead}{}
418 \renewcommand*{\@oddfoot}{}
419 \renewcommand*{\@evenhead}{}
420 \renewcommand*{\@evenfoot}{}
421 \renewcommand*{\aliaspagestyle}[2]{}
422 \renewcommand*{\copypagestyle}[2]{}
423
424 \renewcommand*{\makeevenhead}[4]{}
425 \renewcommand*{\makeoddhead}[4]{}
426 \renewcommand*{\makeevenfoot}[4]{}
427 \renewcommand*{\makeoddfoot}[4]{}
428 \renewcommand*{\makerunningwidth}[3]{}
429 % \newlength{\headwidth}
430 \renewcommand*{\makeheadrule}[3]{}
431 \renewcommand*{\makefootrule}[3]{}
432 \renewcommand*{\makeheadfootruleprefix}[3]{}
433 % \newlength{\normalrulethickness}
434 % \setlength{\normalrulethickness}{.4pt}
435 % \newlength{\footruleheight}
436 % \newlength{\footruleskip}
437 \renewcommand*{\makeheadposition}[5]{}
438 \renewcommand{\makepsmarks}[2]{}
439 \renewcommand*{\makeheadfootstrut}[3]{}

440 \renewcommand{\createmark}[5]{\csdef{#1mark}[1]{}}
441 \renewcommand{\createplainmark}[3]{\csdef{#1mark}{}}

442 \renewcommand{\memUChead}[1]{}
443 \renewcommand*{\clearplainmark}[1]{}
444 \renewcommand*{\clearmark}[1]{}
445 \renewcommand{\addtopsmarks}[3]{}
446 \renewcommand{\ifonlyfloats}[2]{#2}
447 \renewcommand*{\mergepagefloatstyle}[3]{}
448
449 \renewcommand*{\framepichead}{}
450 \renewcommand*{\framepictextfoot}{}
451 \renewcommand*{\framepichook}{}
```

452 `\renewcommand*{\showheadfootlocoff}{}`
453 `\renewcommand*{\showtextblocklocoff}{}`

§695.9 Paragraphs and lists

454 `\renewcommand{\hangfrom}[1]{#1}`
455 `\let\centerfloat\centering`
456 `\renewcommand*{\raggedyright}[1][]{}`
457 `% \newlength{\ragrparindent}`
458 `\renewcommand{\sourceatright}[2][]{\attribution{#2}}`
459 `\let\memorigdbs\LWR@endofline`

460 `\renewcommand*{\memorigpar}{\par}`

461 `\let\atcentercr\LWR@endofline`
462
463 `\renewcommand*{\linenottooshort}[1][]{}`
464 `\renewcommand*{\russianpar}{}`
465 `\renewcommand*{\lastlinerulefill}{}`
466 `\renewcommand*{\lastlineparrule}{}`
467 `\renewcommand*{\justlastraggedleft}{}`
468 `\renewcommand*{\raggedrightthenleft}{}`
469 `\renewcommand*{\leftcenterright}{}`
470
471 `\renewcommand{\leftspringright}[4]{%`
472 `    \begin{minipage}{#1\linewidth}#3\end{minipage}\qquad%`
473 `  \begin{minipage}{#2\linewidth}\begin{flushright}#4\end{flushright}\end{minipage}%`
474 `}`
475
476 `\renewenvironment*{blockdescription}`
477 `{\LWR@descriptionstart\LWR@origdescription}`
478 `{\enddescription}`
479
480 `\renewcommand*{\blockdescriptionlabel}[1]{\textbf{#1}}`
481 `\renewenvironment*{labelled}[1]{\begin{description}}{\end{description}}`
482 `\renewenvironment*{flexlabelled}[6]{\begin{description}}{\end{description}}`
483 `\renewcommand*{\tightlists}{}`
484 `\renewcommand*{\defaultlists}{}`
485 `\RenewDocumentCommand{\firmlists}{s}{}`
486 `\renewcommand*{\firmlist}{}`
487 `\renewcommand*{\tightlist}{}`
488 `\renewcommand*{\zerotrivseps}{}`
489 `\renewcommand*{\savetrivseps}{}`
490 `\renewcommand*{\restoretrivseps}{}`

§695.10 Contents lists

491 `\csletcs{tableofcontents*}{tableofcontents}`
492 `\csletcs{listoffigures*}{listoffigures}`
493 `\csletcs{listoftables*}{listoftables}`
494 `\renewenvironment{KeepFromToc}{}{}`
495 `\renewcommand*{\onecoltocetc}{}`
496 `\renewcommand*{\twocoltocetc}{}`
497 `\renewcommand*{\ensureonecol}{}`
498 `\renewcommand*{\restorefromonecol}{}`
499 `\renewcommand*{\doccoltocetc}{}`
500
501 `\renewcommand{\tocheadstart}{}`
502 `\renewcommand{\printtoctitle}[1]{}`
503 `\renewcommand{\tocmark}{}`

```
504 \renewcommand{\aftertoctitle}{}
505 \renewcommand{\lofheadstart}{}
506 \renewcommand{\printloftitle}[1]{}
507 \renewcommand{\lofmark}{}
508 \renewcommand{\afterloftitle}{}
509 \renewcommand{\lotheadstart}{}
510 \renewcommand{\printlottitle}[1]{}
511 \renewcommand{\lotmark}{}
512 \renewcommand{\afterlottitle}{}
513
514 \renewcommand*{\setpnumwidth}[1]{}
515 \renewcommand*{\setrmarg}[1]{}
516 \renewcommand*{\cftbookbreak}{}
517 \renewcommand*{\cftpartbreak}{}
518 \renewcommand*{\cftchapterbreak}{}

519 % \newlength{\cftbeforebookskip}
520 % \newlength{\cftbookindent}
521 % \newlength{\cftbooknumwidth}
522 \renewcommand*{\cftbookfont}{}
523 \renewcommand*{\cftbookname}{}
524 \renewcommand*{\cftbookpresnum}{}
525 \renewcommand*{\cftbookaftersnum}{}
526 \renewcommand*{\cftbookaftersnumb}{}
527 \renewcommand*{\cftbookleader}{}
528 \renewcommand*{\cftbookdotsep}{1}
529 \renewcommand*{\cftbookpagefont}{}
530 \renewcommand*{\cftbookafterpnum}{}
531 \renewcommand*{\cftbookformatpnum}[1]{}
532 \renewcommand*{\cftbookformatpnumhook}[1]{}
```

Part is already defined by tocloft.

```
533 % \newlength{\cftbeforechapterskip}
534 % \newlength{\cftchapterindent}
535 % \newlength{\cftchapternumwidth}
536 \renewcommand*{\cftchapterfont}{}
537 \renewcommand*{\cftchaptername}{}
538 \renewcommand*{\cftchapterpresnum}{}
539 \renewcommand*{\cftchapteraftersnum}{}
540 \renewcommand*{\cftchapteraftersnumb}{}
541 \renewcommand*{\cftchapterleader}{}
542 \renewcommand*{\cftchapterdotsep}{1}
543 \renewcommand*{\cftchapterpagefont}{}
544 \renewcommand*{\cftchapterafterpnum}{}
545 \renewcommand*{\cftchapterformatpnum}[1]{}
546 \renewcommand*{\cftchapterformatpnumhook}[1]{}

547 % \newlength{\cftbeforesectionskip}
548 % \newlength{\cftsectionindent}
549 % \newlength{\cftsectionnumwidth}
550 \renewcommand*{\cftsectionfont}{}
551 \renewcommand*{\cftsectionname}{}
552 \renewcommand*{\cftsectionpresnum}{}
553 \renewcommand*{\cftsectionaftersnum}{}
554 \renewcommand*{\cftsectionaftersnumb}{}
555 \renewcommand*{\cftsectionleader}{}
556 \renewcommand*{\cftsectiondotsep}{1}
557 \renewcommand*{\cftsectionpagefont}{}
558 \renewcommand*{\cftsectionafterpnum}{}
```

```
559 \renewcommand*{\cftsectionformatpnum}[1]{}
560 \renewcommand*{\cftsectionformatpnumhook}[1]{}

561 % \newlength{\cftbeforesubsectionskip}
562 % \newlength{\cftsubsectionindent}
563 % \newlength{\cftsubsectionnumwidth}
564 \renewcommand*{\cftsubsectionfont}{}
565 \renewcommand*{\cftsubsectionname}{}
566 \renewcommand*{\cftsubsectionpresnum}{}
567 \renewcommand*{\cftsubsectionaftersnum}{}
568 \renewcommand*{\cftsubsectionaftersnumb}{}
569 \renewcommand*{\cftsubsectionleader}{}
570 \renewcommand*{\cftsubsectiondotsep}{1}
571 \renewcommand*{\cftsubsectionpagefont}{}
572 \renewcommand*{\cftsubsectionafterpnum}{}
573 \renewcommand*{\cftsubsectionformatpnum}[1]{}
574 \renewcommand*{\cftsubsectionformatpnumhook}[1]{}

575 % \newlength{\cftbeforesubsubsectionskip}
576 % \newlength{\cftsubsubsectionindent}
577 % \newlength{\cftsubsubsectionnumwidth}
578 \renewcommand*{\cftsubsubsectionfont}{}
579 \renewcommand*{\cftsubsubsectionname}{}
580 \renewcommand*{\cftsubsubsectionpresnum}{}
581 \renewcommand*{\cftsubsubsectionaftersnum}{}
582 \renewcommand*{\cftsubsubsectionaftersnumb}{}
583 \renewcommand*{\cftsubsubsectionleader}{}
584 \renewcommand*{\cftsubsubsectiondotsep}{1}
585 \renewcommand*{\cftsubsubsectionpagefont}{}
586 \renewcommand*{\cftsubsubsectionafterpnum}{}
587 \renewcommand*{\cftsubsubsectionformatpnum}[1]{}
588 \renewcommand*{\cftsubsubsectionformatpnumhook}[1]{}

589 % \newlength{\cftbeforeparagraphskip}
590 % \newlength{\cftparagraphindent}
591 % \newlength{\cftparagraphnumwidth}
592 \renewcommand*{\cftparagraphfont}{}
593 \renewcommand*{\cftparagraphname}{}
594 \renewcommand*{\cftparagraphpresnum}{}
595 \renewcommand*{\cftparagraphaftersnum}{}
596 \renewcommand*{\cftparagraphaftersnumb}{}
597 \renewcommand*{\cftparagraphleader}{}
598 \renewcommand*{\cftparagraphdotsep}{1}
599 \renewcommand*{\cftparagraphpagefont}{}
600 \renewcommand*{\cftparagraphafterpnum}{}
601 \renewcommand*{\cftparagraphformatpnum}[1]{}
602 \renewcommand*{\cftparagraphformatpnumhook}[1]{}

603 % \newlength{\cftbeforesubparagraphskip}
604 % \newlength{\cftsubparagraphindent}
605 % \newlength{\cftsubparagraphnumwidth}
606 \renewcommand*{\cftsubparagraphfont}{}
607 \renewcommand*{\cftsubparagraphname}{}
608 \renewcommand*{\cftsubparagraphpresnum}{}
609 \renewcommand*{\cftsubparagraphaftersnum}{}
610 \renewcommand*{\cftsubparagraphaftersnumb}{}
611 \renewcommand*{\cftsubparagraphleader}{}
612 \renewcommand*{\cftsubparagraphdotsep}{1}
613 \renewcommand*{\cftsubparagraphpagefont}{}
614 \renewcommand*{\cftsubparagraphafterpnum}{}
615 \renewcommand*{\cftsubparagraphformatpnum}[1]{}
```

```
616 \renewcommand*{\cftsubparagraphformatpnumhook}[1]{}

617 % \newlength{\cftbeforefigureskip}
618 % \newlength{\cftfigureindent}
619 % \newlength{\cftfigurenumwidth}
620 \renewcommand*{\cftfigurefont}{}
621 \renewcommand*{\cftfigurename}{}
622 \renewcommand*{\cftfigurepresnum}{}
623 \renewcommand*{\cftfigureaftersnum}{}
624 \renewcommand*{\cftfigureaftersnumb}{}
625 \renewcommand*{\cftfigureleader}{}
626 \renewcommand*{\cftfiguredotsep}{1}
627 \renewcommand*{\cftfigurepagefont}{}
628 \renewcommand*{\cftfigureafterpnum}{}
629 \renewcommand*{\cftfigureformatpnum}[1]{}
630 \renewcommand*{\cftfigureformatpnumhook}[1]{}

631 % \newlength{\cftbeforesubfigureskip}
632 % \newlength{\cftsubfigureindent}
633 % \newlength{\cftsubfigurenumwidth}
634 \newcommand*{\cftsubfigurefont}{}
635 \newcommand*{\cftsubfigurename}{}
636 \newcommand*{\cftsubfigurepresnum}{}
637 \newcommand*{\cftsubfigureaftersnum}{}
638 \newcommand*{\cftsubfigureaftersnumb}{}
639 \newcommand*{\cftsubfigureleader}{}
640 \newcommand*{\cftsubfiguredotsep}{1}
641 \newcommand*{\cftsubfigurepagefont}{}
642 \newcommand*{\cftsubfigureafterpnum}{}
643 \newcommand*{\cftsubfigureformatpnum}[1]{}
644 \newcommand*{\cftsubfigureformatpnumhook}[1]{}

645 % \newlength{\cftbeforetableskip}
646 % \newlength{\cfttableindent}
647 % \newlength{\cfttablenumwidth}
648 \renewcommand*{\cfttablefont}{}
649 \renewcommand*{\cfttablename}{}
650 \renewcommand*{\cfttablepresnum}{}
651 \renewcommand*{\cfttableaftersnum}{}
652 \renewcommand*{\cfttableaftersnumb}{}
653 \renewcommand*{\cfttableleader}{}
654 \renewcommand*{\cfttabledotsep}{1}
655 \renewcommand*{\cfttablepagefont}{}
656 \renewcommand*{\cfttableafterpnum}{}
657 \renewcommand*{\cfttableformatpnum}[1]{}
658 \renewcommand*{\cfttableformatpnumhook}[1]{}

659 % \newlength{\cftbeforesubtableskip}
660 % \newlength{\cftsubtableindent}
661 % \newlength{\cftsubtablenumwidth}
662 \newcommand*{\cftsubtablefont}{}
663 \newcommand*{\cftsubtablename}{}
664 \newcommand*{\cftsubtablepresnum}{}
665 \newcommand*{\cftsubtableaftersnum}{}
666 \newcommand*{\cftsubtableaftersnumb}{}
667 \newcommand*{\cftsubtableleader}{}
668 \newcommand*{\cftsubtabledotsep}{1}
669 \newcommand*{\cftsubtablepagefont}{}
670 \newcommand*{\cftsubtableafterpnum}{}
671 \newcommand*{\cftsubtableformatpnum}[1]{}
672 \newcommand*{\cftsubtableformatpnumhook}[1]{}
```

```
673 \renewcommand*{\booknumberline}[1]{}
674 \renewcommand*{\partnumberline}[1]{}
675 \renewcommand*{\chapternumberline}[1]{}
676 \renewcommand*{\numberlinehook}[1]{}
677 % \renewcommand*{\cftwhatismyname}{}%
678 \renewcommand*{\booknumberlinehook}[1]{}
679 \renewcommand*{\partnumberlinehook}[1]{}
680 \renewcommand*{\chapternumberlinehook}[1]{}
681 \renewcommand{\numberlinebox}[2]{}
682 \renewcommand{\booknumberlinebox}[2]{}
683 \renewcommand{\partnumberlinebox}[2]{}
684 \renewcommand{\chapternumberlinebox}[2]{}
685 %
686 % \newlength{\cftparfillskip}
687 \renewcommand*{\cftpagenumbersoff}[1]{}
688 \renewcommand*{\cftpagenumberson}[1]{}
689 \renewcommand*{\cftlocalchange}[3]{}
690 \renewcommand*{\cftaddtitleline}[4]{}
691 \renewcommand*{\cftaddnumtitleline}[4]{}
692 \renewcommand{\cftinsertcode}[2]{}
693 \renewcommand{\cftinserthook}[2]{}
694 \renewcommand{\settocpreprocessor}[2]{}
695 \DeclareRobustCommand{\cftpagenumbersoff}[1]{}
696 \DeclareRobustCommand{\cftpagenumberson}[1]{}
```

§ 695.11 **Floats and captions**

\@xfloat

\@xdblfloat Reestablish lwarp's takeover the float handing, which memoir tried to grab:

```
697 \AtBeginDocument{
698 \def\@xfloat #1[#2]{%
699     \LWR@floatbegin{#1}[#2]
700     \normalsize
701     \@nameuse{#1adjustment}%
702     \LWR@futurenonspacelet\LWR@mynexttoken\LWR@floatalignment%
703 }
704 \def\@xdblfloat #1[#2]{%
705     \LWR@floatbegin{#1}[#2]
706     \normalsize
707     \@nameuse{#1adjustment}%
708     \LWR@futurenonspacelet\LWR@mynexttoken\LWR@floatalignment%
709 }
710 }
```

\newfloat $[\langle 1:\,within\rangle]$ $\{\langle 2:\,type\rangle\}$ $\{\langle 3:\,ext\rangle\}$ $\{\langle 4:\,capname\rangle\}$

```
711 \RenewDocumentCommand{\newfloat}{o m m m}{%
712     \def\LWR@tempone{#4}%
713     \def\LWR@temptwo{\@nameuse{#2name}}%
714     \ifdefequal{\LWR@tempone}{\LWR@temptwo}{% recursive name, already defined
715         \IfValueTF{#1}%
716             {\DeclareFloatingEnvironment[fileext=#3,within=#1]{#2}}%
717             {\DeclareFloatingEnvironment[fileext=#3]{#2}}%
718     }{% not recursive name
719         \IfValueTF{#1}%
720           {\DeclareFloatingEnvironment[fileext=#3,within=#1,name={#4}]{#2}}%
721             {\DeclareFloatingEnvironment[fileext=#3,name={#4}]{#2}}%
722     }%
```

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.

```
723        \cslet{listof#2s}\relax%
724        \cslet{listof#2es}\relax%
725 }
```

\newlistof   [⟨*within*⟩] {⟨*type*⟩} {⟨*ext*⟩} {⟨*listofname*⟩}

Emulated through the \newfloat mechanism. Note that memoir uses a different
syntax than tocloft for the name.

```
726 \RenewDocumentCommand{\newlistof}{o m m m}
727 {%
728        \IfValueTF{#1}%
729            {\newlistentry[#1]{#2}{#3}{0}}%
730            {\newlistentry{#2}{#3}{0}}%
731        \@namedef{ext@#2}{#3}%
732        \@ifundefined{c@#3depth}{\newcounter{#3depth}}{}%
733        \setcounter{#3depth}{1}%
734        \@namedef{#3mark}{}%
735        \@namedef{#2}{\LWR@listof{#2}{#4}}%
736        \@namedef{@cftmake#3title}{}%
737        \@ifundefined{cftbefore#3titleskip}{%
738            \expandafter\newlength\csname cftbefore#3titleskip\endcsname%
739            \expandafter\newlength\csname cftafter#3titleskip\endcsname%
740        }{}%
741        \@namedef{cft#3titlefont}{}%
742        \@namedef{cftafter#3title}{}%
743        \@namedef{cft#3prehook}{}%
744        \@namedef{cft#3posthook}{}%
745 }
```

```
746 \renewcommand{\setfloatadjustment}[2]{}
```

Borrowed from the lwarp version of keyfloat:

```
747 \NewDocumentEnvironment{KFLTmemoir@marginfloat}{O{-1.2ex} m}
748 {% start
749   \LWR@BlockClassWP{float:right; width:2in; margin:10pt}{}(note){marginblock}%
750     \renewcommand*{\@captype}{#2}%
751 }
752 {%
753     \endLWR@BlockClassWP%
754 }
755
756 \DeclareDocumentEnvironment{marginfigure}{o}
757   {\begin{KFLTmemoir@marginfloat}{figure}}
758   {\end{KFLTmemoir@marginfloat}}
759
760 \DeclareDocumentEnvironment{margintable}{o}
761   {\begin{KFLTmemoir@marginfloat}{table}}
762   {\end{KFLTmemoir@marginfloat}}
```

```
763 \renewcommand{\setmarginfloatcaptionadjustment}[2]{}
764 \renewcommand{\setmpjustification}[2]{}
765 \renewcommand*{\mpjustification}{}
766 \renewcommand*{\setfloatlocations}[2]{}
767 \DeclareDocumentCommand{\suppressfloats}{o}{}
768 \renewcommand*{\FloatBlock}{}
```

```
769 \renewcommand*{\FloatBlockAllowAbove}{}
770 \renewcommand*{\FloatBlockAllowBelow}{}
771 \renewcommand*{\setFloatBlockFor}{}
772
773 \renewcommand{\captiontitlefinal}[1]{}
```

\flegtable, \flegfigure, \flegtoctable, \flegtocfigure are defined by memoir
using \newfloat. These are defined with an @ in ccaption.

```
774 \renewcommand{\flegtable}{\tablename}
775 \renewcommand{\flegfigure}{\figurename}
776 \renewcommand{\flegtoctable}{}
777 \renewcommand{\flegtocfigure}{}
```

```
778 \renewcommand{\@makesubfloatcaption}[2]{%
779     \minipagefullwidth
780     \begin{minipage}{\linewidth}%
781     #1 \ignorespaces #2 \unskip%
782     \end{minipage}
783 }
784
785 \renewcommand*{\tightsubcaptions}{}
786 \renewcommand*{\loosesubcaptions}{}
787
788 \renewcommand*{\subcaptionsize}[1]{}
789 \renewcommand*{\subcaptionlabelfont}[1]{}
790 \renewcommand*{\subcaptionfont}[1]{}
791 \renewcommand*{\subcaptionstyle}[1]{}
792
793 \renewcommand*{\hangsubcaption}{}
794 \renewcommand*{\shortsubcaption}{}
795 \renewcommand*{\normalsubcaption}{}
796
797 \RenewDocumentEnvironment{sidecaption}{o m o}
798 {}
799 {%
800     \IfValueTF{#1}{\caption[#1]{#2}}{\caption{#2}}%
801     \IfValueT{#3}{\label{#3}}%
802 }
803
804 % \newlength{\sidecapwidth}
805 % \newlength{\sidecapsep}
806 \renewcommand*{\setsidecaps}[2]{}
807 \renewcommand*{\sidecapmargin}[1]{}
808 % \newif\ifscapmargleft
809 \scapmargleftfalse
810 \renewcommand*{\setsidecappos}[1]{}
```

Env   sidecontcaption

```
811 \RenewDocumentEnvironment{sidecontcaption}{m o}
812 {}
813 {%
814     \ifdef{\ContinuedFloat}%
815         {\ContinuedFloat}%
816         {\addtocounter{\@captype}{-1}}%
817     \caption{#1}%
```

Without \@captype, the section is referred to instead.

```
818     \IfValueT{#2}{\label[\@captype]{#2}}%
819 }
```

\sidenamedlegend does not appear to use the TOC argument.

```
820 \renewenvironment{sidenamedlegend}[2][]{
821     \begin{center}
822     \@nameuse{\@captype name}\CaptionSeparator#2
823     \end{center}
824 }
825 {}
826
827 \renewenvironment{sidelegend}[1]
828 {\begin{center}
829     #1
830
831 }
832 {\end{center}}
833
834 \renewcommand*{\sidecapstyle}{}
835 \renewcommand*{\overridescapmargin}[1]{}
836 % \newlength{\sidecapraise}
837 \renewcommand*{\sidecapfloatwidth}{\linewidth}
838
839 \LetLtxMacro\ctabular\tabular
840 \LetLtxMacro\endctabular\endtabular
841
842 \renewcommand{\autorows}[5][]{%
843     #5%
844 }
845
846 \renewcommand{\autocols}[5][]{%
847     #5%
848 }
```

§ 695.12   **Footnotes and page notes**

```
849 \renewcommand*{\feetabovefloat}{}
850 \renewcommand*{\feetbelowfloat}{}
851 \renewcommand*{\feetatbottom}{}
852
853 \renewcommand*{\verbfootnote}[2][]{%
854     \PackageError{lwarp,memoir}%
855     {Verbatim footnotes are not yet supported by lwarp}%
856     {This may be improved some day.}%
857 }
858
859 \renewcommand*{\plainfootnotes}{}
860 \renewcommand*{\twocolumnfootnotes}{}
861 \renewcommand*{\threecolumnfootnotes}{}
862 \renewcommand*{\paragraphfootnotes}{}
863 \renewcommand*{\footfudgefiddle}{}
864
865 \renewcommand*{\newfootnoteseries}[1]{%
866     \PackageError{lwarp,memoir}%
867     {Memoir footnote series are not yet supported by lwarp}%
868     {This may be improved some day.}%
869 }
870
```

```
871 \renewcommand*{\plainfootstyle}[1]{}
872 \renewcommand*{\twocolumnfootstyle}[1]{}
873 \renewcommand*{\threecolumnfootstyle}[1]{}
874 \renewcommand*{\paragraphfootstyle}[1]{}
875
876 \renewcommand*{\footfootmark}{}
877 \renewcommand*{\footmarkstyle}[1]{}
878
879 % \newlength{\footmarkwidth}
880 % \newlength{\footmarksep}
881 % \newlength{\footparindent}
882
883 \renewcommand*{\foottextfont}{}
884
885 \renewcommand*{\marginparmargin}[1]{}
886 \renewcommand*{\sideparmargin}[1]{}
887
888 \LetLtxMacro\sidepar\marginpar
889 \renewcommand*{\sideparfont}{}
890 \renewcommand*{\sideparform}{}
891 \LWR@providelength{\sideparvshift}
892
893 \renewcommand*{\parnopar}{}
894
895 \renewcommand{\sidebar}[1]{\begin{quote}#1\end{quote}}
896 \renewcommand*{\sidebarmargin}[1]{}
897 \renewcommand*{\sidebarfont}{}
898 \renewcommand*{\sidebarform}{}
899 % \newlength{\sidebarhsep}
900 % \newlength{\sidebarvsep}
901 % \newlength{\sidebarwidth}
902 % \newlength{\sidebartopsep}
903 \renewcommand{\setsidebarheight}[1]{}
904 \renewcommand*{\setsidebars}[6]{}
905 \renewcommand*{\footnotesatfoot}{}
906 \renewcommand*{\footnotesinmargin}{}
907
908 \LetLtxMacro\sidefootnote\footnote
909 \LetLtxMacro\sidefootnotemark\footnotemark
910 \LetLtxMacro\sidefootnotetext\footnotetext
911
912 \renewcommand*{\sidefootmargin}[1]{}
913 % \newlength{\sidefoothsep}
914 % \newlength{\sidefootvsep}
915 % \newlength{\sidefootwidth}
916 % \newlength{\sidefootadjust}
917 % \newlength{\sidefootheight}
918 \renewcommand*{\setsidefootheight}[1]{}
919 % \renewcommand*{\sidefootfont}{}% in docs but not in the package
920 \renewcommand*{\setsidefeet}[6]{}
921 \renewcommand*{\sidefootmarkstyle}[1]{}
922 \renewcommand*{\sidefoottextfont}{}
923 \renewcommand*{\sidefootform}{}

924 \renewcommand*{\continuousnotenums}{\pncontopttrue}% from pagenote
925 \renewcommand*{\notepageref}{}
926 \renewcommand*{\prenotetext}{}
927 \renewcommand*{\postnotetext}{}
928 \LetLtxMacro\printpageinnoteshyperref\printpageinnotes
929 \renewcommand*{\foottopagenote}{}
```

930 `\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.

```
931 \xpatchcmd{\m@m@wrpnote}
932     {\string\startnoteentry{\thepagenote}}
933     {\string\startnoteentry{{\thepagenote}{\arabic{pagenote}}}}
934     {}
935     {\LWR@patcherror{memoir}{m@m@wrpnote}}
936
937 \renewcommand\startnoteentrystart[4]{%
938    \prenoteinnotes%
939    \noteidinnotes{\@firstoftwo#1}{#2}%
940    \@ifmtarg{#2}{%
941 %        \phantomsection\def\@currentlabel{#1}%        original
942       \def\@currentlabel{\@firstoftwo#1}%             lwarp
943       \def\cref@currentlabel{%                        lwarp
944          [pagenote][\@secondoftwo#1][]\@firstoftwo#1%  lwarp
945       }%                                               lwarp
946    }{}%
947    \pagenoteanchor{#4}%
948    \pageinnotes{#3}%
949    \prenotetext%
950 }
```

§ 695.13 **Decorative text**

```
951 \renewcommand*{\epigraphposition}[1]{}
952 \renewcommand*{\epigraphtextposition}[1]{}
953 \renewcommand*{\epigraphsourceposition}[1]{}
954 \renewcommand*{\epigraphfontsize}[1]{}
955 \renewcommand*{\epigraphforheader}[2][]{}
956 \renewcommand*{\epigraphpicture}{}
```

§ 695.14 **Poetry**

```
957 \renewcommand*{\vinphantom}{}
958 \renewcommand*{\vleftofline}[1]{#1}
959 % \let\linenumberfrequency\poemlines
960 % \renewcommand*{\linenumberfont}[1]{}
961
962 \DeclareDocumentCommand{\PoemTitle}{s o o m}{%
963    \IfValueTF{#2}%
964       {\poemtitle[#2]{#4}}%
965       {\poemtitle{#4}}%
966 }
967
968 \renewcommand*{\NumberPoemTitle}{}
969 \renewcommand*{\PlainPoemTitle}{}
970 \renewcommand*{\poemtitlepstyle}{}
```

```
971 \renewcommand*{\poemtitlestarmark}[1]{}
972 \renewcommand*{\poemtitlestarpstyle}{}
973 \renewcommand*{\PoemTitleheadstart}{}
974 \renewcommand*{\printPoemTitlenonum}{}
975 \renewcommand*{\printPoemTitlenum}{}
976 \renewcommand*{\afterPoemTitlenum}{}
977 \renewcommand*{\printPoemTitletitle}[1]{}
978 \renewcommand*{\afterPoemTitle}{}
979 \newlength{\midpoemtitleskip}
980 \renewcommand*{\PoemTitlenumfont}{}
981 \renewcommand*{\PoemTitlefont}{}
```

§ 695.15 **Boxes, verbatims and files**

```
982 \renewenvironment{qframe}{\framed}{\endframed}
983 \renewenvironment{qshade}{\shaded}{\endshaded}


984 \renewcommand*{\setverbatimfont}[1]{}
985 \renewcommand*{\tabson}[1]{}
986 \renewcommand*{\tabsoff}{}
987 \renewcommand*{\wrappingon}{}
988 \renewcommand*{\wrappingoff}{}
989 \renewcommand*{\verbatimindent}{}
990 \renewcommand*{\verbatimbreakchar}[1]{}


991 \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.

```
992 \renewcommand*{\bvbox}{}
993 \renewcommand*{\bvtopandtail}{}
994 \renewcommand*{\bvsides}{}
995 \renewcommand*{\nobvbox}{}
996 % \newlength\bvboxsep
997 \renewcommand*{\bvtoprulehook}{}
998 \renewcommand*{\bvtopmidhook}{}
999 \renewcommand*{\bvendrulehook}{}
1000 \renewcommand*{\bvleftsidehook}{}
1001 \renewcommand*{\bvrightsidehook}{}
1002 \renewcommand*{\bvperpagetrue}{}
1003 \renewcommand*{\bvperpagefalse}{}
1004 \renewcommand{\bvtopofpage}[1]{}
1005 \renewcommand{\bvendofpage}[1]{}
1006 \renewcommand*{\linenumberfrequency}[1]{}
1007 \renewcommand*{\resetbvlinenumber}{}
1008 \renewcommand*{\setbvlinenums}[2]{}
1009 \renewcommand*{\linenumberfont}[1]{}
1010 \renewcommand*{\bvnumbersinside}{}
1011 \renewcommand*{\bvnumbersoutside}{}
```

§ 695.16 **Cross referencing**

```
1012 \renewcommand*{\fref}[1]{\cref{#1}}
1013 \renewcommand*{\tref}[1]{\cref{#1}}
1014 \renewcommand*{\pref}[1]{\cpageref{#1}}
1015 \renewcommand*{\Aref}[1]{\cref{#1}}
1016 \renewcommand*{\Bref}[1]{\cref{#1}}
1017 \renewcommand*{\Pref}[1]{\cref{#1}}
```

```
1018 \renewcommand*{\Sref}[1]{\cref{#1}}
1019 \renewcommand*{\figurerefname}{Figure}
1020 \renewcommand*{\tablerefname}{Table}
1021 \renewcommand*{\pagerefname}{page}
1022 \renewcommand*{\bookrefname}{Book~}
1023 \renewcommand*{\partrefname}{Part~}
1024 \renewcommand*{\chapterrefname}{Chapter~}
1025 \renewcommand*{\sectionrefname}{\S}
1026 \renewcommand*{\appendixrefname}{Appendix~}
1027 \LetLtxMacro\titleref\nameref
1028 \renewcommand*{\headnameref}{}
1029 \renewcommand*{\tocnameref}{}
1030
1031 \providecounter{LWR@currenttitle}
1032
1033 \renewcommand*{\currenttitle}{%
1034     \addtocounter{LWR@currenttitle}{1}%
1035     \label{currenttitle\arabic{LWR@currenttitle}}%
1036     \nameref{currenttitle\arabic{LWR@currenttitle}}%
1037 }
1038
1039 \renewcommand*{\theTitleReference}[2]{}
1040 \renewcommand*{\namerefon}{}
1041 \renewcommand*{\namerefoff}{}
```

§ 695.17 **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.

```
1042 \AtBeginDocument{
1043
1044 \def\@@wrindexhyp#1||\\{%
1045     \addtocounter{LWR@autoindex}{1}%                    lwarp
1046 %     \ifshowindexmark\@showidx{#1}\fi
1047     \protected@write\@auxout{}%
1048 %         {\string\@@wrindexm@m{\@idxfile}{#1}{\thepage}}%
1049       {\string\@@wrindexm@m{\@idxfile}{#1}{\arabic{LWR@autoindex}}}% lwarp
```

The label is assigned after the file write to avoid conflict with cleveref.

```
1050     \label{LWRindex-\arabic{LWR@autoindex}}%      lwarp
1051     \endgroup
1052     \@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.

```
1053 \def\@@wrspindexhyp#1||\\{%
1054     \addtocounter{LWR@autoindex}{1}%
1055 %     \ifshowindexmark\@showidx{#1}\fi
1056     \protected@write\@auxout{}%
1057 %         {\string\@@wrindexm@m{\@idxfile}{#1}{\@nameuse{the\@sptheidx}}}%
1058       {\string\@@wrindexm@m{\@idxfile}{#1}{\arabic{LWR@autoindex}}}%
```

The label is assigned after the file write to avoid conflict with cleveref.

```
1059     \label{LWRindex-\arabic{LWR@autoindex}}%
1060     \endgroup
1061     \@esphack}%
1062
```

1063 }% \AtBeginDocument

\@spindex   Patched to append _html to the file:

1064 \renewcommand{\@spindex}[2]{%
1065   \@ifundefined{#1@idxfile}%
1066   {\ifreportnoidxfile
1067   \@memwarn{Undefined index file #1}%
1068   \fi
1069   \begingroup
1070   \@sanitize
1071   \@nowrindex}%
1072   {\def\@idxfile{#1_html}%
1073   \def\@sptheidx{#2}%
1074   \begingroup
1075   \@sanitize
1076   \@wrspindex}}

\makeindex   Patched to use _html filename and \BaseJobname:

1077 \catcode'_=12%
1078 \renewcommand*{\makeindex}[1][\BaseJobname]{%
1079   \if@filesw
1080   \def\gindex{\@bsphack%
1081   \@ifnextchar [{\@index}{\@index[\BaseJobname]}}
1082   \def\specialindex{\@bsphack\@spindex}%
1083   \makememindexhook
1084   \expandafter\newwrite\csname #1@idxfile\endcsname
1085   \expandafter\immediate\openout \csname #1@idxfile\endcsname #1_html.idx\relax
1086   \typeout{Writing index file #1_html.idx }%
1087   \fi}
1088 \catcode'_=8%

\printindex   Patched to use _html filename and \BaseJobname. This will later be patched by
                             the lwarp core.

1089 \catcode'_=12%
1090 \renewcommand{\printindex}[1][\BaseJobname]{\@input@{#1_html.ind}}
1091 \catcode'_=8%

1092 \DeclareDocumentCommand{\newblock}{}{}
1093 %
1094 \renewcommand*{\showindexmarks}{}
1095 \renewcommand*{\hideindexmarks}{}
1096
1097 \renewcommand*{\xindyindex}{}

§ 695.18 **Miscellaneous**

1098 \renewcommand*{\changemarks}{}
1099 \renewcommand*{\nochangemarks}{}
1100 \renewcommand*{\added}[1]{}
1101 \renewcommand*{\deleted}[1]{}
1102 \renewcommand*{\changed}[1]{}
1103
1104 \renewcommand*{\showtrimsoff}{}
1105 \renewcommand*{\showtrimson}{}
1106 \renewcommand*{\trimXmarks}{}
1107 \renewcommand*{\trimLmarks}{}

```
1108 \renewcommand*{\trimFrame}{}
1109 \renewcommand*{\trimNone}{}
1110 \renewcommand*\trimmarkscolor{}
1111 \renewcommand*{\trimmarks}{}
1112 \renewcommand*{\tmarktl}{}
1113 \renewcommand*{\tmarktr}{}
1114 \renewcommand*{\tmarkbr}{}
1115 \renewcommand*{\tmarkbl}{}
1116 \renewcommand*{\tmarktm}{}
1117 \renewcommand*{\tmarkmr}{}
1118 \renewcommand*{\tmarkbm}{}
1119 \renewcommand*{\tmarkml}{}
1120 \renewcommand*{\trimmark}{}
1121 \renewcommand*{\quarkmarks}{}
1122 \renewcommand*{\registrationColour}[1]{}
1123
1124 \renewcommand*{\leavespergathering}[1]{}
1125
1126 \renewcommand*{\noprelistbreak}{}
1127
1128 \renewcommand*{\cleartorecto}{}
1129 \renewcommand*{\cleartoverso}{}
1130
1131 \renewenvironment{vplace}[1][]{}{}
```

§ 695.19   ccaption emulation

```
1132 \renewcommand*{\captiondelim}[1]{\renewcommand*{\CaptionSeparator}{#1}}
1133 \renewcommand*{\captionnamefont}[1]{}
1134 \renewcommand*{\captiontitlefont}[1]{}
1135 \renewcommand*{\flushleftright}{}
1136 \renewcommand*{\centerlastline}{}
1137 \renewcommand*{\captionstyle}[2][]{}
1138 \DeclareDocumentCommand{\captionwidth}{m}{}
1139 \renewcommand*{\changecaptionwidth}{}
1140 \renewcommand*{\normalcaptionwidth}{}
1141 \renewcommand*{\hangcaption}{}
1142 \renewcommand*{\indentcaption}[1]{}
1143 \renewcommand*{\normalcaption}{}
1144 \renewcommand{\precaption}[1]{}
1145 \renewcommand{\postcaption}[1]{}
1146 \renewcommand{\midbicaption}[1]{}
1147 \renewcommand{\contcaption}[1]{%
1148 %   \ContinuedFloat%
1149 %   \caption{#1}%
1150    \begin{LWR@figcaption}% later becomes \caption*
1151    \LWR@isolate{\@nameuse{\@captype name}}~%
1152    \thechapter.\the\value{\@captype}\CaptionSeparator\LWR@isolate{#1}%
1153    \end{LWR@figcaption}%
1154 }
1155 \newlength{\abovelegendskip}
1156 \setlength{\abovelegendskip}{0.5\baselineskip}
1157 \newlength{\belowlegendskip}
1158 \setlength{\belowlegendskip}{\abovelegendskip}
```

The extra \\ here forces a <br> in HTML when \legend is used in a \marginpar.

```
1159 \renewcommand{\legend}[1]{\begin{center}#1\\\end{center}}
1160
```

```
1161 \renewcommand{\namedlegend}[2][]{%
1162     \begin{center}
1163     \@nameuse{fleg\@captype}\CaptionSeparator#2\\
1164     \end{center}
1165     \@nameuse{flegtoc\@captype}{#1}
1166 }
```

\flegtable, \flegfigure, \flegtoctable, \flegtocfigure are defined by memoir using \newfloat. These are defined with an @ in ccaption.

```
1167 \renewcommand{\newfixedcaption}[3][\caption]{%
1168   \renewcommand{#2}{\def\@captype{#3}#1}}
1169 \renewcommand{\renewfixedcaption}[3][\caption]{%
1170   \renewcommand{#2}{\def\@captype{#3}#1}}
1171 \renewcommand{\providefixedcaption}[3][\caption]{%
1172   \providecommand{#2}{\def\@captype{#3}#1}}
1173
1174 \renewcommand{\bitwonumcaption}[6][]{%
1175     \ifblank{#2}{\caption{#3}}{\caption[#2]{#3}}%
1176     \addtocounter{\@captype}{-1}%
1177     \begingroup%
1178     \csdef{\@captype name}{#4}%
1179     \ifblank{#5}{\caption{#6}}{\caption[#5]{#6}}%
1180     \endgroup%
1181     \ifblank{#1}{}{\label{#1}}%
1182 }
1183
1184 \LetLtxMacro\bionenumcaption\bitwonumcaption% todo
1185
1186 \renewcommand{\bicaption}[5][]{%
1187     \ifblank{#2}{\caption{#3}}{\caption[#2]{#3}}%
1188     \begin{LWR@figcaption}% later becomes \caption*
1189     \LWR@isolate{#4} % space
1190     \thechapter.\the\value{\@captype}\CaptionSeparator\LWR@isolate{#5}%
1191     \end{LWR@figcaption}%
1192     \ifblank{#1}{}{\label{#1}}%
1193 }
1194
1195 \renewcommand{\bicontcaption}[3]{%
1196     \contcaption{#1}%
1197     \begingroup%
1198     \csdef{\@captype name}{#2}%
1199     \contcaption{#3}%
1200     \endgroup%
1201 }
```

Only in ccaption, not in memoir:

```
1202 % \LetLtxMacro\longbitwonumcaption\bitwonumcaption%
1203 % \LetLtxMacro\longbionenumcaption\bitwonumcaption%
1204 % \LetLtxMacro\longbicaption\bicaption%
```

Patches for subfloats to support additional lwarp labels:

```
1205 \renewcommand{\@memsubbody}{%
1206   \bgroup
1207   \let\label=\memsub@label
1208   \ifdonemaincaption\else
1209     \advance\csname c@\@captype\endcsname\@ne
1210   \fi
```

```
1211   % \refstepcounter{sub\@captype}\@contkeep%
1212 %    \leavevmode%                  lwarp
1213   \@ifnextchar [%
1214     {\@memsubfig}%
1215     {\@memsubfig[\@empty]}}
1216
1217 \renewcommand{\@memcontsubbody}{%
1218   \bgroup
1219   \let\label=\memsub@label
1220   \@contset
1221   % \refstepcounter{sub\@captype}\@contkeep%
1222 %    \leavevmode%                  lwarp
1223   \@ifnextchar [%
1224     {\@memsubfig}%
1225     {\@memsubfig[\@empty]}}
1226
1227
1228 \long\def\@memsubfloat#1[#2][#3]#4{%
1229 %    \@tempcnta=\@ne
1230 %    \if@tightsubcap
1231 %      \if@minipage
1232 %        \@tempcnta=\z@
1233 %      \else
1234 %        \ifdim\lastskip=\z@
1235 %          \@tempcnta=\@ne
1236 %        \else
1237 %          \@tempcnta=\tw@
1238 %        \fi
1239 %      \fi
1240 %    \fi
1241 %    \if@contbotsub
1242 %      \def\subfig@top{\subfloattopskip}%
1243 %      \def\subfig@bottom{\subfloatbottomskip}%
1244 %    \else
1245 %      \def\subfig@top{\subfloatbottomskip}%
1246 %      \def\subfig@bottom{\subfloattopskip}%
1247 %    \fi
1248 %    \setbox\@tempboxa \hbox{#4}%
1249 %    \@tempdima=\wd\@tempboxa
1250 %    \vbox
1251   \bgroup%
1252     \mem@step@subcounter%
1253 %      \vbox
1254     \LWR@stoppars%
1255     \minipagefullwidth%                  lwarp
1256     \begin{minipage}{\linewidth}%        lwarp
1257     \bgroup
1258 %    \ifcase\@tempcnta
1259 %      \@minipagefalse
1260 %    \or
1261 %      \vspace{\subfig@top}
1262 %    \or
1263 %      \ifdim \lastskip=\z@ \else
1264 %        \@tempskipb\subfig@top\@xaddvskip
1265 %      \fi
1266 %    \fi
1267     \if@contbotsub
1268       #4% \box\@tempboxa
1269       \egroup
1270       \ifx \@empty#3\relax \else
```

```
1271 %         \vskip\subfloatcapskip
1272        \@memsubcaption{#1}{#2}{#3}%
1273      \fi
1274    \else
1275      \ifx \@empty#3\relax \else
1276        \@memsubcaption{#1}{#2}{#3}%
1277 %         \vskip\subfloatcapskip
1278 %         \vskip\subfloatcaptopadj
1279      \fi\egroup
1280      #4% \box\@tempboxa
1281    \fi
1282 %     \vspace{\subfig@bottom}
1283    \end{minipage}%                    lwarp
1284    \LWR@startpars%                     lwarp
1285  \egroup
1286 \egroup
1287 }
```

§ 695.20 **Final patchwork**

```
1288 \newlistof{tableofcontents}{toc}{\contentsname}
1289 \newlistof{listoffigures}{lof}{\listfigurename}
1290 \newlistof{listoftables}{lot}{\listtablename}
```

File 587 **lwarp-common-multimedia.sty**

§ 696   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 588 **lwarp-common-mathjax-letters.sty**

§ 697   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 }
```

\LWR@mathjax@addgreek@l@it

* {⟨*2: prefix*⟩} {⟨*3: postfix*⟩}

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 }
```

* {⟨*2: prefix*⟩} {⟨*3: postfix*⟩}

\LWR@mathjax@addgreek@l@bfit

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 }
```

\LWR@mathjax@addgreek@u@bfit

    * {⟨*2: prefix*⟩} {⟨*3: postfix*⟩}

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 589   **`lwarp-common-mathjax-newpxtxmath.sty`**

§ 698   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 590   **`lwarp-common-mathjax-nonunicode.sty`**

§ 699   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 591   **lwarp-common-mathjax-overlaysymbols.sty**

§ 700   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

§ 701 **Chg Hist**

For the most recent changes, see page .

1304

1314

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.

1347

1351

1353

1355

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 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.

Index of Indexes