Lwarp

LaTeX HTML5

The lwarp package

LaTeX to HTML

v0.67 — 2019/02/23

© 2016–2019 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. Hundreds of LaTeX packages are supported.

Documents may be produced by DVI or PDF LaTeX, LuaLaTeX, XeLaTeX; 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 testing of package load order and image generation offers useful advice for resolving errors.

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.

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

To update existing projects, see section 2: Updates.

Lwarp is still in development. Changes are likely.

License:

This work may be distributed and/or modified under the conditions of the LaTeX Project Public License, either version 1.3 of this license or (at your option) any later version. The latest version of this license is in http://www.latex-project.org/lppl.txt and version 1.3 or later is part of all distributions of LaTeX version 2005/12/01 or later.

1 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 pdfTEX 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

1 Support TEX development 2

List of Figures 52

List of Tables 52

2 Updates 53

3 Introduction 72
3.1 Typesetting conventions . . . 74
3.2 Supported packages and features . . . 75

4 Alternatives 80
4.1 internet class . . . 80
4.2 TEX4HT . . . 80
4.3 Translators . . . 80
4.4 ASCIIDOC and ASCIIDOCTOR . . . 81
4.4.1 ASCIIDOCTOR-LATEX . . . 81
4.5 PANDOC . . . 81
4.6 Word processors . . . 81
4.7 Commercial systems . . . 81
4.8 Comparisons . . . 81

5 Installation 83
5.1 Installing the lwarp package . . . 85
5.2 Installing the *lwarpmk* utility . . . 86
5.2.1 Using a local copy of *lwarpmk* . . . 88
5.3 Installing additional utilities . . . 88

6 Tutorial 90
6.1 Starting a new project . . . 90
6.2 Compiling the print version with *lwarpmk* . . . 94
6.3 Compiling the HTML version with *lwarpmk* . . . 95
6.4 Generating the SVG images . . . 96
6.5 Using MATHJAX for math . . . 97
6.6 Changing the CSS style . . . 98
6.7 Customizing the HTML output . . . 98
6.8 Using *latexmk* . . . 99
6.9 Using XELATEX or LUALATEX . . . 100
6.10 Using DVI LATEX . . . 100
6.11 Using a glossary . . . 101
6.11.1 gloss package . . . 101
6.11.2 glossaries package . . . 101
6.12 Cleaning auxiliary files . . . 102
6.13 Cleaning auxiliary and output files . . . 102
6.14 Cleaning the images from the `<project>-images` directory . . . 102
6.15 Converting PDF or EPS images to SVG . . . 102
6.16 Creating HTML from an incomplete compile . . . 102
6.17 Processing multiple projects in the same directory . . . 102
6.18 Using the *make* utility . . . 103

7 Converting an existing document 104

8 Additional details 105
8.1 Shell escape . . . 105
8.2 Font and UTF-8 support . . . 105
8.2.1 Indexes, glossaries, and encoding . . . 106
8.3 lwarp package loading and options . . . 107
8.4 Customizing the HTML output . . . 111
8.4.1 Example HTML file naming . . . 116
8.5 Customizing the CSS . . . 117
8.6 Selecting the operating system . . . 118
8.7 Selecting actions for print or HTML output . . . 118
8.8 Commands to be placed into the warpprint environment . . . 119
8.9 Title page . . . 120
8.10 HTML page meta descriptions . . . 121
8.11 HTML homepage meta title . . . 121
8.12 HTML page meta author . . . 121

9 Special cases and limitations 121
9.1 Things to avoid . . . 122
9.1.1 Invalid HTML . . . 122
9.2 Formatting . . . 122
9.2.1 Text formatting . . . 122
9.2.2 Horizontal space . . . 123
9.2.3 Text alignment . . . 123
9.2.4 Accents . . . 123
9.2.5 textcomp package . . . 123
9.2.6 Superscripts and other non-math uses of math mode . . . 123
9.2.7 Empty \item followed by a new line of text or a nested list: . . . 124
9.2.8 Filenames and URLs in lists or footnotes . . . 124
9.2.9 relsize package . . . 124
9.3 Boxes and minipages . . . 124
9.3.1 Marginpars . . . 124
9.3.2 Save Boxes . . . 124
9.3.3 Minipages . . . 125
9.3.4 Side-by-side minipages . . . 125
9.3.5 Framed minipages and other environments . . . 126
9.3.6 fancybox package . . . 127
9.3.7 mdframed package . . . 128
9.4 Section names . . . 129
9.5 Cross-references . . . 129
9.5.1 Page references . . . 130
9.5.2 cleveref and varioref packages . . . 130
9.5.3 Hyperlinks, hyperref, and url . . . 130
9.5.4 Footnotes and page notes . . . 130
9.6 Front and back matter . . . 131
9.6.1 Custom classes with multiple authors and affiliations . . . 131
9.6.2 Starred chapters and sections . . . 132
9.6.3 abstract package . . . 132
9.6.4 titling and authblk . . . 132
9.6.5 tocloft package . . . 133

9.6.6 appendix package . . . 133
9.6.7 pagenote package . . . 133
9.6.8 endnotes package . . . 133
9.6.9 BibTeX . . . 133
9.6.10 gloss package . . . 134
9.6.11 glossaries package . . . 134
9.6.12 nomencl package . . . 135
9.6.13 Indexing overview . . . 135
9.6.14 Indexing with basic LaTeX and makeidx . . . 135
9.6.15 Indexing with index . . . 136
9.6.16 Indexing with splitidx . . . 137
9.6.17 Indexing with imakeidx . . . 139
9.6.18 Indexes with memoir . . . 142
9.6.19 Using a custom *makeindex* style file . . . 144
9.6.20 Using a custom *xindy* style file . . . 145
9.6.21 Additional indexing limitations . . . 145
9.6.22 Index positions, toc, tocbibind . . . 146
9.7 Math . . . 147
9.7.1 Math in section names . . . 147
9.7.2 Rendering tradeoffs . . . 147
9.7.3 svg option . . . 148
9.7.4 MathJax option . . . 148
9.7.5 Customizing MathJax . . . 149
9.7.6 MathJax limitations . . . 149
9.7.7 Catcode changes . . . 150
9.7.8 Complicated inline math objects . . . 150
9.7.9 Complicated display math objects . . . 150
9.7.10 ntheorem package . . . 150
9.7.11 siunitx package . . . 151
9.7.12 units and nicefrac packages . . . 151
9.7.13 newtxmath package . . . 151
9.8 Graphics . . . 151
9.8.1 tikz package . . . 153
9.8.2 grffile package . . . 153
9.8.3 color package . . . 153
9.8.4 xcolor package . . . 153
9.8.5 epstopdf package . . . 154
9.8.6 pstricks package . . . 154
9.8.7 pdftricks package . . . 154
9.8.8 psfrag package . . . 154
9.8.9 pstool package . . . 155
9.8.10 asymptote package . . . 155
9.8.11 overpic package . . . 155
9.9 Tabbing . . . 155
9.10 Tabular . . . 156
9.10.1 longtable package . . . 158
9.10.2 supertabular and xtab packages . . . 159
9.10.3 bigdelim package . . . 159
9.11 Floats . . . 159
9.11.1 Float contents alignment . . . 159
9.11.2 float, trivfloat, and/or algorithmicx together . . . 159

9.11.3 caption and subcaption packages . . . 160
9.11.4 subfig package . . . 160
9.11.5 floatrow package . . . 160
9.11.6 keyfloat package . . . 161
9.12 KOMA-SCRIPT classes . . . 161
9.13 MEMOIR class . . . 161
9.14 International languages . . . 162
9.15 Miscellaneous packages . . . 162
9.15.1 verse and memoir . . . 162
9.15.2 newclude package . . . 163
9.15.3 babel package . . . 163
9.15.4 polyglossia package . . . 163
9.15.5 todonotes and luatodonotes packages . . . 164
9.15.6 fixme . . . 164
9.15.7 chemfig package . . . 164
9.15.8 chemformula package . . . 164
9.15.9 mhchem package . . . 164
9.15.10 xparse package . . . 164
9.15.11 kotex package . . . 165

10 Compiling using custom shell commands 166
10.1 Command options . . . 166
10.2 Literal character macros . . . 166
10.3 *latexmk* . . . 167
10.4 perltex package . . . 168
10.5 pythontex package . . . 168
10.6 Other packages . . . 168
10.7 *make* program . . . 169
10.8 UTF-8 locale . . . 169

11 EPUB conversion 170

12 Word-processor conversion 172
12.1 Activating word-processor conversion . . . 172
12.2 Additional modifications . . . 173
12.3 Recommendations . . . 175
12.4 Limitations . . . 176

13 Modifying lwarp 177
13.1 Creating a development system . . . 177
13.2 Modifying a package for lwarp . . . 178
13.2.1 Adding a package to the lwarp.dtx file . . . 179
13.3 Modifying a class for lwarp . . . 179
13.4 Testing lwarp . . . 180
13.5 Modifying *lwarpmk* . . . 180

14 Troubleshooting 182
14.1 Using the lwarp.sty package . . . 182
14.1.1 Debug tracing output . . . 185
14.2 Compiling the `lwarp.dtx` file . . . 185

15 Trademarks 187

1 lwarp.sty 188

16 Implementation 188

17 Section depths and HTML headings 189

18 Source code 190

19 Detecting the TEX engine — *pdflatex*, *lualatex*, *xelatex* 191

20 Early package requirements 191

21 Package load order 191
21.1 Tests of package load order . . . 192
21.2 Error for disallowed packages and classes loaded before lwarp . . . 194
21.3 Enforcing package loading after lwarp . . . 195

22 MD5 hashing 203

23 pdfLATEX T1 and UTF-8 encoding 203

24 Unicode input characters 204

25 Upright quotes 204

26 Miscellaneous tools 205

27 Operating-System portability 206
27.1 Literal characters . . . 206
27.2 Common portability code . . . 207
27.3 UNIX, LINUX, and MAC OS . . . 207
27.4 MS-WINDOWS . . . 208

28 Package options 208
28.1 Additional options support . . . 212
28.2 Conditional compilation . . . 214

29 Required packages 215

30 Loading packages 219

31 Additional required packages 224

32 File handles 224

33 Include a file 224

34 Copying a file 225

35 Debugging messages 226

36 Defining print and HTML versions of macros and environments 227

37 HTML-conversion output modifications 231
37.1 User-level controls . . . 232

37.2 Heading adjustments . . . 233

38 Remembering original formatting macros 235

39 Accents 237

40 Configuration files 238
40.1 Decide whether to generate configuration files . . . 238
40.2 <project>_html.tex . . . 239
40.3 *lwarpmk* configuration files . . . 239
40.3.1 Helper macros . . . 239
40.3.2 lwarpmk.conf . . . 245
40.3.3 <project>.lwarpmkconf . . . 245
40.4 lwarp.css . . . 245
40.5 lwarp_sagebrush.css . . . 273
40.6 lwarp_formal.css . . . 278
40.7 sample_project.css . . . 282
40.8 lwarp.ist . . . 282
40.9 lwarp.xdy . . . 283
40.10 lwarp_one_limage.cmd . . . 283
40.11 lwarp_mathjax.txt . . . 284
40.12 lwarpmk.lua — lwarpmk option . . . 286

41 Stacks 303
41.1 Assigning depths . . . 303
41.2 Closing actions . . . 304
41.3 Closing depths . . . 304
41.4 Pushing and popping the stack . . . 305

42 Data arrays 306

43 Localizing catcodes 306

44 Localizing dynamic math 307

45 HTML entities 308

46 HTML filename generation 309

47 Homepage link 312

48 \LWRPrintStack diagnostic tool 313

49 Closing stack levels 313

50 PDF pages and styles 314

51 HTML tags, spans, divs, elements 315
51.1 Mapping LaTeX sections to HTML sections . . . 316
51.2 Babel-French tag modifications . . . 316
51.3 HTML output formatting . . . 317
51.4 HTML tags . . . 317
51.5 Block tags and comments . . . 319

51.6 Div class and element class . . . 320
51.7 Single-line elements . . . 321
51.8 HTML5 semantic elements . . . 322
51.9 High-level block and inline classes . . . 322
51.10 Closing HTML tags . . . 324

52 Paragraph handling 325

53 Paragraph start/stop handling 328

54 Indentfirst 331

55 Page headers and footers 331

56 CSS 332

57 Title, HTML meta author, HTML meta description 332

58 Footnotes 333
58.1 Regular page footnotes . . . 334
58.2 Minipage footnotes . . . 334
58.3 Titlepage thanks . . . 334
58.4 Regular page footnote implementation . . . 334
58.5 Minipage footnote implementation . . . 336
58.6 Printing pending footnotes . . . 338

59 Marginpars 338

60 Splitting HTML files 340
60.1 Customizing MathJax . . . 344

61 Sectioning 349
61.1 User-level starred section commands . . . 350
61.2 Book class commands . . . 350
61.3 Sectioning support macros . . . 351
61.4 Pre- and post- sectioning names . . . 358
61.5 `\section` and friends . . . 358

62 Starting a new file 360

63 Starting HTML output 364

64 Ending HTML output 367

65 Title page 368
65.1 Setting the title, etc. . . . 369
65.2 `\if@titlepage` . . . 370
65.3 Changes for `\affiliation` . . . 370
65.4 Printing the thanks . . . 371
65.5 Printing the title, etc. in HTML . . . 372
65.6 Printing the title, etc. in print form . . . 373
65.7 `\maketitle` for HTML output . . . 373
65.8 `\published` and `\subtitle` . . . 376

66 Abstract 377

67 Quote and verse 378
67.1 Attributions . . . 378
67.2 Quotes, quotations . . . 378
67.3 Verse . . . 379
67.3.1 LaTeX core verse environment . . . 379
67.3.2 verse and memoir . . . 380

68 Verbatim and tabbing 380

69 Theorems 383

70 Lists 384
70.1 List environment . . . 384
70.2 Itemize . . . 388
70.3 Enumerate . . . 388
70.4 Description . . . 389
70.5 Patching the lists . . . 389

71 Tabular 391
71.1 Limitations . . . 391
71.2 Temporary package-related macros . . . 393
71.2.1 arydshln . . . 393
71.3 Token lookahead . . . 394
71.4 Tabular variables . . . 394
71.5 Handling &, @, !, and bar . . . 397
71.5.1 Handling & . . . 399
71.5.2 Filling an unfinished row . . . 400
71.6 Handling \\ . . . 401
71.7 Looking ahead in the column specifications . . . 402
71.8 Parsing @, >, <, !, bar columns . . . 403
71.9 Parsing 'l', 'c', or 'r' columns . . . 406
71.10 Parsing 'p', 'm', or 'b' columns . . . 407
71.11 Parsing 'w' columns . . . 407
71.12 Parsing 'D' columns . . . 407
71.13 Parsing the column specifications . . . 408
71.14 colortbl and xparse tabular color support . . . 413
71.15 Starting a new row . . . 414
71.16 Printing vertical bar tags . . . 415
71.17 Printing at or bang tags . . . 415
71.18 Data opening tag . . . 416
71.19 Midrules . . . 418
71.20 Cell colors . . . 424
71.21 Multicolumns . . . 427
71.21.1 Parsing multicolumns . . . 427
71.21.2 Multicolumn factored code . . . 431
71.21.3 Multicolumn . . . 434
71.21.4 Longtable captions . . . 434
71.21.5 Counting HTML tabular columns . . . 437
71.22 Multirow if not loaded . . . 438
71.23 Multicolumnrow . . . 438

71.24 Utility macros inside a table . . . 440
71.25 Special-case tabular markers . . . 440
71.26 Checking for a new table cell . . . 441
71.27 \mrowcell . . . 444
71.28 \mcolrowcell . . . 444
71.29 HTML tabular environment . . . 444

72 Cross-references 450
72.1 Setup . . . 452
72.2 New lwarp labels. . . . 452
72.3 Labels . . . 453
72.4 References . . . 455
72.5 Hyper-references . . . 457

73 Floats 461
73.1 Float environment . . . 461
73.2 Float tracking . . . 463
73.3 Caption inside a float environment . . . 465
73.4 Caption and LOF linking and tracking . . . 465

74 Table of Contents, LOF, LOT 468
74.1 Reading and printing the TOC . . . 469
74.2 High-level TOC commands . . . 471
74.3 Side TOC . . . 472
74.4 Low-level TOC line formatting . . . 473

75 Index and glossary 476

76 Bibliography presentation 479

77 Restoring original formatting 480

78 Math 482
78.1 Limitations . . . 482
78.1.1 Math in section names . . . 482
78.1.2 Rendering tradeoffs . . . 482
78.1.3 SVG option . . . 483
78.1.4 MathJax option . . . 483
78.1.5 Customizing MathJax . . . 484
78.1.6 MathJax limitations . . . 484
78.1.7 Catcode changes . . . 485
78.1.8 Complicated inline math objects . . . 485
78.1.9 Complicated display math objects . . . 485
78.2 HTML alt tag names . . . 486
78.3 Inline and display math . . . 486
78.4 MathJax support . . . 497
78.5 Equation environment . . . 499
78.6 \displaymathnormal and \displaymathother . . . 502
78.7 AMS Math environments . . . 503
78.7.1 Support macros . . . 503
78.7.2 Environment patches . . . 504

79 Lateximages 507
79.1 Description . . . 507
79.2 Support counters and macros . . . 508
79.3 Font size . . . 509
79.4 Sanitizing math expressions for HTML . . . 509
79.5 Equation numbers . . . 511
79.6 HTML alt tags . . . 512
79.7 lateximage environment . . . 512

80 center, flushleft, flushright 518

81 Preloaded packages 519

82 siunitx 521

83 Graphics print-mode modifications 522
83.1 General limitations . . . 522
83.2 Print-mode modifications . . . 523

84 xcolor boxes 524

85 chemmacros environments 527

86 cleveref 527

87 picture environment 530

88 Boxes and Minipages 531
88.1 Counters and lengths . . . 532
88.2 Footnote handling . . . 532
88.3 Minipage handling . . . 532
88.4 \parbox, \mbox, \makebox, \framebox, \fbox, \raisebox . . . 536

89 Direct formatting 541

90 Skips, spaces, font sizes 552

91 \phantomsection 559

92 \LaTeX and other logos 559
92.1 HTML logos . . . 560
92.2 Print logos . . . 562

93 \AtBeginDocument, \AtEndDocument 563

94 Loading KOMA-SCRIPT class patches 563

95 Loading MEMOIR class patches 563

96 ut* class patches 564

97 CTEX patches 565

98 kotexutf patches 566

2 lwarp-2in1.sty 567

99 2in1 567

3 lwarp-2up.sty 567

100 2up 567

4 lwarp-a4.sty 567

101 a4 567

5 lwarp-a4wide.sty 568

102 a4wide 568

6 lwarp-a5comb.sty 568

103 a5comb 568

7 lwarp-abstract.sty 568

104 abstract 568

8 lwarp-academicons.sty 570

105 academicons 570

9 lwarp-accsupp.sty 571

106 accsupp 571

10 lwarp-acro.sty 571

107 acro 571

11 lwarp-acronym.sty 573

108 acronym 573

12 lwarp-adjmulticol.sty 574

109 adjmulticol 574

13 lwarp-addlines.sty 575
110 addlines 575

14 lwarp-afterpage.sty 575
111 afterpage 575

15 lwarp-algorithm2e.sty 575
112 algorithm2e 575

16 lwarp-algorithmicx.sty 579
113 algorithmicx 579

17 lwarp-alltt.sty 580
114 alltt 580

18 lwarp-amsmath.sty 580
115 amsmath 580

19 lwarp-amsthm.sty 587
116 amsthm 587

20 lwarp-anonchap.sty 590
117 anonchap 590

21 lwarp-anysize.sty 591
118 anysize 591

22 lwarp-appendix.sty 591
119 appendix 591

23 lwarp-ar.sty 592
120 ar 592

24 lwarp-arabicfront.sty 593

121 arabicfront 593

25 lwarp-array.sty 593

122 array 593

26 lwarp-arydshln.sty 594

123 arydshln 594

27 lwarp-asymptote.sty 596

124 asymptote 596

28 lwarp-atbegshi.sty 597

125 atbegshi 597

29 lwarp-attachfile.sty 597

126 attachfile 597

30 lwarp-attachfile2.sty 598

127 attachfile2 598

31 lwarp-authblk.sty 600

128 authblk 600

32 lwarp-axessibility.sty 601

129 axessibility 601

33 lwarp-axodraw2.sty 601

130 axodraw2 601

34 lwarp-backref.sty 602

131 backref 602

35 lwarp-balance.sty 602

132 balance 602

36 lwarp-bbding.sty 602

133 bbding 602

37 lwarp-biblatex.sty 607

134 biblatex 607

38 lwarp-bibunits.sty 608

135 bibunits 608

39 lwarp-bigdelim.sty 608

136 bigdelim 608

40 lwarp-bigstrut.sty 609

137 bigstrut 609

41 lwarp-bitpattern.sty 609

138 bitpattern 609

42 lwarp-blowup.sty 610

139 blowup 610

43 lwarp-booklet.sty 610

140 booklet 610

44 lwarp-bookmark.sty 611

141 bookmark 611

45 lwarp-booktabs.sty 611

142 booktabs 611

46 lwarp-bophook.sty 613

143 bophook 613

47 lwarp-bounddvi.sty 613

144 bounddvi 613

48 lwarp-boxedminipage2e.sty 613

145 boxedminipage2e 613

49 lwarp-breakurl.sty 614

146 breakurl 614

50 lwarp-breqn.sty 614

147 breqn 614

51 lwarp-bsheaders.sty 616

148 bsheaders 616

52 lwarp-bxpapersize.sty 616

149 bxpapersize 616

53 lwarp-bytefield.sty 616

150 bytefield 616

54 lwarp-cancel.sty 617

151 cancel 617

55 lwarp-canoniclayout.sty 618

152 canoniclayout 618

56 lwarp-caption.sty 618

153 caption 618

57 lwarp-cases.sty 621

154 cases 621

58 lwarp-changebar.sty 621

155 changebar 621

59 lwarp-changepage.sty 622

156 changepage 622

60 lwarp-changes.sty 622

157 changes 622

61 lwarp-chappg.sty 623

158 chappg 623

62 lwarp-chapterbib.sty 624

159 chapterbib 624

63 lwarp-chemfig.sty 624

160 chemfig 624

64 lwarp-chemformula.sty 625

161 chemformula 625

65 lwarp-chemgreek.sty 630

162 chemgreek 630

66 lwarp-chemmacros.sty 630

163 chemmacros 630
163.1 Changes to the user's document 631
163.2 Code 631
163.3 Loading modules 631
163.4 New environments 631
163.5 Acid-base 632

163.6 Charges . . . 634
163.7 Nomenclature . . . 634
163.8 Particles . . . 636
163.9 Phases . . . 637
163.10 Mechanisms . . . 637
163.11 Newman . . . 639
163.12 Orbital . . . 640
163.13 Reactions . . . 640
163.14 Redox . . . 641
163.15 Scheme . . . 642
163.16 Spectroscopy . . . 642
163.17 Thermodynamics . . . 646

67 lwarp-chemnum.sty 648

164 chemnum 648

68 lwarp-chkfloat.sty 649

165 chkfloat 649

69 lwarp-chngpage.sty 649

166 chngpage 649

70 lwarp-cite.sty 649

167 cite 649

71 lwarp-CJK.sty 650

168 CJK 650

72 lwarp-CJKutf8.sty 650

169 CJKutf8 650

73 lwarp-clrdblpg.sty 650

170 clrdblpg 650

74 lwarp-cmdtrack.sty 651

171 cmdtrack 651

75 lwarp-color.sty 651

172 color 651

76 lwarp-colortbl.sty 651

173 colortbl 651

77 lwarp-continue.sty 653

174 continue 653

78 lwarp-copyrightbox.sty 653

175 copyrightbox 653

79 lwarp-crop.sty 654

176 crop 654

80 lwarp-cuted.sty 654

177 cuted 654

81 lwarp-cutwin.sty 654

178 cutwin 654

82 lwarp-dblfloatfix.sty 655

179 dblfloatfix 655

83 lwarp-dblfnote.sty 655

180 dblfnote 655

84 lwarp-dcolumn.sty 656

181 dcolumn 656

85 lwarp-diagbox.sty 656

182 diagbox 656

86 lwarp-dingbat.sty 658

183 dingbat 658

87 lwarp-dprogress.sty 659

184 dprogress 659

88 lwarp-draftcopy.sty 659

185 draftcopy 659

89 lwarp-draftfigure.sty 659

186 draftfigure 659

90 lwarp-draftwatermark.sty 660

187 draftwatermark 660

91 lwarp-easy-todo.sty 660

188 easy-todo 660

92 lwarp-ebook.sty 661

189 ebook 661

93 lwarp-ed.sty 661

190 ed 661

94 lwarp-ellipsis.sty 662

191 ellipsis 662

95 lwarp-embrac.sty 662

192 embrac 662

96 lwarp-emptypage.sty 663

193 emptypage 663

97 lwarp-endfloat.sty 663

194 endfloat 663

98 lwarp-endheads.sty 664

195 endheads 664

99 lwarp-endnotes.sty 664

196 endnotes 664

100 lwarp-enumerate.sty 665

197 enumerate 665

101 lwarp-enumitem.sty 665

198 enumitem 665

102 lwarp-epigraph.sty 666

199 epigraph 666

103 lwarp-epsfig.sty 667

200 epsfig 667

104 lwarp-epstopdf.sty 668

201 epstopdf 668

105 lwarp-epstopdf-base.sty 668

202 epstopdf-base 668

106 lwarp-errata.sty 669

203 errata 669

107 lwarp-eso-pic.sty 670

204 eso-pic 670

108 lwarp-eurosym.sty 671

205 eurosym 671

109 lwarp-everypage.sty 671

206 everypage 671

110 lwarp-everyshi.sty 672

207 everyshi 672

111 lwarp-extramarks.sty 672

208 extramarks 672

112 lwarp-fancybox.sty 673

209 fancybox 673

113 lwarp-fancyhdr.sty 678

210 fancyhdr 678

114 lwarp-fancyref.sty 679

211 fancyref 679

115 lwarp-fancytabs.sty 680

212 fancytabs 680

116 lwarp-fancyvrb.sty 680

213 fancyvrb 680

117 lwarp-figcaps.sty 686

214 figcaps 686

118 lwarp-figsize.sty 686

215 figsize 686

119 lwarp-fitbox.sty 687

216 fitbox 687

120 lwarp-fix2col.sty 687

217 fix2col 687

121 lwarp-fixme.sty 687

218 fixme 687

122 lwarp-fixmetodonotes.sty 689

219 fixmetodonotes 689

123 lwarp-flafter.sty 689

220 flafter 689

124 lwarp-flippdf.sty 689

221 flippdf 689

125 lwarp-float.sty 690

222 float 690

126 lwarp-floatflt.sty 691

223 floatflt 691

127 lwarp-floatpag.sty 692

224 floatpag 692

128 lwarp-floatrow.sty 692

225 floatrow 692

129 lwarp-fltrace.sty 697

226 fltrace 697

130 lwarp-flushend.sty 698

227 flushend 698

131 lwarp-fnbreak.sty 698

228 fnbreak 698

132 lwarp-fncychap.sty 698

229 fncychap 698

133 lwarp-fnlineno.sty 699

230 fnlineno 699

134 lwarp-fnpos.sty 699

231 fnpos 699

135 lwarp-fontawesome.sty 700

232 fontawesome 700

136 lwarp-fontawesome5.sty 701

233 fontawesome5 701

137 lwarp-fontenc.sty 702

234 fontenc 702

138 lwarp-footmisc.sty 702

235 footmisc 702

139 lwarp-footnote.sty 703

236 footnote 703

140 lwarp-footnotehyper.sty 704

237 footnotehyper 704

141 lwarp-footnoterange.sty 704

238 footnoterange 704

142 lwarp-footnpag.sty 705

239 footnpag 705

143 lwarp-foreign.sty 705

240 foreign 705

144 lwarp-forest.sty 705

241 forest 705

145 lwarp-framed.sty 706

242 framed 706

146 lwarp-ftnright.sty 708

243 ftnright 708

147 lwarp-fullminipage.sty 708

244 fullminipage 708

148 lwarp-fullpage.sty 708

245 fullpage 708

149 lwarp-fullwidth.sty 709

246 fullwidth 709

150 lwarp-fwlw.sty 709

247 fwlw 709

151 lwarp-gentombow.sty 709

248 gentombow 709

152 lwarp-geometry.sty 710

249 geometry 710

153 lwarp-gloss.sty 710

250 gloss 710

154 lwarp-glossaries.sty 711

251 glossaries 711

155 lwarp-gmeometric.sty 713

252 gmeometric 713

156 lwarp-graphics.sty 713

253 graphics 713
253.1 Graphics extensions 713
253.2 Length conversions and graphics options 714
253.3 Printing HTML styles 717
253.4 \includegraphics 717
253.5 Boxes 722

157 lwarp-graphicx.sty 725

254 graphicx 725

158 lwarp-grffile.sty 726

255 grffile 726

159 lwarp-grid.sty 726

256 grid 726

160 lwarp-grid-system.sty 726

257 grid-system 726

161 lwarp-gridset.sty 727

258 gridset 727

162 lwarp-hang.sty 727

259 hang 727

163 lwarp-hanging.sty 728

260 hanging 728

164 lwarp-hypcap.sty 729

261 hypcap 729

165 lwarp-hypdestopt.sty 730

262 hypdestopt 730

166 lwarp-hypernat.sty 730

263 hypernat 730

167 lwarp-hyperref.sty 730

264 hyperref 730

168 lwarp-hyperxmp.sty 735

265 hyperxmp 735

169 lwarp-hyphenat.sty 735

266 hyphenat 735

170 lwarp-idxlayout.sty 736

267 idxlayout 736

171 lwarp-ifoddpage.sty 737

268 ifoddpage 737

172 lwarp-imakeidx.sty 737

269 imakeidx 737

173 lwarp-index.sty 742

270 index 742

174 lwarp-intopdf.sty 743

271 intopdf 743

175 lwarp-karnaugh-map.sty 743

272 karnaugh-map 743

176 lwarp-keyfloat.sty 746

273 keyfloat 746

177 lwarp-layaureo.sty 748

274 layaureo 748

178 lwarp-layout.sty 749

275 layout 749

179 lwarp-leading.sty 749

276 leading 749

180 lwarp-letterspace.sty 749

277 letterspace 749

181 lwarp-lettrine.sty 750

278 lettrine 750

182 lwarp-lineno.sty 750

279 lineno 750

183 lwarp-lips.sty 753

280 lips 753

184 lwarp-listings.sty 753

281 listings 753

185 lwarp-lltjext.sty 757

282 lltjext 757

186 lwarp-longtable.sty 758

283 longtable 758

187 lwarp-lscape.sty 760

284 lscape 760

188 lwarp-ltablex.sty 760

285 ltablex 760

189 lwarp-ltcaption.sty 761

286 ltcaption 761

190 lwarp-ltxgrid.sty 761

287 ltxgrid 761

191 lwarp-ltxtable.sty 761

288 ltxtable 761

192 lwarp-lua-check-hyphen.sty 762

289 lua-check-hyphen 762

193 lwarp-lua-visual-debug.sty 762

290 lua-visual-debug 762

194 lwarp-luacolor.sty 762

291 luacolor 762

195 lwarp-luatodonotes.sty 762

292 luatodonotes 762

196 lwarp-magaz.sty 764

293 magaz 764

197 lwarp-makeidx.sty 765

294 makeidx 765

198 lwarp-marginal.sty 765

295 marginal 765

199 lwarp-marginfit.sty 765

296 marginfit 765

200 lwarp-marginfix.sty 766

297 marginfix 766

201 lwarp-marginnote.sty 766

298 marginnote 766

202 lwarp-marvosym.sty 767

299 marvosym 767

203 lwarp-mcaption.sty 767

300 mcaption 767

204 lwarp-mdframed.sty 767

301 mdframed 767

301.1 Limitations 768
301.2 Package loading 768
301.3 Patches 768
301.4 Initial setup 769
301.5 Color and length HTML conversion 769

301.6 Environment encapsulation 770
301.7 Mdframed environment 771
301.8 Titles and subtitles 772
301.9 New environments 774

205 lwarp-memhfixc.sty 776

302 memhfixc 776

206 lwarp-metalogo.sty 777

303 metalogo 777

207 lwarp-mhchem.sty 777

304 mhchem 777

208 lwarp-microtype.sty 779

305 microtype 779

209 lwarp-midfloat.sty 780

306 midfloat 780

210 lwarp-midpage.sty 780

307 midpage 780

211 lwarp-morefloats.sty 781

308 morefloats 781

212 lwarp-moreverb.sty 781

309 moreverb 781

213 lwarp-mparhack.sty 782

310 mparhack 782

214 lwarp-multicap.sty 782

311 multicap 782

215 lwarp-multicol.sty 783

312 multicol 783

216 lwarp-multicolrule.sty 784

313 multicolrule 784

217 lwarp-multirow.sty 784

314 multirow 784
314.1 Multirow 785
314.2 Combined multicolumn and multirow 786

218 lwarp-multitoc.sty 787

315 multitoc 787

219 lwarp-musicography.sty 787

316 musicography 787

220 lwarp-nameauth.sty 789

317 nameauth 789

221 lwarp-nameref.sty 791

318 nameref 791

222 lwarp-natbib.sty 791

319 natbib 791

223 lwarp-nccfancyhdr.sty 792

320 nccfancyhdr 792

224 lwarp-needspace.sty 793

321 needspace 793

225 lwarp-nextpage.sty 793

322 nextpage 793

226 lwarp-nicefrac.sty 793

323 nicefrac 793

227 lwarp-nomencl.sty 794

324 nomencl 794

228 lwarp-nonfloat.sty 795

325 nonfloat 795

229 lwarp-nonumonpart.sty 795

326 nonumonpart 795

230 lwarp-nopageno.sty 795

327 nopageno 795

231 lwarp-notes.sty 795

328 notes 795

232 lwarp-notespages.sty 796

329 notespages 796

233 lwarp-nowidow.sty 796

330 nowidow 796

234 lwarp-ntheorem.sty 797

331 ntheorem 797

331.1 Limitations 797
331.2 Options 797
331.3 Remembering the theorem style 798
331.4 HTML cross-referencing 801
331.5 \newtheoremstyle 801

331.6 Standard styles . . . 802
331.7 Additional objects . . . 803
331.8 Renewed standard configuration . . . 803
331.9 amsthm option . . . 804
331.10 Ending a theorem . . . 807
331.11 \NoEndMark . . . 807
331.12 List-of . . . 807
331.13 Symbols . . . 808
331.14 Cross-referencing . . . 808

235 lwarp-octave.sty 808

332 octave 808

236 lwarp-overpic.sty 809

333 overpic 809

237 lwarp-pagegrid.sty 810

334 pagegrid 810

238 lwarp-pagenote.sty 810

335 pagenote 810

239 lwarp-pagesel.sty 811

336 pagesel 811

240 lwarp-paralist.sty 811

337 paralist 811

241 lwarp-parnotes.sty 811

338 parnotes 811

242 lwarp-parskip.sty 812

339 parskip 812

243 lwarp-pbox.sty 813

340 pbox 813

244 lwarp-pdfcomment.sty 813

341 pdfcomment 813

245 lwarp-pdflscape.sty 814

342 pdflscape 814

246 lwarp-pdfmarginpar.sty 814

343 pdfmarginpar 814

247 lwarp-pdfpages.sty 814

344 pdfpages 814

248 lwarp-pdfprivacy.sty 817

345 pdfprivacy 817

249 lwarp-pdfrender.sty 817

346 pdfrender 817

250 lwarp-pdfsync.sty 817

347 pdfsync 817

251 lwarp-pdftricks.sty 817

348 pdftricks 817

252 lwarp-pdfx.sty 818

349 pdfx 818

253 lwarp-pfnote.sty 818

350 pfnote 818

254 lwarp-phfqit.sty 819

351 phfqit 819

255 lwarp-pifont.sty 819

352 pifont 819

256 lwarp-placeins.sty 820

353 placeins 820

257 lwarp-plarydshln.sty 820

354 plarydshln 820

258 lwarp-plext.sty 820

355 plext 820

259 lwarp-plextarydshln.sty 821

356 plextarydshln 821

260 lwarp-plextcolortbl.sty 821

357 plextcolortbl 821

261 lwarp-prelim2e.sty 822

358 prelim2e 822

262 lwarp-prettyref.sty 822

359 prettyref 822

263 lwarp-preview.sty 822

360 preview 822

264 lwarp-psfrag.sty 823

361 psfrag 823

265 lwarp-psfragx.sty 823

362 psfragx 823

266 lwarp-pst-eps.sty 824

363 pst-eps 824

267 lwarp-pstool.sty 824

364 pstool 824

268 lwarp-pstricks.sty 825

365 pstricks 825

269 lwarp-pxatbegshi.sty 825

366 pxatbegshi 825

270 lwarp-pxeveryshi.sty 825

367 pxeveryshi 825

271 lwarp-pxftnright.sty 826

368 pxftnright 826

272 lwarp-pxjahyper.sty 826

369 pxjahyper 826

273 lwarp-quotchap.sty 826

370 quotchap 826

274 lwarp-quoting.sty 827

371 quoting 827

275 lwarp-ragged2e.sty 828

372 ragged2e 828

276 lwarp-realscripts.sty 828

373 realscripts 828

277 lwarp-refcheck.sty 829

374 refcheck 829

278 lwarp-register.sty 830

375 register 830

279 lwarp-relsize.sty 830

376 relsize 830

280 lwarp-repeatindex.sty 832

377 repeatindex 832

281 lwarp-resizegather.sty 833

378 resizegather 833

282 lwarp-rmpage.sty 833

379 rmpage 833

283 lwarp-romanbar.sty 833

380 romanbar 833

284 lwarp-romanbarpagenumber.sty 833

381 romanbarpagenumber 833

285 lwarp-rotating.sty 834

382 rotating 834

286 lwarp-rotfloat.sty 834

383 rotfloat 834

287 lwarp-rviewport.sty 835

384 rviewport 835

288 lwarp-savetrees.sty 835

385 savetrees 835

289 lwarp-scalefnt.sty 836

386 scalefnt 836

290 lwarp-schemata.sty 836

387 schemata 836

291 lwarp-scrextend.sty 836

388 scrextend 836

292 lwarp-scrhack.sty 839

389 scrhack 839

293 lwarp-scrlayer.sty 839

390 scrlayer 839

294 lwarp-scrlayer-notecolumn.sty 841

391 scrlayer-notecolumn 841

295 lwarp-scrlayer-scrpage.sty 841

392 scrlayer-scrpage 841

296 lwarp-scrpage2.sty 842

393 scrpage2 842

297 lwarp-section.sty 843

394 section 843

298 lwarp-sectionbreak.sty 844

395 sectionbreak 844

299 lwarp-sectsty.sty 845

396 sectsty 845

300 lwarp-semantic-markup.sty 845

397 semantic-markup 845

301 lwarp-setspace.sty 846

398 setspace 846

302 lwarp-shadow.sty 847

399 shadow 847

303 lwarp-showidx.sty 848

400 showidx 848

304 lwarp-showkeys.sty 848

401 showkeys 848

305 lwarp-sidecap.sty 848

402 sidecap 848

306 lwarp-sidenotes.sty 849

403 sidenotes 849

307 lwarp-SIunits.sty 850

404 SIunits 850

308 lwarp-siunitx.sty 851

405 siunitx 851

309 lwarp-soul.sty 856

406 soul 856

310 lwarp-soulpos.sty 858

407 soulpos 858

311 lwarp-soulutf8.sty 858

408 soulutf8 858

312 lwarp-splitidx.sty 858

409 splitidx 858

313 lwarp-srcltx.sty 860

410 srcltx 860

314 lwarp-srctex.sty 860

411 srctex 860

315 lwarp-stabular.sty 861

412 stabular 861

316 lwarp-stfloats.sty 861

413 stfloats 861

317 lwarp-struktex.sty 862

414 struktex 862

318 lwarp-subcaption.sty 862

415 subcaption 862

319 lwarp-subfig.sty 863

416 subfig 863

320 lwarp-subfigure.sty 867

417 subfigure 867

321 lwarp-supertabular.sty 868

418 supertabular 868

322 lwarp-syntonly.sty 869

419 syntonly 869

323 lwarp-tabls.sty 870

420 tabls 870

324 lwarp-tabularx.sty 870

421 tabularx 870

325 lwarp-tabulary.sty 871

422 tabulary 871

326 lwarp-tascmac.sty 871

423 tascmac 871

327 lwarp-textarea.sty 873

424 textarea 873

328 lwarp-textcomp.sty 873

425 textcomp 873
425.1 Limitations . . . 873
425.2 Package loading . . . 873
425.3 HTML symbols . . . 874
425.3.1 pdfLaTeX symbols . . . 874
425.3.2 XeLaTeX and LuaLaTeX symbols . . . 874
425.4 HTML diacritics . . . 875

329 lwarp-textfit.sty 877

426 textfit **877**

330 lwarp-textpos.sty 877

427 textpos **877**

331 lwarp-theorem.sty 878

428 theorem **878**
428.1 Remembering the theorem style 878
428.2 css patches 880

332 lwarp-thinsp.sty 882

429 thinsp **882**

333 lwarp-threadcol.sty 882

430 threadcol **882**

334 lwarp-threeparttable.sty 882

431 threeparttable **882**

335 lwarp-thumb.sty 883

432 thumb **883**

336 lwarp-thumbs.sty 883

433 thumbs **883**

337 lwarp-tikz.sty 884

434 tikz **884**

338 lwarp-titleps.sty 885

435 titleps **885**

339 lwarp-titleref.sty 888

436 titleref **888**

340 lwarp-titlesec.sty 888

437 titlesec 888

341 lwarp-titletoc.sty 890

438 titletoc 890

342 lwarp-titling.sty 892

439 titling 892

343 lwarp-tocbasic.sty 896

440 tocbasic 896

344 lwarp-tocbibind.sty 897

441 tocbibind 897

345 lwarp-tocenter.sty 898

442 tocenter 898

346 lwarp-tocloft.sty 899

443 tocloft 899

347 lwarp-tocstyle.sty 905

444 tocstyle 905

348 lwarp-todo.sty 905

445 todo 905

349 lwarp-todonotes.sty 906

446 todonotes 906

350 lwarp-tram.sty 908

447 tram 908

351 lwarp-transparent.sty 908

448 transparent 908

352 lwarp-trimclip.sty 909

449 trimclip 909

353 lwarp-trivfloat.sty 909

450 trivfloat 909
450.1 Combining \newfloat, \trivfloat, and algorithmicx 910

354 lwarp-truncate.sty 910

451 truncate 910

355 lwarp-turnthepage.sty 911

452 turnthepage 911

356 lwarp-twoup.sty 911

453 twoup 911

357 lwarp-typearea.sty 911

454 typearea 911

358 lwarp-typicons.sty 912

455 typicons 912

359 lwarp-ulem.sty 912

456 ulem 912

360 lwarp-umoline.sty 914

457 umoline 914

361 lwarp-underscore.sty 915

458 underscore 915

362 lwarp-units.sty 915

459 units 915

363 lwarp-unitsdef.sty 916

460 unitsdef 916

364 lwarp-upref.sty 916

461 upref 916

365 lwarp-url.sty 917

462 url 917

366 lwarp-uspace.sty 917

463 uspace 917

367 lwarp-verse.sty 917

464 verse 917

368 lwarp-versonotes.sty 919

465 versonotes 919

369 lwarp-vertbars.sty 919

466 vertbars 919

370 lwarp-vmargin.sty 920

467 vmargin 920

371 lwarp-vowel.sty 920

468 vowel 920

372 lwarp-vpe.sty 921

469 vpe **921**

373 lwarp-vwcol.sty 921

470 vwcol **921**

374 lwarp-wallpaper.sty 923

471 wallpaper **923**

375 lwarp-watermark.sty 924

472 watermark **924**

376 lwarp-widows-and-orphans.sty 924

473 widows-and-orphans **924**

377 lwarp-wrapfig.sty 924

474 wrapfig **924**

378 lwarp-xbmks.sty 925

475 xbmks **925**

379 lwarp-xcolor.sty 926

476 xcolor **926**
476.1 Limitations . . . 926
476.2 xcolor definitions: location and timing . . . 926
476.3 Package loading . . . 928
476.4 Remembering and restoring original definitions . . . 928
476.5 HTML color style . . . 929
476.6 HTML border . . . 930
476.7 High-level macros . . . 930
476.8 Row colors . . . 935

380 lwarp-xechangebar.sty 936

477 xechangebar **936**

381 lwarp-xellipsis.sty 936

478 xellipsis 936

382 lwarp-xetexko-vertical.sty 937

479 xetexko-vertical 937

383 lwarp-xfakebold.sty 937

480 xfakebold 937

384 lwarp-xfrac.sty 938

481 xfrac 938

385 lwarp-xltabular.sty 940

482 xltabular 940

386 lwarp-xltxtra.sty 941

483 xltxtra 941

387 lwarp-xmpincl.sty 941

484 xmpincl 941

388 lwarp-xpiano.sty 942

485 xpiano 942

389 lwarp-xpinyin.sty 942

486 xpinyin 942

390 lwarp-xtab.sty 943

487 xtab 943

391 lwarp-xunicode.sty 944

488 xunicode 944

392 lwarp-xurl.sty 945

489 xurl 945

393 lwarp-xy.sty 946

490 xy 946

394 lwarp-zhlineskip.sty 946

491 zhlineskip 946

395 lwarp-zwpagelayout.sty 947

492 zwpagelayout 947

396 lwarp-patch-komascript.sty 948

493 patch-komascript 948

397 lwarp-patch-memoir.sty 950

494 patch-memoir 950

494.1 Packages . . . 951
494.2 Preliminary setup . . . 952
494.3 Laying out the page . . . 952
494.4 Text and fonts . . . 954
494.5 Titles . . . 955
494.6 Abstracts . . . 955
494.7 Docment divisions . . . 955
494.8 Pagination and headers . . . 958
494.9 Paragraphs and lists . . . 959
494.10 Contents lists . . . 960
494.11 Floats and captions . . . 964
494.12 Page notes . . . 967
494.13 Decorative text . . . 968
494.14 Poetry . . . 969
494.15 Boxes, verbatims and files . . . 969
494.16 Cross referencing . . . 970
494.17 Back matter . . . 971
494.18 Miscellaneous . . . 972
494.19 ccaption emulation . . . 972
494.20 Final patchwork . . . 974

Change History 975

495 Chg Hist **975**

Index of Objects **1001**

General Index **1020**

Troubleshooting Index **1024**

Index of Indexes **1029**

List of Figures

1 tutorial.tex listing 91

List of Tables

1 Typesetting conventions 74
2 LaTeX lwarp package — Supported features 75
3 Required software programs 84
4 Configuration files created by print version 93
5 lwarp package options 108
6 HTML settings 112
7 Literal character macros 167
8 Section HTML headings for word-processor conversion 175
9 Section depths and HTML headings 189
10 Tabular baseline 408
11 Tabular HTML column conversions 409
12 Cross-referencing data structures 451
13 Float data structures 461
14 amsthm package — CSS styling of theorems and proofs 587
15 Ntheorem package — CSS styling of theorems and proofs 797
16 Theorem package — CSS styling of theorems and proofs 878

2 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 1000.

v0.67: Filename generation, symbol fonts.

docs
- Documentation fix for `<project>-images`, `<project>-images.txt`.
- Added discussion regarding section names. See section 9.4.

filenames
- Added `\FilenameNullify` and `\FilenameSimplify` for filename generation. See section 9.4.
- 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 9.3.3.

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

HTML
- Improved HTML output formatting.

docs
- Added discussion regarding invalid HTML. See section 9.1.1.
- Added discussion regarding math in section names, \imagegraphics scale option. See section 7.
- Added discussion regarding international languages in section names. See section 9.14.

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 9.13.
- 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 10.8.

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 8.4.

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. See page 1029.
- Improved documentation regarding font selection. See section 8.2.
- Added documentation regarding debugging options. See section 35.
- Added documentation regarding HTML entities inside program listings. See section 9.2.1.

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 10.

⚠ 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 9.8.

DVI *latex*
- Added support for DVI *latex*. See section 8.3.

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, Slunits.
- 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 9.6.13.
- 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 8.3.
- Added lwarp options `makeindex` and `xindy` to set `PrintIndexCmd`, `HTMLIndexCmd`, and `LatexmkIndexCmd` to sensible values for a typical single index. See section 8.3.
- Added lwarp option `makeindexStyle` to tell *lwarpmk* to use a custom style instead of `lwarp.ist`. See section 9.6.19.
- 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.

packages

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

MathJax

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

math macros

- Added `\StartDefiningMath` and `\StopDefiningMath` for use when defining macros in the preamble which contain $. See section 9.7.7.

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 9.7.8.

⚠ 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 \mathimagename, which defaults to "math image". For packages, the alt tag is set using the package name followed by \packagediagramname, which defaults to "diagram". Ex:

  ```
  (-xy- diagram)
  ```

  See section 8.4.

- 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 9.8.

- Added support for --shell-escape. See section 8.1.

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.
- Tested to work as-is with babelbib, bibunits, 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 9.6.9.

polyglossia
- Added documentation regarding polyglossia. See section 9.15.4.

macros in section names
- Added documentation regarding the use of macros in section names. See section 9.1.

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 8.2.

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 9.5.4.
- Added documentation regarding font selection when using XeLaTeX or LuaLaTeX with fontspec and traditional font packages. See section 8.2.

SVG math
- Fix: Limit the number of background tasks when generating `lateximages`.
- Added user-adjustable SVG math font scaling. See section 79.3.
- 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 9.7.

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.

v0.51: Improved SVG math, added numerous chemistry packages.

documentation
- Docs: Added Things to avoid.
- Docs: Added to Converting an existing document.
- Docs: Multiple authors and affiliations with custom classes. See section 9.6.1.
- Docs: tikz with matrices. See section 9.8.1.

SVG math
- Improved SVG math baseline.
- Improved SVG math font and color.
- Faster SVG math rendering.
- Improved support for display math containing complicated math objects, such as tikz-cd. See section 9.7.9.
- Fix: `\addcontentsline` inside SVG math.
- Fix: SVG math containing an embedded `lateximage`.

MathJax
- MathJax now handles `\ensuremath` in expressions.

misc. fixes
- Fix: Added `alignat` environment.
- Fix: afterpackage no longer required, which conflicted with scrlfile.
- Fix: titling `\thanks` mark.
- Fix: fancybox improvements.
- Fix: tikz `\tikz` macro. (Previously only the `tikzpicture` environment worked.)
- Fix: tikz with optional argument.

packages
- Added mhchem, chemfig, chemformula, chemmacros, chemnum, chemgreek, epstopdf-base, grid, ltxgrid.

v0.50: Improved SVG math.

SVG math
- SVG math and other `lateximages` now are converted to SVG using parallel background tasks, utilizing all available CPU cores.
- Inline SVG math image file names now are MD5 hashes made from their source LaTeX code. Identical inline math expressions, such as multiple instance of x, now share a single image file. This reduces the number of images to store, transmit, process, and display. Each image file is only converted to SVG a single time, and reused if it already exists. Display math and other forms of SVG image such as `picture` and Ti*k*z still use individual image files which are recreated each time **`lwarpmk limages`** is run.
- Fixes: SVG math and/or `\underline` in a sectioning file name.
- Improved SVG display math and tags.
- Improved SVG math and siunitx `alt` tags.
- Improved siunitx units.
- Fix: `\ensuremath` with MathJax now creates a `lateximage`.
- Fix: `\centering`, etc. in SVG math, `lateximage`, Ti*k*z.

misc. fixes
- Fix: Made various macros robust, additionally fixing authblk.
- Fix: ntheorem if neither `standard` nor `amsthm` selected.
- Fix: listings: Improved column alignment.
- Fix: Load fontspec if necessary.

packages
- Added xy, epstopdf, diagbox, pbox, bytefield, axodraw2, phfqit, schemata, dblfloatfix, nonfloat, morefloats.

v0.49:

tabular
- Added xcolor `\rowcolors`.
- Fix: `\noalign` inside a `tabular`.

math
- Fix: `\eqref` in a caption.

misc fixes
- Fix: Incorrect PDF font size changes caused occasional HTML corruption.
- Fix: printlen changes are now grouped for HTML output.

packages
- Added vwcol, vertbars, hyphenat, lineno, fnlineno, figsize, hypdestopt, pagegrid, pdfrender, luacolor, resizegather.

v0.48:

documentation
- Added some documentation regarding converting an existing document. See section 7.

cleveref
- Updated compatibility for new cleveref v0.21.

tabular
- Fix: Ignores optional tabular column arguments.

minor updates
- Added `\leftline`, `\centerline`, `\rightline`.
- Lists have improved font control via `\makelabel`.
- Print-mode `lateximage` now boxed to the natural width of its multiline contents.
- `abstract` now allows an optional name, as required by some classes.

math
- Fix: Improved spacing, `\mbox`, and font sizes with SVG math, Ti*k*z.
- siunitx: Improved SVG math, fraction compatibility, color output.

misc. fixes
- Fix: LOF/LOT links.
- Fix: Virtual page size grouping caused excessive PDF page breaks.
- Fix: Parsing similar package names in a single `\usepackage`.
- Fix: Adapts to classes without `\part`.
- Fix: `\newline` in `\title` was causing `<br>` in window title.
- Fix: `\maketitle` with `\cr`, `\crcr`, `\noalign`, for IEEEtran class.
- Fix: xfrac neutralized `BlockClass` and others.
- Fix: todonotes and luatodonotes: Improved `\todototoc`.

packages
- Added colortbl, chapterbib, acro, acronym, hypernat, hypcap, stfloats, vmargin, fancyheadings.
- fancyref: Now directly supported.

v0.47:

math
- Improved SVG math baseline and sizing.
- Fixes: SVGmath in captions, subcaptions, `\nameref`.
- Fixes: Line wrap at hyphen in HTML output.

packages
- Added endheads, multitoc, sectionbreak, blowup, xurl.

v0.46:

⚠ name change
- `\PrintStack` changed to `\LWRPrintStack`.

misc. fixes
- Fix: Empty lines between `tabular` rows.
- Fix: Stack unnesting.
- Fix: SVG math and `lateximages` in numerous situations.
- Fix: Spaces in `\usepackage`.
- Fix: Now allows MathJax inside `verse`.

v0.45:

documentation
- Improved *MiKTeX* install instructions.
- Improved graphics and epstopdf instructions.
- Updates to the Introduction.

memoir
- Added memoir, memhfixc. See section 9.13.

cross-references
- Fix: Now allows underscores in labels.
- Fix: `\_` and `\<blank>` in section/file names.

math
- Fix: Now allows MathJax inside `tabbing`.

bibliography
- Fix: Bibliography `\em` names.
- Added cite, natbib, backref. (Also works as-is with biblatex.)

misc. fixes
- Fix: Empty lines between `tabular` rows.
- Fix: "Improper `\prevdepth`" with minipages, lists.
- Fix: Incorrect SVG math and `lateximages` with subfig.
- Fix: Latexmages from incorrect pages with Mathjax.
- Fix: Missing sidetoc if using listings.
- Fix: Added an array emulation package.

packages
- Added subfigure, prettyref, hanging, midpage, flafter, fltrace, changebar, endfloat, continue, fwlw, turnthepage, footnpag, pagesel, textfit, titleref.

v0.44:

koma-script
- Added koma-script classes (except scrlttr2, scrjura).
- Added scrextend, scrlayer, scrlayer-notecolumn, scrlayer-scrpage, scrhack, tocstyle, tocbasic.

HTML title and author
- Added `\HTMLTitle`. Fixed web page title if `\HTMLTitle` empty and no `\title` given and not using titling package.
- Fixed web page author if `\HTMLauthor` is empty and `\author` is not given.

encodings
- If using *pdflatex*, automatically loads T1 and UTF-8 encodings. (Additional fontenc encodings may be loaded after lwarp.)

lists
- Added `list` and `trivlist` environments, hang.

tabular
- Fix: `\multicolumn` alignment if formatting for a word processor.
- Added ltxtable.

math
- Fix: MathJax combined with `lateximages`.

- algorithmicx: Improved comment symbol and floating.

packages
- Completed todonotes and luatodonotes.
- Added todo, easy-todo, fixmetodonotes, fixme.
- Added soulutf8, soulpos, cancel.
- Added section, fancyref, ifoddpage.
- Added preview, atbegshi, watermark.
- Improved tocloft `\newlistof` and `\newlistentry`.

v0.43:

- Docs: Reorganized HTML customization, added an HTML settings table. See section 8.4.

footnotes
- Added `FootnoteDepth` to control the placement of pending footnotes before section breaks. By default, pending footnotes are printed before each `\subparagraph` or higher.

sectioning
- Fix: Expansion in section name.

tabular
- Fix: Ignore spaces in tabular column specification.
- Fix: Tabular rules at bottom or when finishing incomplete rows.
- Fix: `\multicolumn` at/bang/before/after specifications, trim, and vertical rules.
- Fix: supertabular and xtab column misalignment.

math
- Fix: `equation*`.
- Fix: SVG math in a section name.
- Fix: `\ref` and `\eqref` in SVG math.

packages
- Added todonotes and luatodonotes (but only `disabled`).
- Added breakurl.
- hyperref: Fix: Several macros were made robust, `\Gauge` added.

v0.42:

Support TEX!
- Added TEX development support page, Support TEX development.

word-processor conversion
- Improved assitance for word-processor conversions when boolean `FormatWP` is set `true`. See section 12.

⚠ name change
  - The boolean `FormatWordProcessor` has been renamed `FormatWP`.

⚠ name change
  - The boolean `HTMLMarkFloats` has been renamed `WPMarkFloats`.
  - New booleans control whether to place additional marks around minipages, at the table of contents, at the LOF and LOT, and whether to print math as LaTeX source for copy/paste into the *LibreOffice Writer TeXMaths* extension.
  - Improved formatting for numerous objects. See section 12.

tabbing
- Add: `tabbing` environment.

overpic
- Add: overpic package. See section 333.

math
- Fix: Text copy/paste of $\mathcal{AMS}$ math environment numbers and names.
- Improved `\ensuremath`.

- MathJax with siunitx: Updated script and documentation.

symbols
- textcomp: Improved `\interrobangdown`.
- realscripts: Fix for subscripts in a `lateximage`.

load order
- morewrites: Enforces loading before lwarp.

v0.41:

tabular
- Added tabular vertical rules, subject to some limitations. See the rules section of section 9.10.
- Improved booktabs: Width and trim are honored.

⚠ new syntax
- Added `\mcolrowcell` for empty cells inside a `\multicolumnrow`. Use `\mcolrowcell` instead of `\mrowcell` for two-dimensional cells created by `\multicolumnrow`. Continue to use `\mrowcell` for empty cells in a `\multirow`. See section 314.2 on page 786.
- Fix: Unfinished tabular rows are automatically filled.
- Fix for tabular column specifiers while using babel-french. (`\NoAutoSpacing` is activated then nullified inside the tabular, due to a conflict with the tabular column parsing code.)

v0.40:

graphics, graphicx
- graphics and graphicx have been moved from the lwarp core, and are only loaded if requested with `\usepackage`.

`\includegraphics` path
⚠ image file extensions
- Improved graphics `\graphicspath` support. Multiple image directories may now be used. Refer to `.pdf` files without a file extension to allow the HTML version to use a `.svg`, `.png`, `.jpg`, or `.gif` version instead. See section 9.8.
- grffile is now directly supported instead of emulated.

bigdelim
- Fix for bigdelim, and improved documentation. See section 136.

symbols
- Improved LaTeX and textcomp symbols.

fixes
- Fix for LaTeX logos and `\InlineClass`, etc. inside a `lateximage`.
- Fix for xltxtra with XeLaTeX.
- Fixes for tocbibind with `\simplechapter`, etc.
- Fixes for `\multicolumnrow` and `\nullfonts` with older versions of multirow and xparse.
- Added `\underline`.

margins
- Added adjmulticol.

columns
- Added cuted, midfloat.

footnotes
- Added pfnote, fnpos, dblfnote.

tabular
- Added stabular, tabls.

sectioning
- Added sectsty, anonchap, quotchap.

v0.39:

title pages
- Improved the titlepage HTML code, `\thanks` notes, and `\maketitle`. titling is no longer required, but is still supported. The `\published` and `\subtitle` fields are no longer provided, but `\AddSubtitlePublished` replicates them using titling. See section 65.8. authblk is added, and should be loaded before titling. See section 65.

⚠ `\published` and `\subtitle` load order

tabular
- `\multirow` now supports the new optional `vpos` argument.

multi column/row cell
- Added `\multicolumnrow` for combined `\multicolumn` and `\multirow`. See section 314.2.
- Tabular special cases:
  - Added `\TabularMacro` to mark custom macros inside tabular data cells, avoiding row corruption. See section 9.10.

⚠ macros inside tabular

⚠ tabular defined inside another environment
  - Added `\ResumeTabular` for use when a `tabular` environment is defined inside another environment. See section 9.10.

tabular
- Added supertabular, xtab, bigstrut, bigdelim.

margins
- Added fullwidth.

page layout
- Added addlines, anysize, a4, a4wide, a5comb, textarea, zwpagelayout, typearea, ebook.

v0.38:

forced single-pass compile
- Added **`lwarpmk print1`** and **`lwarpmk html1`** actions to force a compile of the project a single time. Useful when multiple passes are not needed, or changes were not detected.

starred sections
- Added `\ForceHTMLPage` and `\ForceHTMLTOC` to force a starred sectional unit onto its own HTML page and with its own TOC entry. See section 9.6.2.

updated tutorial
- Modified the tutorial to use the new `\ForceHTMLPage` and `\ForceHTMLTOC` macros.

packages
- Added appendix, tocbibind, fncychap, fix2col.

font size
- Added relsize, scalefnt.
- Added realscripts, metalogo, xltxtra.
- Added grffile, romanbar.

page numbering
- Added arabicfront, chappg, nonumonpart, nopageno, romanbarpagenumber.

front & back matter
- Docs: Improved description of the use of front/back matter. See section 9.6.
- Fix: color requests xcolor.
- Fix: `\part` for article class.

v0.37:

`\include` for HTML
- `\include` now maintains independent `.aux` files for HTML versions.

latexmk
- comment, used by lwarp, now maintains independent cut files for print and HTML versions, helping *latexmk* to better know whether to recompile.

accents and symbols
- Improved support for LaTeX accents, textcomp, siunitx symbols.

babel-french
- Improved babel-french handling for load order and ~ tilde.

v0.36:

- Recorganized the documentation section regarding special cases and limitations. (Section 9)
- Improved source formatting.

boxes and frames
- `\fbox` and related now use `\fboxsep` and `\fboxrule`.
- `\makebox` and `\framebox` now use width and position.
- `\fcolorbox` and related now work inside a `lateximage`.

babel-french
- babel-french: Improvements for French variants, load order, footnotes, ellipses.

footnotes
- Improved footnote numbering. `lateximage` footnotes now appear as regular footnotes to match the numbering of the print version. Also fixed a regression with MathJax.

siunitx
- Improved siunitx units.
- Fix for filenames while using MathJax.
- Fix for `\rule` when xcolor is not loaded.
- Added transparent, upref.

v0.35: Fix: `\textbf` and related.

v0.34:

⚠ Optional arguments
- `BlockClass`'s optional argument has been moved in front of the mandatory argument:

  `BlockClass[style]{class}` (NEW)

  instead of:

  `BlockClass{class}[style]` (OLD)

  This change makes it more consistent with LaTeX standards, and avoids problems with space between arguments.

⚠ Optional arguments
- Likewise, `\InlineClass`'s optional argument now comes before the mandatory arguments:

  `\InlineClass[style]{class}{text}`

spans with minipages
- Improved compatibility between spans, minipages, lists, frames, and math. Handles minipages and lists inside an HTML span, such as an `\fbox` containing a minipage, although with minimal HTML fomatting. See section 9.3.3.

framing minipages
- `\fboxBlock` is added to frame minipages, tables, and lists with full HTML formatting but no longer inline, and behaves as `\fbox` for print output. The `fminipage` environment is added for framed minipages, as an environment with full HTML formatting, and draws a framed minipage in print output. See section 9.3.5. `\fbox` and minipages now often work in SVG math and `lateximages`. MathJax supports `\fbox`, but not `\fboxBlock` nor `fminipage`.

`lateximage`, SVG math, `tabular`
- Improved compatibility between `lateximage` and `minipage`, `\parbox`, `\makebox`, `\fbox`, `\framebox`, `\raisebox`, `\scalebox`, `\reflectbox`, `tabular`, booktabs.
- Improved font control for `lateximagees` and SVG math.

eqnarray
- Added the `eqnarray` environments.

verbatim packages
- fancyvrb is no longer required (preloaded), but is still supported.

- Added verbatim and moreverb.

framing packages

- Added fancybox, boxedminipage2e and shadow.

list packages

- enumitem is no longer required, but is still supported.
- Added enumerate and paralist.
- titleps is no longer required, but is still supported.
- Added crop.
- Added rotfloat, marginfit, and several minor packages; see the change log.

babel-french

- Adds fixed-width HTML spaces around punctuation when using babel-french. LuaTEX does not yet use the extra punctuation spacing.

v0.33:

- Tabular `@` and `!` columns now havetheir own HTML columns.
- `&` catcode changes are localized, perhaps causing errors about the tab alignment character `&`, so any definitions of macros or environments which themselves contain `tabular` and `&` must be enclosed within `\StartDefiningTabulars` and `\StopDefiningTabulars` (previously called `\EndDefiningTabulars`. See section 43. This change is not required for the routine use of tables, but only when a table is defined inside another macro or environment, and while also using the `&` character inside the definition. This may include the use inside conditional expressions.
- Several math environments were incorrectly placed inline. Also, for amsmath with SVG math, the `fleqn` option has been removed, resulting in improved spacing for aligned equations.
- Bug fixes; see the changelog.

v0.32: Bug fixes; no source changes needed:

- *lwarpmk* has been adjusted to work with the latest *luatex*.
- Spaces in the `\usepackage` and `\RequirePackage` package lists are now accepted and ignored.
- Fix for the glossaries package and `\glo@name`.

v0.31: Bug fix; no source changes needed:

- Improved compatibility with keyfloat, including the new `keywrap` environment.

v0.30:

⚠ lwarp-newproject

- lwarp-newproject has been removed, and its functions have been combined with lwarp.

  To modify existing documents, remove from the document source:

  ```
  \usepackage{lwarp-newproject}
  ```

  The lwarp package now produces the configuration files during print output, and also accepts the option `lwarpmk` if desired.

⚠ HTML setup changes.

- A number of macros related to HTML settings have been converted to options, and other macros and options have been renamed to create a consistent syntax:

Old Macro	New Package Option
\HomeHTMLFileName	HomeHTMLFilename
\HTMLFileName	HTMLFilename
\useLatexmk	latexmk
\warpOSwindows	OSWindows

Old Package Option	New Package Option
lwarpmklang	xindyLanguage
(new)	xindyStyle

Old Macro	New Macro
\MetaLanguage	\HTMLLanguage
\HTMLauthor	\HTMLAuthor
\NewHTMLdescription	\HTMLDescription
\SetFirstPageTop	\HTMLFirstPageTop
\SetPageTop	\HTMLPageTop
\SetPageBottom	\HTMLPageBottom
\NewCSS	\CSSFilename

- Per the above changes, in existing documents, modify the package load of lwarp, such as:

```
\usepackage[
    HomeHTMLFilename=index,
    HTMLFilename={},
    xindyLanguage=english
]{lwarp}
```

- The file lwarp_html.xdy has been renamed lwarp.xdy. To update each document's project:
  1. Make the changes shown above.
  2. Recompile the document in print mode. This updates the project's configuration files, and also generates the new file lwarp.xdy.
  3. The old file lwarp_html.xdy may be deleted.
- The new lwarp package option xindyStyle may be used to tell *lwarpmk* to use a custom .xdy file instead of lwarp.xdy. See section 9.6.20.
- Improvements in index processing:
  - *xindy*'s language is now used for index processing as well as glossary.
  - Print mode without *latexmk* now uses *xindy* instead of *makeindex*.
  - *texindy*/*xindy* usage depends on *pdflatex* vs *xelatex*, *lualatex*.
  - For *pdflatex* and *texindy*, the -C utf8 option is used. This is supported in modern distributions, but a customized lwarpmk.lua may need to be created for use with older distributions.

v0.29:

- Add: lwarpmklang option for lwarp-newproject and lwarp. Sets the language to use while processing the glossary. (As of v0.30, this has been changed to the IndexLanguage option.) (As of v0.54, this has been changed to the xindyLanguage option.)

- Fix: `\includegraphics` when no optional arguments.

v0.28:

- `\HTMLAuthor {`⟨*name*⟩`}` assigns HTML meta author if non-empty. Defaults to `\theauthor`.
- Booean `HTMLDebugComments` controls whether HTML comments are added for closing `<div>`s, opening and closing sections, etc.
- Boolean `FormatEPUB` changes HTML output for easy EPUB conversion via an external program. Removes per-file headers, footers, and nav. Adds footnotes per chapter/section.
- Boolean `FormatWordProcessor` changes HTML output for easier conversion by a word processor. Removes headers and nav, prints footnotes per section, and also forces single-file output and turns off HTML debug comments. Name changed to `FormatWP` as of v0.42.
- Boolean `HTMLMarkFloats` adds text marks around floats only if the boolean `FormatWordProcessor` is `true`. These make it easier to identify float boundaries, which are to be manually converted to word-processor frames. Name changed to `WPMarkFloats` as of v0.42.
- Updated for the new MathJax CDN repository.
- Adds tabulary.
- Supports the options syntax for graphics.
- Improved index references, now pointing exactly to their target.
- Adds glossaries. *lwarpmk* is modified to add `printglosssary` and `htmlglossary` actions.

3 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 Ti*k*z 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%.

3.1 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.

3.2 Supported packages and features

Table 2 lists some of the various LaTeX features and packages which may be used. Many are tested to work as-is, some are patches for the original packages, and some are emulations written for source-level compatibility. Many are nullified as being irrelevent to HTML output.

Table 2: LaTeX lwarp package — Supported features

Category	Status and supported features.
Engines:	DVI LaTeX, pdfLaTeX, XeLaTeX, LuaLaTeX, upLaTeX
Compiling:	latexmk, perltex, pythontex, *make*, etc.
Classes:	article, book, report, scrartcl, scrbook, scrreprt, memoir, CJK-related as listed below.
Koma-script:	scrextend, scrhack, scrlayer. Others as listed below.
Memoir:	memhfixc
Languages:	babel, polyglossia. cjkpunct, xeCJK.
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, blowup, booklet, bophook, bounddvi, bxpapersize, canoniclayout, changepage, chngpage, clrdblpg, continue, draftcopy, draftfigure, draftwatermark, ebook, everyshi, fancyhdr, fancytabs, flippdf, fullminipage, fullpage, fwlw, geometry, gmeometric, grid, grid-system, gridset, layaureo, leading, ltxgrid, nccfancyhdr, notespages, pagegrid, pdfprivacy, pagesel, preview, rmpage, scrlayer-scrpage, scrpage2, textarea, threadcol, thumb, thumbs, titleps, tocenter, turnthepage, twoup, typearea, vmargin, watermark, zwpagelayout. Tested to work as-is: underlin.

lwarp Supported Functions — continued

Category	Status
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, fncychap, indentfirst, quotchap, section, sectionbreak, sectsty, titlesec. Tested to work as-is: secdot.
Table of contents, figures, tables:	Supported, with hyperlinks. multitoc, shorttoc, titletoc, tocbasic, tocbibind, tocloft, tocstyle.
Title page:	`\maketitle`, `titlepage`, authblk, titling.
Front & back matter:	abstract, appendix.
Indexing:	*makeindex* and *xindy* are supported, with hyperlinks. idxlayout, imakeidx, index, makeidx, repeatindex, splitidx. Tested to work as-is: hvindex, sepfootnotes.
Glossary:	gloss, glossaries and *xindy*, nomencl.
Bibliography:	babelbib, backref, biblatex, bibunits, chapterbib, cite, hypernat, natbib. Tested to work as-is: notes2bib.
Cross-references:	bookmark, breakurl, cleveref, fancyref, hypdestopt, hyperref, prettyref, titleref, url, varioref, xr, xr-hyper, xurl.
Margin notes:	marginal, marginfit, marginfix, scrlayer-notecolumn, versonotes.
Footnotes:	Adds `FootnoteDepth` to print footnotes at section breaks. dblfnote, endheads, endnotes, fnbreak, fnpos, footmisc, footnote, footnoterange, footnpag, marginnote, nccfoots, pagenote, parnotes, pfnote, sidenote. Tested to work as-is: fixfoot.
Math:	Converted to SVG images with HTML `<alt>` tags containing the LaTeX source for the math expression. MathJax supported as an alternative. amsmath: $\mathcal{AMS}$ environments are supported. User-defined macros are available during converson, due to native LaTeX processing.
Theorems:	Native LaTeX theorems, amsthm, ntheorem, theorem.

lwarp Supported Functions — continued

Category	Status
Additional math:	Math fonts via SVG images, breqn, cases, resizegather, xfakebold, xy. Tested to work as-is: amscd, bm, braket, delarray, guass, nicematrix, pb-diagram, tikz-cd, etc.
Display math with `\displaymathother`:	Complicated math objects in display math, such as tikz-cd, etc.
Units and fractions:	nicefrac, SIunits, siunitx, units, unitsdef, xfrac. Tested to work as-is: SIunits.
Floats:	Appear where declared. capt-of, caption, cutwin, dblfloatfix, endfloat, fix2col, flafter, float, floatflt, floatrow, fltrace, hypcap, keyfloat, morefloats, multicap, newfloat, nonfloat, placeins, rotfloat, stfloats, subcaption, subfig, subfigure, subfloat, trivfloat, wrapfig.
Tabular:	`tabular` environment, array, arydshln, bigdelim, booktabs, colortbl, diagbox, longtable, ltablex, ltxtable, multirow, supertabular, tabularx, tabulary, threeparttable, 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. picture, tikz, and xy are converted to an SVG image. asymptote, epsfig, epstopdf, figsize, fitbox, grffile, overpic, psfrag, psfragx, pst-eps, pstool, pstricks, rviewport. Tested to work as-is: curves, eepic, tikz-3dplot.
xcolor:	Full package color names, any color models, and mixing. `\textcolor`, `\colorbox`, `\fcolorbox`. Enhanced for HTML compatibility.
Lists:	Standard LaTeX environments, enumerate, enumitem, hang, paralist.
Environments:	Standard LaTeX environments.
`minipage`, `\parbox`:	Some HTML5-imposed limitations. Nested minipages are supported. pbox.
Quotations:	copyrightbox, csquotes, epigraph, quoting, verse.

lwarp Supported Functions — continued

Category	Status
Verbatim:	fancyvrb, moreverb, shortvrb, verbatim.
Frames:	boxedminipage2e, fancybox, framed, mdframed, shadow, 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, ellipsis, embrac, enparen, hyphenat, lettrine, lips, lua-check-hyphen, luacolor, magaz, nowidow, pdfrender, realscripts, relsize, scalefnt, soul, soulpos, soulutf8, textfit, thinsp, trimclip, truncate, ulem, umoline, underscore, uspace, widows-and-orphans, 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.
Symbols:	Native LaTeX diacriticals, academicons, bbding, chemgreek, dingbat, eurosym, fontawesome, fontawesome5, marvosym, pifont, textalpha, textcomp, textgreek, typicons, xunicode. Tested to work as-is: euro, gensymb.
Files:	attachfile, attachfile2, hyperxmp, intopdf, pdfpages, pdfx, xmpincl.

lwarp Supported Functions — continued

Category	Status
Science and engineering:	algorithm2e, algorithmicx, ar, axodraw2, bitpattern, bytefield, chemfig, chemformula, chemgreek, chemmacros, chemnum, karnaugh-map, listings, mhchem, phfqit, register, struktex. Tested to work as-is: askmaps, blochsphere, bodegraph, bohr, circuitikz, elements, engtlc, fast-diagram, hepnicenames, heppennames, karnaughmap, linop, pgfgantt, physics, simpler-wick, slashed, structmech, tikz-karnaugh, tikzcodeblocks.
Arts and humanities:	foreign, forest, musicography, nameauth, octave, schemata, semantic-markup, vowel, xpiano. Tested to work as-is: phonrule, piano, tikz-dependency.
Admonitions:	notes.
Editorial:	changebar, changes, easy-todo, ed, errata, fixme, fixmetodonotes, pdfcomment, pdfmarginpar, todo, todonotes, tram, xechangebar. Tested to work as-is: changelog, easyReview.
Accessibility:	accsupp, axessibility.
Debug:	chkfloat, cmdtrack, dprogress, lua-visual-debug, refcheck, 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.

4 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.

4.1 internet class

Cls internet 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

4.2 TeX4ht

Prog TeX4ht http://tug.org/tex4ht/

Prog htlatex

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.

4.3 Translators

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

Prog Hevea **HeVeA:** http://hevea.inria.fr/ (not on CTAN)

Prog TtH **TtH:** http://hutchinson.belmont.ma.us/tth/

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

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

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

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

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

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

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

4.4 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.

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

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

4.4.1 Asciidoctor-LaTeX

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

Asciidoctor-LateX:
http://www.noteshare.io/book/asciidoctor-latex-manual
Prog Asciidoctor-LaTeX https://github.com/asciidoctor/asciidoctor-latex

4.5 Pandoc

Prog Pandoc 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.)

4.6 Word processors

Prog Word
Prog LibreOffice
Prog OpenOffice

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

4.7 Commercial systems

Prog Adobe
Prog FrameMaker
Prog InDesign
Prog Flare
Prog Madcap

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

4.8 Comparisons

AsciiDoc, Pandoc, and various other markup languages typically have a syntax which tries to be natural and human-readable, but the use of advanced features tends to

require many combinations of special characters, resulting in a complicated mess of syntax. By contrast, LaTeX spells things out in readable words but takes longer to type, although integrated editors exist which can provide faster entry and a graphic user interface. For those functions which are covered by the typical markup language it is arguable that LaTeX is comparably easy to learn, while LaTeX provides many more advanced features where needed, along with a large number of pre-existing packages which provide solutions to numerous common tasks.

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

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

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

5 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

5.1 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 5.2.1 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 5.2.

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!

5.2 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 5.2.1.)

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 5.2.1.

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 5.2.1.)

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.

5.2.1 Using a local copy of *lwarpmk*

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

1. When compiling the tutorial in section 6, 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.

5.3 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 84.

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

Prog [requirement] pdftotext
Prog [requirement] pdfseparate
Prog [requirement] pdftocairo

To install the POPPLER utilities to a UNIX/LINUX system:

The tools from the POPPLER project should be provided by your operating 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)"
```

Enter

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 3 on page 84.
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.

Prog [requirement] perl

To install PERL to a WINDOWS machine:

1. Download and install a version of PERL, such as STRAWBERRY 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.

6 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. The Index of Objects contains automated entries for each package, macro, environment, counter, boolean, and other objects; individually and also sorted by category. A Troubleshooting section is also available.

6.1 Starting a new project

1. Create a new project directory called `tutorial`.

File `tutorial.tex`

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:

File `lwarp_tutorial.txt`

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

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

⚠ Note: `.txt` suffix!

   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. This file should not be modified by the user.

comment.cut: A temporary file used by lwarp to conditionally process blocks of text. This file 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.

6.2 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.

1. 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.

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

3. 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.

4. 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]

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

   Enter ⇒ **`lwarpmk printindex`**

6. Recompile again to include the index.

   Enter ⇒ **`lwarpmk print`**

7. 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 9.6.11.

[7] Also see section 9.6.14 for index options.

6.3 Compiling the HTML version with *lwarpmk*

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

1. 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]

2. View the HTML page in a web browser.

   Open the file `tutorial.html` in a web browser.

   math

   Note that math is still displayed as its `alt` tag, which is the plain-text LaTeX source, until the images of the math expressions have been generated. Math may be displayed as SVG images or by a MathJax script, as seen in sections 6.4 and 6.5.

3. Force a recompile:

   Enter ⇒ **`lwarpmk again`**

   Enter ⇒ **`lwarpmk html`**

   Enter ⇒ **`lwarpmk print`**

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

   Enter ⇒ **`lwarpmk htmlindex`**

   Enter ⇒ **`lwarpmk html`**

   `_Index.html` is updated for the new LaTeX index.

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

6. 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 9.6.11.

[11] Also see section 9.6.14 for index options.

6.4 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.

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* will inform the user with an error message.

⚠ 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. *lwarpmk* attempts to detect this situation and print a warning.

⚠ 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 6.5 to use MathJax instead.

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

6.5 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.

6.6 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 8.5 for more information about modifying the css styling of the document.

6.7 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 8.4 for more information.

6.8 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 the source, and a single recompile may be forced with:

forced single-pass recompile

Enter ⇒ **`lwarpmk print1`**

and/or

Enter ⇒ **`lwarpmk html1`**

6.9 Using XeLaTeX or LuaLaTeX

XeLaTeX 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. XeLaTeX 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`**

6.10 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 8.2 regarding font selection to avoid the use of bitmapped fonts.

6.11 Using a glossary

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

6.11.1 gloss package

See section 9.6.10.

6.11.2 glossaries package

To process the glossary for the print version:

Enter ⇒ **`lwarpmk printglossary`**

⚠ (If **`makeglossaries`** is not found, see section 9.6.11.)

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 9.6.11 to set options for processing glossaries.

6.12 Cleaning auxiliary files

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

Enter ⇒ **`lwarpmk clean`**

6.13 Cleaning auxiliary and output files

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

Enter ⇒ **`lwarpmk cleanall`**

6.14 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 `lateximages` directory:

Enter ⇒ **`lwarpmk cleanlimages`**

6.15 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.

6.16 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]`**

6.17 Processing multiple projects in the same directory

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 html -p project_b**

6.18 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.

7 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, and/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`:

```
\section{A math \texorpdfstring{$1+2=3$}{text description} name}
```

⚠ scale

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

```
\includegraphics[scale=<xx>]{ . . . }
```

   to:

```
\includegraphics[width=<yy>\linewidth]{ . . . }
```

   (c) Possible changes to `tabular` environments include `*` columns, multirow, longtable, supertabular, xtab, bigdelim. See section 9.10.
   (d) Possible option clashes with memoir. See section 9.13.
   (e) If using indexes, see section 9.6.14.
   (f) If using many indexes, glossaries, `.aux` files, etc., see section 9.6.14 regarding morewrites. If morewrites is already used, be sure to add the setup with `allocate=10`.
   (g) Other changes as per Special cases and limitations, section 9.
3. Convert any PDF images to SVG. See section 9.8.
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`, Ti*k*z, etc.

Need help?

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

8 Additional details

8.1 Shell escape

Opt --shell-escape

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*.

8.2 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 8.3.

vector fonts

Computer Modern

⚠ *pdflatex*

⚠ DVI *latex*

Pkg cm-super

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.

Pkg lmodern

To use Latin Modern instead, add

```
usepackage{lmodern}
```

to the preamble.

Pkg dejavu

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* and *lualatex*

XeLaTeX 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 XeLaTeX or LuaLaTeX:

     – fontspec and font choices

     Pkg fontspec

     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*:

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

     Pkg lmodern

     (b) `\usepackage[T1]{fontenc}`

     Pkg fontenc

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

     Pkg inputenc

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

     Pkg inputenx

     Pkg newunicodechar

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

     File glyphtounicode.tex

     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:

     – `\usepackage{cmap}`

     Pkg cmap

     (g) `\usepackage{textcomp}`

     Pkg textcomp

3. `\usepackage{lwarp}` (section 8.3) goes after any of the above, followed by:

4. `\usepackage{newtxmath}` or other math-related font packages. Many of these load amsmath, which must be loaded after lwarp, so they must also be loaded after lwarp.

⚠ fontspec with traditional font packages

5. `\setmonofont{TeX Gyre Cursor}` or similar may be required if using XeLaTeX 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.

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 10.8.

8.2.1 Indexes, glossaries, and encoding

lwarp uses the *xindy* program to processes indexes. *xelatex* and *lualatex* use *xindy* and *pdflatex* uses *texindy*.

The lwarp option `xindyLanguage` may be used to set the language option for xindy, and the `xindyCodepage` option may be used to set the codepage option for xindy. These are used for index generation.

8.3 lwarp package loading and options

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

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, . . . ⟩}
```

Pkg lwarp

lwarp package options are as follows:

Opt mathsvg
Opt mathjax

mathsvg, mathjax: For math display, select `mathsvg` (default) or `mathjax`. For more information about the math options, see section 9.7.

Opt latexmk
Default: false

latexmk: Tells *lwarpmk* to use *latexmk* to recompile the document several times if necessary. Otherwise, *lwarpmk* attempts to determing for itself whether to recompile. See section 8.4.

Opt dvips
Default: false

dvips: Tells *lwarpmk* to use *dvips* and *ps2pdf* to convert DVI output to PDF.

Opt dvipdfm
Default: false

dvipdfm: Tells *lwarpmk* to use *dvipdfm* to convert DVI output to PDF.

Opt dvipdfmx
Default: false

dvipdfmx: Tells *lwarpmk* to use *dvipdfmx* to convert DVI output to PDF.

Opt HomeHTMLFilename
Default: {}

HomeHTMLFilename: See section 8.4.

Opt HTMLFilename
Default: {}

HTMLFilename: See section 8.4.

Opt PrintLatexCmd
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`.

Opt HTMLLatexCmd
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`.

Opt makeindex
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.

Opt xindy
Default: makeindex

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

Opt makeindexStyle
Default: lwarp.ist

makeindexStyle: If you wish to use a custom `.ist` file for index generation, see section 28.

Opt xindyStyle
Default: lwarp.xdy

xindyStyle: If you wish to use a custom `.xdy` file for index generation, see section 28.

Opt xindyLanguage
Default: english

xindyLanguage: If using an index or glossary, see section 28.

Opt xindyCodepage
Default: utf8

xindyCodepage: If using an index, see section 28.

Opt PrintIndexCmd
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 9.6.14.

Table 5: 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.
PrintLatexCmd	The shell commands for **lwarpmk print**.
HTMLLatexCmd	The shell commands for **lwarpmk html**.
For indexing (section 9.6.14) and glossaries (section 9.6.11):	
makeindex	Use *makeindex* to generate indices.
xindy	Use *xindy* to generate indices.
makeindexStyle	Set a custom style for *makeindex*.
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.
PrintIndexCmd	Shell commands executed by **lwarpmk printindex**.
HTMLIndexCmd	Shell commands executed by **lwarpmk htmlindex**.
LatexmkIndexCmd	Shell commands executed by *latexmk*.
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.

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.

Opt `HTMLIndexCmd`
Default: `<automatic>`

`HTMLIndexCmd:` Sets the shell commands executed by **`lwarpmk htmlindex`**. If not specified, will be set by the selection of `makeindex` or `xindy`. May be used to specify the creation of multiple indexes. See section 9.6.14.

⚠ 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 worthwhile to use a shell script, then refer to that script with `HTMLIndexCmd`.

Opt `LatexmkIndexCmd`
Default: `<automatic>`

`LatexmkIndexCmd:` Sets the shell commands executed by *latexmk*. Unlike `PrintIndexCmd` and `HTMLIndexCmd`, `LatexmkIndexCmd` does not include any filenames, which will be provided instead by *latexmk*. See section 9.6.14.

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

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

automatic setting

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

⚠ xindy

If specifying `LatexmkIndexCmd` manually, be sure to assign an *xindy* language

and codepage with the `-L` and `-C` *xindy* options, as the lwarp `xindyLanguage` and `xindyCodepage` options are not used for the `LatexmkIndexCmd` option when it is set manually.

Opt GlossaryCmd
Default: **makeglossaries**

GlossaryCmd: Sets the shell command executed by **`lwarpmk printglossary`** and **`lwarpmk htmlglossary`**. The print or HTML glossary filename is appended to this command. See section 9.6.11.

Opt OSWindows

OSWindows: lwarp attempts to automatically sense WINDOWS, but it may be forced with this option. See section 8.6.

Opt pdftotextEnc
Default: UTF-8

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

Opt lwarpmk

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 28.

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

Opt warpprint
Opt warpHTML

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

Opt BaseJobname
Default: `\jobname`

BaseJobname: Not intended for the user. Used internally by lwarp when creating the `*_html.tex` file used to compile the HTML version. See section 28.

8.4 Customizing the HTML output

Table 6 shows several settings may be used to customize the HTML output. Watch for the correct placement of each!

⚠ Placement!

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

1. Clear all the HTML, PDF, and auxiliary files:
   Enter ⇒ **`lwarpmk clearall`**
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`**

Options for the lwarp package:

Use the following as options for `\usepackage[<options>]{lwarp}`:

Opt HomeHTMLFilename
Default: \BaseJobname

HomeHTMLFilename: 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 `HomeHTMLFilename` and `HTMLFilename` options, but may need to be escaped elsewhere, such as when appearing in a list:

filename underscores

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

See section 8.4.1 for examples of naming and numbering HTML files.

Opt HTMLFilename
Default: <empty>

HTMLFilename: A filename prefix for the rest of the HTML web pages. Useful for numbered web pages with a common prefix. May be empty. See section 8.4.1 for examples of naming and numbering HTML files.

Opt latexmk
Default: false

latexmk: Controls whether lwarp uses *latexmk* to compile the document. This setting is written to *lwarpmk*'s configuration files.

Opt mathsvg
Default: true

mathsvg: Selects SVG display for math output. (The default.)

Opt mathjax
Default: false

mathjax: Selects MathJax for math output.

Opt makeindex
Default: makeindex

makeindex: Selects *makeindex* for index generation by *lwarpmk*.

Opt xindy
Default: makeindex

xindy: Selects *xindy* for index generation by *lwarpmk*.

Placed in the preamble before `\begin{document}`:

`\linkhomename`
Default: Home

`\linkhomename`: Name of the link to the home page. Paragraphs are allowed.

Table 6: HTML settings

Macro/Cntr/Bool	Description
\linkhomename	Name of the link to the homepage.
SideTOCDepth	Sectioning depth of the sideTOC.
\sidetocname	Name of the sideTOC.
FileDepth	Sectioning depth of the file splits.
CombineHigherDepths	Combine higher section levels.
FileSectionNames	Use section names for file names, else use numbers.
\FilenameLimit	Maximum length of the generated filenames.
FootnoteDepth	Sectioning depth of footnotes.
\abstractname	The name of the abstract.
\mathimagename	The SVG math image lateximage alt tag.
\packagediagramname	The suffix for a package's lateximage alt tags.
\CSSFilename	The CSS for the following files.
\HTMLLanguage	The HTML lang tag.
\HTMLTitle	The homepage's <title>, overriding \title.
\HTMLTitleBeforeSection	Set subpage <title>s to \HTMLTitle - sectionname
\HTMLTitleAfterSection	Set subpage <title>s to sectioname - \HTMLTitle
\HTMLAuthor	The HTML author meta tag, overriding \author.
\HTMLDescription	The HTML description meta tag.
\HTMLFirstPageTop	Heading for the home page.
\HTMLPageTop	Heading for the other pages.
\HTMLPageBottom	Footing for all pages.
HTMLDebugComments	Boolean to generate HTML comments.

Ctr tocdepth

tocdepth: Sectioning depth of the table of contents. See section 17 for a list of LaTeX stack depths.

Ctr SideTOCDepth
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 web page 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 and a link back to **Home** at the bottom.

It is recommended to set:

```
SideTOCDepth = FileDepth
```

or

```
SideTOCDepth = FileDepth+1
```

⚠ inaccessible pages

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

\sidetocname
Default: Contents

\sidetocname: Name of the sideTOC. Paragraphs are allowed.

Ctr FileDepth
Default: -5

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

- To place the entire file into one HTML page, use:
  `\setcounter{FileDepth}{-5}`
- To split the HTML file at \section depth, use:
  `\setcounter{FileDepth}{1}`
- ⚠ To ensure that the HTML pages/files are accessible:
  Place a \tableofcontents somewhere before the first section break (therefore in the "home page"), and set
  `tocdepth >= FileDepth`

Bool CombineHigherDepths
Default: true

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

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

⚠ Inaccesible pages!

⚠ 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.

Bool FileSectionNames
Default: true

⚠ Unique filename!

FileSectionNames: If true, web page filenames are derived from a sanitized version of the section names. If false, web pages are numbered. Either way, the HTMLFilename option is used as a prefix. See section 8.4.1 for examples of naming and numbering HTML files. The user must ensure 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
Default: 80

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

Ctr FootnoteDepth
Default: 3

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

Bool HTMLDebugComments
Default: false

HTMLDebugComments: Set true to generate HTML comments, such as which section or 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".

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.

\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.

\HTMLFirstPageTop
Default: <empty>

\HTMLFirstPageTop: {⟨*contents*⟩} A user-definable custom action applied to the top of the home page. Useful for logos, etc. Defaults empty. Ignored in print output.

\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. Ignored in print output.

\HTMLPageBottom
Default: <empty>

\HTMLPageBottom: {⟨*contents*⟩} A user-definable custom action applied to the bottom of each web page. Useful for authors, copyright notices, contact information, etc. Defaults empty. \LinkHome may be used to place a link back to the homepage. Ignored in print output.

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:

\mathimagename
Default: math image

\mathimagename: When creating an SVG math image, its 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 \mathimagename, and the image is not reused.

This 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.

\packagediagramname
Default: diagram

\packagediagramname: For many packages, the output is placed inside a lateximage with an HTML alt tag set to the package name followed by \packagediagramname. 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 `lateximages`.

Env warpprint

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}`.

Env warpHTML

warpHTML: An environment which is only used 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}`.

\warpprintonly

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

\warpHTMLonly

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

8.4.1 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}
```

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

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}
```

8.5 Customizing the css

`\CSSFilename`
Default: `lwarp.css`

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

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

The styles provided by lwarp include:

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

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

`lwarp_sagebrush.css`: A style with muted colors, gradient backgrounds, 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.

File lwarp.css
File project.css
File sample_project.css

It is best to make a local project-specific CSS file such as `project.css`, containing only things which are different from `lwarp.css`. The file `project.css` should 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.

8.6 Selecting the operating system

Prog Unix
Prog Mac OS
Prog Linux
Prog MS-Windows
Prog Windows
Opt OSWindows

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.

8.7 Selecting actions for print or HTML 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.

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.

Env warpHTML

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.

Env `warpprint` 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}`.

Env `warpall` 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}
```

As above, do not place anything else on the line with `\end{warpall}`.

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.

8.8 Commands to be placed into the `warpprint` environment

Certain print-related commands should always be placed inside a `warpprint` environment, or may need other special handling. These are unrelated to HTML output, but are hard to isolate automatically. For example:

- Paragraph formatting: `\parindent` `\parskip`
- Manual page positions such as the textpos package, which is emulated but only in a limited way.

- Anything changing the page counter. lwarp requires that the page counter not be adjusted during HTML output.

Some packages require additional setup commands. Where these packages are emulated for HTML, setup commands may work for the emulated HTML output as well as for print output. See the details for each package in this document for more information.

Also see section 14: Troubleshooting.

8.9 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 65.8.

`\maketitle` Use `\maketitle` just after the `\begin{document}`, as this will establish the title of the homepage. Optionally, use a `titlepage` environment instead.

Env `titlepage` 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.

Env `titlingpage` Another form of custom title page, where `\maketitle` is allowed, and additional information may be included as well.

`\title` {⟨*title*⟩}

⚠ newlines Avoid newlines in the `\title`; these will interfere with the file break and css detection. Use a `\subtitle` command instead (section 65.8). 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.

8.10 HTML page meta descriptions

\HTMLDescription {⟨*A description of the web page.*⟩} The default is no description.

limitations Each page of HTML output should have its own HTML meta description, which 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{}
```

8.11 HTML homepage meta title

\HTMLTitle {⟨*title*⟩} Sets the contents of the web page <meta name="title"> element. Defaults to \HTMLtitle{\thetitle}. May be set empty to cancel the meta title tag.

See section 8.4 for \HTMLTitleBeforeSection and \HTMLTitleAfterSection, used to set the title for HTML subpages.

8.12 HTML page meta author

\HTMLAuthor {⟨*author*⟩} Sets the contents of the web page <meta name="author"> element. Defaults to \HTMLAuthor{\theauthor}. 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.

9 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.

Need help?

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

9.1 Things to avoid

In the document, avoid the following:

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}}
```

9.1.1 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.

9.2 Formatting

9.2.1 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.

9.2.2 Horizontal space

`\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`.

`\,`

`~` and `\,` are converted to HTML entities.

`\kern` `\hskip`

`\kern` and `\hskip` are entered into the HTML PDF output as-is, then interpreted by *pdftotext*, and thus usually appear as a single space.

9.2.3 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}
. . .
```

9.2.4 Accents

Native LaTeX accents such as `\"` will work, but many more kinds of accents are available when using Unicode-aware XeLaTeX and LuaLaTeX.

9.2.5 textcomp package

Pkg `textcomp`

Some textcomp symbols do not have Unicode equivalents, and thus are not supported.

⚠ missing symbols

Many textcomp symbols are not supported by many fonts. In the CSS try referencing fonts which are more complete, but expect to see gaps in coverage.

9.2.6 Superscripts and other non-math uses of math mode

Use `\textsuperscript{x}` instead of `$^{x}$`

9.2.7 Empty `\item` followed by a new line of text or a nested list:

lists Use a trailing backslash: `\item[label] \`

9.2.8 Filenames and URLs in lists or footnotes

filename underscore Escape underscores in the filenames:

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

9.2.9 relsize package

Pkg relsize 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.

9.3 Boxes and minipages

9.3.1 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.

9.3.2 Save Boxes

TEX boxes are placed inline and do not allow line breaks, so boxes with long contents may overflow the line during HTML conversion. This is mostly a problem when the boxes contain objects which themselves hold large HTML tags, such as rotation commands with long contents. When this object overflows the line, some HTML code will be lost and the page will be corrupted.

9.3.3 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 `\linewidth`, `\textwidth`, and `\textheight`, widths and heights in HTML are scaled proportionally to a 6×9 inch text area, and inside a `multicols` `\linewidth` is divided by the specified number of columns.

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.

⚠ 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.

9.3.4 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.

9.3.5 Framed minipages and other environments

\fbox can only be used around inline items during HTML output, but HTML cannot place a block element such as a for a minipage or a list inside of a . Several options are provided for framing an object, depending on which kind of object and which packages are loaded:

\fbox
\fboxBlock
Env fminipage

For a framed object, options include:

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.

For inline text:

To frame the contents inline with some formatting losses in HTML: This is the 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 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.

For inline minipage and lists:

To frame the contents on their own line with improved formatting in HTML: A 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 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 not a , 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.

For display tabular, minipages, and lists:

To create a framed minipage in both print and HTML: A new environment fminipage is included. For print output, this is identical to minipage, except that it is also framed. For HTML output, this forms a of class framed, the contents preserve their HTML formatting, and valid HTML is created for a tabular. Also see section 84 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 476 for xcolor's \colorbox and \fcolorbox, and lwarp's additional \colorboxBlock and \fcolorboxBlock.

⚠ Misplaced alignment tab character &

To frame tables or verbatim environments: Place the contents inside a fminipage, or perhaps a \fboxBlock for a tabular. Also, if using \fboxblock with tabular, you will have to use \StartDefiningTabulars before the start of the macro which uses \fboxBlock and the tabular, and \StopDefiningTabulars afterwards. Also see the lwarp documentation for the fancybox package.

To frame equations: See section 209 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}
```

9.3.6 fancybox package

Pkg fancybox

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 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 loses 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.

9.3.7 mdframed package

Pkg mdframed support

Most basic functionality is supported, including frame background colors and single-border colors and thickness, title and subtitle background colors and borders and thickness, border radius, and shadow. CSS classes are created for mdframed environments and frame titles.

⚠ loading

When used, lwarp loads mdframed in HTML with `framemethod=none`.

font

For title font, use

```
frametitlefont=\textbf,
```

instead of

```
frametitlefont=\bfseries,
```

where \textbf must appear just before the comma and will receive the following text as its argument (since the text happens to be between braces in the mdframed source). Since lwarp does not support \bfseries and friends, only one font selection may be made at a time.

theoremtitlefont

theoremtitlefont is not supported, since the following text is not in braces in the mdframed source.

ignored options

userdefinedwidth and align are currently ignored.

css classes

Environments created or encapsulated by mdframed are enclosed in a of class mdframed, and also class md<environmentname> for new environments.

Frame titles are placed in a of class |mdframedtitle|. Subtitles are in a of class |mdframedsubtitle|, and likewise for subsubtitles.

9.4 Section names

If using named HTML files, by selecting `\booltrue{FileSectionNames}`:

`\FilenameSimplify` 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 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`:

```
\section{A math \texorpdfstring{$1+2=3$}{text description} name}
```

9.5 Cross-references

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.

9.5.1 Page references

⚠ LaTeX page numbers

The printed page does not translate to the HTML page, so `\pageref` references are converted to parentheses containing `\pagerefPageFor`, which defaults to "see ", followed by a hyperlink to the appropriate object.

Ex:

```
\ref{sec:name} on page \pageref{sec:name}
```

in HTML becomes:

"Sec. 1.23 on page (**see** sec. 1.23)".

`\pagerefPageFor` may be redefined to "page for ", empty, etc. See page 457.

9.5.2 cleveref and varioref packages

Pkg cleveref
Pkg varioref

⚠ 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 529 to redefine the message which is printed for page number references.

9.5.3 Hyperlinks, hyperref, and url

Pkg hyperref
Pkg url

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.

9.5.4 Footnotes 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.

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

If using memoir class, with which lwarp preloads footmisc, the `stable` option must be declared before lwarp is loaded:

```
\PassOptionsToPackage{stable}{footmisc}
\usepackage{lwarp}
. . .
```

Do not use a starred sectioning command. As an alternative, it may be possible to adjust `\secnumdepth` instead.

⚠ `\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.

⚠ 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.

9.6 Front and back matter

9.6.1 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}
```

9.6.2 Starred chapters and sections

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 but no correct TOC entry pointing to that page, the page will be inaccessible unless some other link is created.

⚠ inaccessible HTML page

\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.

9.6.3 abstract package

Pkg abstract

⚠ 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.

9.6.4 titling and authblk

Pkg titling

Pkg authblk

package support

⚠ load order

\published and \subtitle

lwarp supports the native LaTeX titling commands, and also supports the packages authblk and titling. If both are used, authblk should be loaded before titling.

If using the titling package, additional titlepage fields for \published and \subtitle may be added by using \AddSubtitlePublished in the preamble. See section 65.8.

9.6.5 tocloft package

Opt [tocloft] titles
Pkg tocloft
Pkg tocloft
⚠ tocloft & other packages

If using tocloft with tocbibind, anonchap, fncychap, or other packages which change chapter title formatting, load tocloft with its `titles` option, which tells tocloft to use standard LaTeX commands to create the titles, allowing other packages to work with it.

9.6.6 appendix package

Pkg appendix
⚠ 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.

9.6.7 pagenote package

Pkg pagenote

pagenote works as-is, but the `page` option is disabled.

9.6.8 endnotes package

Pkg endnotes
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
```

9.6.9 BibTeX

`\etalchar`

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

⚠ Modify *.bib

When enough authors are cited for a source, BibTeX may use the `\etalchar` command to display a math superscript with a + character to indicate "and others". Without modification, this will result in an "Improper `\prevdepth`" error. At present, lwarp requires that `\etalchar` be replaced by a text superscript. To do so, add to the start of the `.bib` file the following:

```
@PREAMBLE{"\let\etalchar\relax \newcommand{\etalchar}[1]{\textsuperscript{#1}}"}
```

9.6.10 gloss package

Pkg gloss

⚠ compiling

To process the HTML glossary:

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

9.6.11 glossaries package

Pkg glossaries

processing glossaries

Opt GlossaryCmd

Default: makeglossaries

Opt [lwarpmk] printglossary

Opt [lwarpmk] htmlglossary

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.

9.6.12 nomencl package

Pkg nomencl — To process the HTML nomenclature:

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

9.6.13 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 9.6.19 and 9.6.20.

source code — See section 75 for lwarp's core index and glossary code, section 270 for index, section 409 for splitidx, section 269 for imakeidx, section 441 for tocbibind, and section 494.17 for memoir's indexing patches.

9.6.14 Indexing with basic LaTeX and makeidx

lwarpmk processing — The following allow the user to process indexes automatically, or using *lwarpmk*'s commands:

Enter ⇒ **`lwarpmk printindex`**

Enter ⇒ **`lwarpmk htmlindex`**

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.

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.

9.6.15 Indexing with index

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.

9.6.16 Indexing with splitidx

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 494.17 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.

9.6.17 Indexing with imakeidx

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 9.6.19 and 9.6.20. 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.

9.6.18 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.

9.6.19 Using a custom *makeindex* style file

Prog makeindex
File lwarp.ist

When using *makeindex*, *lwarpmk* uses the file `lwarp.ist` to process the index. 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.

Opt makeindexStyle

3. In the document source use the `makeindexStyle` option for lwarp:

```
\usepackage[
    ... other options ...
```

```
    \textred{makeindexStyle=projectname.ist},
]{lwarp}
```

Likewise, refer to the custom style file if using `\PrintIndexCmd`, `\HTMLIndexCmd`, or `\LatexmkIndexCmd`.

4. 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`.

9.6.20 Using a custom *xindy* style file

Prog xindy
File lwarp.xdy

When using *xindy*, *lwarpmk* uses the file `lwarp.xdy` to process the index. This file is over-written by lwarp whenever a print version of the document is processed.

To use a custom *xindy* style file:

1. Copy `lwarp.xdy` to a new filename such as `projectname.xdy`
2. Make changes to `projectname.xdy`.

   Keep the lines which refer to `\hyperindexref`:

   ```
   (define-attributes (("hyperindexref")))
   (markup-locref :open "\hyperindexref{" :close "}")
   ...
   (markup-locref :open "\textit{\hyperindexref{" :close "}}" :attr "textit")
   ```

   These lines create the hyperlinks for the HTML index. During print output `\hyperindexref` becomes a null function.

   To create custom styles, refer to the lines for `\textbf` and `\textit`.

Opt xindyStyle

3. In the document source use the `xindyStyle` option for lwarp:

   ```
   \usepackage[
       ... other options ...
       \textred{xindyStyle=projectname.xdy},
   ]{lwarp}
   ```

   Likewise, refer to the custom style file if using `\PrintIndexCmd`, `\HTMLIndexCmd`, or `\LatexmkIndexCmd`.

4. 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`.

9.6.21 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.

9.6.22 Index positions, TOC, tocbibind

placement and TOC options

An index may be placed inline with other HTML text, or on its own HTML page:

Pkg makeidx **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
```

Pkg makeidx **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
```

Pkg tocbibind **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
```

Pkg tocbibind **On its own HTML page, with an automatic TOC entry:**

```
\usepackage[nottoc]{tocbibind}
. . .
\cleardoublepage
\phantomsection % to fix print-version index link
\ForceHTMLPage
\printindex
```

Opt [tocbibind] numindex
numbered index section

Use the tocbibind `numindex` option to generate a numbered index. Without this option, the index heading has no number.

Other packages, such as imakeidx, may also have options for including the index in the Table of Contents.

Pkg tocloft
⚠ tocloft & other packages

If using tocloft with tocbibind, anonchap, fncychap, or other packages which change chapter title formatting, load tocloft with its `titles` option, which tells tocloft to use standard LaTeX commands to create the titles, allowing other packages to work with it.

9.7 Math

9.7.1 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`:

```
\section{A math \texorpdfstring{$1+2=3$}{text description} name}
```

9.7.2 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 in-line 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 11 regarding EPUB output with MathJax.

9.7.3 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:

```
\newcommand*{\LateximageFontScale}{.75}
```

SVG math copy/paste

For SVG math, text copy/paste from the HTML `<alt>` tags lists the equation number or tag for single equations, along with the LaTeX code for the math expression. For $\mathcal{AMS}$ environments with multiple numbers in the same environment, only the first and last is copy/pasted, as a range. No tags are listed inside a starred $\mathcal{AMS}$ environment, although the `\tag` macro will still appear inside the LaTeX math expression.

⚠ SVG math in TeX boxes

SVG math does not work inside TeX boxes, since a `\newpage` is required before and after each image.

9.7.4 MathJax option

MathJax math option

Prog MathJax

The popular MathJax alternative (`mathjax.org`) may be used to display math.

When MathJax is enabled, math is rendered twice:

1. As regular LaTeX PDF output placed inside an HTML comment, allowing equation numbering and cross referencing to be almost entirely under the control of LaTeX, and

[14] See section 481 regarding fonts and fractions.

2. As detokenized printed LaTeX commands placed directly into the HTML output for interpretation by the MathJax display scripts. An additional script is used to pre-set the equation number format and value according to the current LaTeX values, and the MathJax cross-referencing system is ignored in favor of the LaTeX internal system, seamlessly integrating with the rest of the LaTeX code.

9.7.5 Customizing MathJax

MathJax does not have preexisting support every possible math function. Additional MathJax function definitions may be defined. These will be declared at the start of each HTML page, and thus will have a global effect.

Examples:

```
\CustomizeMathJax{
    \newcommand{\expval}[1]{\langle#1\rangle}
    \newcommand{\abs}[1]{\lvert#1\rvert}
}
\CustomizeMathJax{\newcommand{\arsinh}{\text{arsinh}}}
\CustomizeMathJax{\newcommand{\arcosh}{\text{arcosh}}}
\CustomizeMathJax{\newcommand{\NN}{\mathbb{N}}}
```

9.7.6 MathJax limitations

MathJax limitations
Prog MathJax

Limitations when using MathJax include:

subequations
- MathJax itself does not support subequations. This may be improved by parsing the LaTeX math expression to manually insert tags, but this has not yet been done.

footnotes in math
- Footnotes inside equations are not yet supported while using MathJax.

lateximage
- Math appearing inside a lateximage, and therefore also inside a Ti*k*z or `picture` environment, is rendered as SVG math even if MathJax is used in the rest of the document.

siunitx
- Usage of siunitx inside a math equation is supported via a third-party MathJax extension. While inside a math expression, do not use `\SI` or `\si` inside `\text`, where it will be rendered as normal text.

⚠ siunitx inside an equation

https://github.com/burnpanck/MathJax-siunitx

Also see section 9.7.11.

tabbing
- A `tabbing` environment is emulated using an HTML <pre>.While MathJax is enabled inside `tabbing`, the browser may not correctly render the horizontal alignment of the math and text following after on the same line.

⚠ other macros and packages
- Other math-related macros and packages are not supported by MathJax, including `\ensuremath` and bigdelim, along with occasionally-used macros such as `\relax`. lwarp emulates footnotes, units, and nicefrac for MathJax.

9.7.7 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.

9.7.8 Complicated inline math objects

`\inlinemathnormal`
`\inlinemathother`

changing contents

complicated `alt` tag

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. The macro `\inlinemathother` may be used before a dynamic math expression, and `\inlinemathnormal` after. Doing so tells lwarp to use an unhashed SVG math image, even if MathJax is in use. See section 44.

9.7.9 Complicated display math objects

`\displaymathnormal`

By default, or when selecting `\displaymathnormal`, MATHJAX math display environments print their contents as text into HTML, 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`

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 Ti*k*z 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 Ti*k*z pictures are more likely to compile successfully.

9.7.10 ntheorem package

Pkg ntheorem

⚠ 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 $\mathcal{AMS}$ environmentswhen 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.

9.7.11 siunitx package

Pkg siunitx

fractions

Due to *pdftotext* limitations, fraction output is replaced by symbol output for `per-mode` and `quotient-mode`.

⚠ math mode required

Some units will require that the expression be placed inside math mode.

NOTE: As of this writing, the siunitx extension for MathJax is not currently hosted at any public CDN, thus siunitx is not usable with MathJax unless a local copy of this extension is created first.

⚠ tabular

Tabular `S` columns are rendered as simple `c` columns, and tabular `s` columns are not supported. These may be replaced by `c` columns with each cell contained in `\num` or `\si`.

9.7.12 units and nicefrac packages

Pkg units

Pkg nicefrac

units and nicefrac work with lwarp, but MathJax does not have an extension for units or nicefrac. These packages do work with lwarp's option `svgmath`.

9.7.13 newtxmath package

Pkg newtxmath

⚠ loading sequence

The proper load order is:

```
...
\usepackage{lwarp}
...
\usepackage{amsthm}
\usepackage{newtxmath}
...
```

9.8 Graphics

Pkg graphics

Pkg graphicx

⚠ scale

Avoid using the `\includegraphics` `scale` option. Change:

```
\includegraphics[scale=<xx>]{ . . . }
```

to:

```
\includegraphics[width=<yy>\linewidth]{ . . . }
```

`\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
```

Prog pdftocairo

PDF to SVG

To convert a PDF image to SVG, use the utility *pdftocairo*:

Enter ⇒ **`pdftocairo -svg filename.pdf`**

Prog lwarpmk pdftosvg

For a large number of images, use *lwarpmk*:

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

Prog lwarpmk epstopdf

Prog epstopdf

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.

⚠ graphics vs. graphicx

⚠ viewport

⚠ viewport units

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.

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.

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 of "`(image)`".

`\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.

9.8.1 tikz package

Pkg `tikz`

⚠ 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 `\&`.

9.8.2 grffile package

Pkg `grffile`

⚠ 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.

9.8.3 color package

Pkg `color`

color is superceded by xcolor, and lwarp requires several of the features of xcolor. When color is requested, xcolor is loaded as well.

9.8.4 xcolor package

Pkg `xcolor`

`\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.

9.8.5 epstopdf package

Pkg epstopdf

⚠ 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.

9.8.6 pstricks package

Pkg pstricks

⚠ use `pspicture`

All pstricks content should be contained inside a `pspicture` environment.

9.8.7 pdftricks package

Pkg pdftricks

⚠ 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`**

9.8.8 psfrag package

Pkg psfrag

⚠ 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.

9.8.9 pstool package

Pkg pstool

\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**

.

9.8.10 asymptote package

Pkg asymptote

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
```

9.8.11 overpic package

Pkg overpic

⚠ 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.

9.9 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.

9.10 Tabular

Tabular mostly works as expected, but pay special attention to the following, especially if working with environments, macros inside tabulars, multirows, * column specifiers, 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  % because  &  is  used  in  a
definition
\newenvironment{outerenvironment}
{
\tabular{cc}
left & right \\
}
{
\TabularMacro\ResumeTabular
left & right \\
\endtabular
}
\StopDefiningTabulars
```

Cell contents:

⚠ `\multirow`

- 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.

```
. . . &  \multirow{2}{.5in}{text}  & . . .
. . . &  \mrowcell                 & . . .
```

`vposn`

Note that recent versions of multirow include a new optional `vposn` argument.

- 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.

- See section 314.2 for \multicolumrow.

⚠ \multicolumn & \multirow

lwarp does not support directly combining \multicolumn and \multirow. Use \multicolumnrow instead. To create a 2 column, 3 row cell:

```
\multicolumnrow{2}{c}[c]{3}[0]{1in}[0pt]{Text}
```

The two arguments for \multicolumn come first, followed by the five arguments for \multirow, many of which are optional, followed by the contents.

⚠ skipped cells

As per \multirow, skipped cells to the right of the \multicolumnrow statement are not included in the source code on the same line. On the following

⚠ empty cells

lines, \mcolrowcell must be used for each cell of each column and each row to be skipped:

```
... & \multicolumnrow{2}{c}[c]{3}[0]{1in}[0pt]{Text}  & ...
... & \mcolrowcell          &          \mcolrowcell   & ...
... & \mcolrowcell          &          \mcolrowcell   & ...
```

vposn

Note that recent versions of multirow include a new optional vposn argument.

⚠ 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:

⚠ * column specification

- * in a column specification is not used (so far). Repeat the column type the correct number of times.

@ 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

- \newcolumntype is ignored; unknown column types are set to l.

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.

full-width rules

- \toprule, \midrule, \bottomrule, and \hline 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.

colortbl:

⚠ row/cell color

Only use \rowcolor and \cellcolor at the start of a row, in that order.

colortbl ignores the overhang arguments.

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 \\}
  ```

9.10.1 longtable package

Pkg longtable
⚠ Misplaced \noalign

Longtable \endhead, \endfoot, and \endlastfoot rows are not used for HTML, and these rows should be disabled. Use

```
\warpprintonly{row contents}
```

instead of

```
\begin{warpprint} . . . \end{warpprint}
```

Doing so helps avoid "Misplaced \noalign." when using \begin{warpprint}.

Keep the \endfirsthead row, which is still relevent to HTML output.

⚠ \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.

9.10.2 supertabular and xtab packages

Pkg supertabular
Pkg xtab
⚠ Misplaced alignment tab character &

For `\tablefirsthead`, etc., enclose them as follows:

```
\StartDefiningTabulars
\tablefirsthead
. . .
\StopDefiningTabulars
```

See section 9.10.

⚠ lateximage

supertabular and xtab are not supported inside a `lateximage`.

9.10.3 bigdelim package

Pkg bigdelim
⚠ 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{\{}{2}{.25in}[left ] & c & d \\
\mrowcell & e & f \\
<empty> & g & h \\
\end{tabular}
```

<> a b
left { c d
e f
<> g h

9.11 Floats

9.11.1 Float contents alignment

⚠ 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}
. . .
```

9.11.2 float, trivfloat, and/or algorithmicx together

Pkg float
Pkg trivfloat
Pkg algorithmx
⚠ package conflicts

If using `\newfloat`, trivfloat, and/or algorithmicx together, see section 450.1.

9.11.3 caption and subcaption packages

Pkg caption
Pkg subcaption
⚠ options

To pass options to caption, select the options before loading lwarp:

```
\documentclass{article}
. . .
\PassOptionsToPackage{options_list}{caption}
. . .
\usepackage{lwarp}
. . .
\usepackage{caption}
```

⚠ 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.

9.11.4 subfig package

Pkg subfig

⚠ 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.

In the document source, use `\hfill` and `\hspace*` subfig>inline 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.

9.11.5 floatrow package

Pkg floatrow

Use `\StartDefiningTabulars` and `\StopDefiningTabulars` before and after defining macros using `\ttabbox` with a tabular inside. See section 9.10.

⚠ Misplaced alignment tab character &
⚠ subfig package

When combined with the subfig package, while inside a `subfloatrow` `\ffigbox` and `\ttabbox` must have the caption in the first of the two of the mandatory arguments.

⚠ \FBwidth, \FBheight

The emulation of floatrow does not support `\FBwidth` or `\FBheight`. These values are pre-set to `.3\linewidth` and `2in`. Possible solutions include:

- Use fixed lengths. lwarp will scale the HTML lengths appropriately.
- Use `warpprint` and `warpHTML` environments to select appropriate values for each case.

- Inside a warpHTML environment, manually change \FBwidth or \FBheight before the \ffigbox or \ttabbox. Use \FBwidth or \FBheight normally afterwards; it will be used as expected in print output, and will use your custom-selected value in HTML output. This custom value will be used repeatedly, until it is manually changed to a new value.

9.11.6 keyfloat package

Pkg keyfloat

⚠ keywrap

If placing a \keyfig[H] inside a keywrap, use an absolute width for \keyfig, instead of lw-proportional widths. (The [H] option forces the use of a minipage, which internally adjusts for a virtual 6-inch wide minipage, which then corrupts the lw option.)

9.12 KOMA-SCRIPT classes

Cls komascript

Many features are ignored during the HTML conversion. The goal is source-level compatibility.

\titlehead, \subject, \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.

9.13 MEMOIR class

Cls memoir

⚠ options clash

While emulating memoir, lwarp pre-loads a number of packages (section 494.1). This can cause an options clash when the user's document later loads the same packages with options. To fix this problem, specify the options before loading lwarp:

```
\documentclass{memoir}
. . .
\PassOptionsToPackage{options_list}{package_name}
. . .
\usepackage{lwarp}
. . .
\usepackage{package_name}
```

⚠ version numbers

memoir emulates a number of packages, and declares a version date for each which often does not match the date of the corresponding freestanding package. This can cause warnings about incorrect version numbers. Since lwarp is intended to support the freestanding packages, which are often newer than the date declared by memoir, it is hoped that memoir will update and change its emulated version numbers to match.

\verbfootnote is not supported.

\newfootnoteseries, etc. are not supported.

lwarp loads pagenote to perform memoir's pagenote functions, but there are minor differences in \pagenotesubhead and related macros.

Poem numbering is not supported.

The verbatim environment does not yet support the memoir enhancements. It is currently recommended to load and use fancyvrb instead.

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.

9.14 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.)

9.15 Miscellaneous packages

9.15.1 verse and memoir

Pkg verse
Cls memoir
\attrib

When using verse or memoir, always place a \\ after each line.

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}
```

Len \vleftskip
Len \vleftmargini
Len \HTMLvleftskip
Len \HTMLleftmargini

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 and \HTMLleftmargini are provided to control the margins in HTML output. These 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 case a label may be shifted by one space.

9.15.2 newclude package

Pkg newclude

⚠ loading

newclude modifies \label in a non-adaptive way, so newclude must be loaded before lwarp is loaded:

```
\documentclass{article}
. . . <font setup>
\usepackage{newclude}
\usepackage[warpHTML]{lwarp}
. . .
```

9.15.3 babel package

Pkg babel

⚠ \CaptionSeparator

When French is used, the caption separator is changed to a dash. The following may be used to restore it to a colon:

```
\renewcommand*{\CaptionSeparator}{:~}
```

punctuation spaces

⚠ customized spacing

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 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.

9.15.4 polyglossia package

Pkg polyglossia

lwarp uses cleveref, which has some limitations when using polyglossia, possibly resulting in the error

```
! Undefined control sequence. . . . \@begindocumenthook
```

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.

9.15.5 todonotes and luatodonotes packages

Pkg todonotes
Pkg luatodonotes

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.

9.15.6 fixme

Pkg fixme

External layouts (`\fxloadlayouts`) are not supported.

⚠ external layouts

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}
```

9.15.7 chemfig package

If using `\polymerdelim` to add delimiters to a `\chemfig`, wrap both inside a single `lateximage`:

```
\begin{lateximage}[(-chemfig-~\packagediagramname)]
\chemfig{. . . }
\polymerdelim[. . . ]{. . . }
\end{lateximage}
```

9.15.8 chemformula package

⚠ 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.

9.15.9 mhchem package

See section 304.

9.15.10 xparse package

Pkg xparse

To remove from the log any warnings about redeclaring objects, place the following

before lwarp is loaded:

```
\usepackage[log-declarations=false]{xparse}
```

9.15.11 kotex package

Pkg kotex

See section 9.14 regarding *pdflatex* and Korean section names.

⚠ Korean section names

10 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.

10.1 Command options

Opt PrintLatexCmd
Opt HTMLLatexCmd

The lwarp options `PrintLatexCmd` and `HTMLLatexCmd` are used to set customized commands to be executed by **`lwarpmk print`** and **`lwarpmk html`**.

`PrintLatexCmd` should be set to shell commands which take `project.tex` and generate `project.pdf`.

`HTMLLatexCmd` should be set to take `project_html.tex` and generate `project_html.pdf`. *lwarpmk* will then take `project_html.pdf` and automatically convert it and generate `project.html`.

10.2 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 7. `\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 7: 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 tm-pics.ps \LWRopseq
          pdflatex tm.tex
      } ,
  HTMLLatexCmd=
      {
          latex tm_html \LWRopseq
          dvips -o tm_html-pics.ps tm_html.dvi \LWRopseq
          ps2pdf tm_html-pics.ps \LWRopseq
          pdflatex tm_html.tex
      }
}
```

10.3 *latexmk*

Prog latexmk 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.

Pkg sagetex *latexmk* may simplify the use of packages such as sagetex.

10.4 perltex package

Pkg perltex 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 9.7.9.

10.5 pythontex package

Pkg pythontex 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.

⚠ "impure" math
⚠ HTML look-alike

No attempt has yet been made to make pythontex robust with HTML output. Some math objects must be surrounded by `\displaymathother` ... `\displaymathnormal`, or `\inlinemathother` ... `\inlinemathnormal`. Displays of code may have to be enclosed inside a `lateximage` environment to prevent <, > and similar from being interpreted by the browser as HTML entities.

10.6 Other packages

Pkg sympytex Other packages such as sympytex and rterface would be set up similar to pythontex,

Pkg rterface

and the same warnings would apply.

10.7 *make* program

Prog make

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`**.

10.8 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
  }
]
```

Pkg ditaa

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.

11 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`

Bool `FormatEPUB`
Default: `false`

`FormatEPUB` changes HTML output for easy EPUB conversion via an 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.

Calibre

The resulting files will be ready to be loaded into an EPUB conversion program, 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 9. 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.

12 Word-processor conversion

lwarp may be told to modify its HTML output to make it easier to import the HTML document into a word processor. At the time of this writing, it seems that LibreOffice works best at preserving table layout, but it still has some limitations, such as an inability to automatically assign figure and table frames and captions according to user-selected HTML classes. lwarp provides some assistance in locating these frame boundaries, as shown below.

12.1 Activating word-processor conversion

A special boolean is provided to simplify the process of converting lwarp HTML output to EPUB:

`FormatWP`

Bool `FormatWP`
Default: `false`

Changes HTML output for easier conversion by a word processor. 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}
```

to the project's source preamble after lwarp is loaded. The following changes are then made to the HTML output:

formatting adjustments

- 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 `\quad`s, 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.

12.2 Additional modifications

`WPMarkFloats`

Bool `WPMarkFloats`
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

Bool WPMarkMinipages
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

Bool WPMarkTOC
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

Bool WPMarkLOFT
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
Bool WPMarkMath
Default: false
Prog TeXMaths

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.

Table 8: 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
\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 9 on page 189.

WPTitleHeading

Bool WPTitleHeading
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**.

See table 8 on page 175.

12.3 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.

12.4 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.

13 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 87 for an example of the `picture` environment.

css classes

To create a custom HTML block or inline CSS class, see section 51.9.

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.

index recreation

To recreate the index for the lwarp documentation:

```
makeindex -s gglo.ist -o lwarp.gls lwarp.glo
splitindex lwarp.idx -- -s gind.ist
```

13.1 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 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:

```
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}
```

13.2 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 331 (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.

13.2.1 Adding a package to the lwarp.dtx file

When adding a package to lwarp.dtx for permanent inclusing 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 21.1. 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.

13.3 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.

13.4 Testing lwarp

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.

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`
3. `pdflatex lwarp.dtx`
4. If modifying *lwarpmk* the new version should now be active.
5. If modifying CSS files:
   (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.

13.5 Modifying *lwarpmk*

Prog `lwarpmk`
File `lwarpmk.lua`

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`.

14 Troubleshooting

14.1 Using the lwarp.sty package

Also see:
Section 8.8: Commands to be placed into the `warpprint` environment
Section 9: Special cases and limitations

Text is not converting correctly / corrupted HTML tags:

- Font-related UTF-8 information must be embedded in the PDF file. See section 8.2 regarding bitmapped vs. vector fonts.
- See section 9.2.1 regarding HTML entities and the characters &, <, and >.

Undefined HTML settings:

- See the warning regarding the placement of the HTML settings at section 8.4.

Tabular problems: See section 9.10.

Obscure error messages:

Print first: Be sure that a print version of the document compiles and that your document's LaTeX code is correct, before attempting to generate an HTML version.

`\end{warpHTML}, \end{warpprint}, \end{warpall}`: Each of these must be without any other characters on the same line.

Options clash: If using memoir, see section 9.13.

"No room for a new `\write`.": Before `\usepackage{lwarp}`, add:

```
\usepackage{morewrites}
\morewritessetup{allocate=10}
```

"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
```

"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.

⚠ 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 9.6.9.

⚠ polyglossia

"`! Undefined control sequence. ... \@begindocumenthook''`: See section 9.15.4 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`.

⚠ "impure" math objects

Complicated objects inside math: Some objects, such as Ti*k*z, 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 9.7.9.

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

⚠ 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 9.7.9.

Missing sections: See section 8.4 regarding the `FileDepth` and `SideTOCDepth` counters, and the use of `\tableofcontents` in the home page.

Misnumbered footnotes from section headings: See section 9.5.4.

Missing HTML files:

- See the warning regarding changes to the HTML settings at section 8.4.
- Ensure that the filenames are unique after math and short words are removed. See `FileSectionNames` at section 8.4.

Missing / incorrect cross-references:

- Use `lwarpmk again` followed by `lwarpmk html` or `lwarpmk print` to compile the document one more time.

labels

⚠ label characters

`\nameref`

⚠ empty link

- 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` 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 529 to redefine the message which is printed for page number references.

BibTeX errors with `\etalchar`: See section 9.6.9.

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 XeLaTeX 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:

- Enter **`lwarpmk limages`** to refresh the `lateximage` images.

SVG images:

⚠ 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.

Before attempting to create the SVG image files, *lwarpmk* verifies that the HTML version of the document exists and has correct internal image references.[15] If it is necessary to recompile the document's HTML version one more time, *lwarpmk* will inform the user with an error message.

⚠ 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. *lwarpmk* attempts to detect this situation and print a warning.

⚠ 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

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

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 6.5 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.

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:

- 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 8.4.
- 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.

14.1.1 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.

14.2 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}`

15 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.

File 1 **lwarp.sty**

16 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 9: Section depths and HTML headings

Section	LaTeX depth	HTML headings *
title of the entire website		`<h1>`
none	-5	new for this package
book	-2	**not yet used**
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 8 on page 175.

17 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 9. "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.

18 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.

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

index entries

⚠ warnings — Special warnings are marked with a warning icon.

for HTML output: **for PRINT output:** **for HTML & PRINT:** — Green-colored tags in the left margin show which sections of source code apply to the generation of HTML, print, or both forms of output.

— lwarp source code begins on the following page —

19 Detecting the TEX engine — *pdflatex*, *lualatex*, *xelatex*

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

Detects XƎTEX and LuaLATEX:

```
\RequirePackage{iftex}
\newif\ifxetexorluatex
\ifXeTeX
    \xetexorluatextrue
\else
    \ifLuaTeX
        \xetexorluatextrue
    \else
        \xetexorluatexfalse
    \fi
\fi

\ifLuaTeX
\RequirePackage{luatex85}% until the geometry package is updated
\fi

\RequirePackage{ifpdf}

\RequirePackage{ifptex}
```

20 Early package requirements

Pkg etoolbox Provides `\ifbool` and other functions.

Pkg xpatch Patches macros with optional arguments.

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

Pkg ifplatform Provides `\ifwindows` to try to automatically detect WINDOWS OS.

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

Pkg letltxmacro

```
\RequirePackage{letltxmacro}
```

21 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.

21.1 Tests of package load order

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

```
\newcommand*{\LWR@loadafter}[1]{%
\@ifpackageloaded{#1}
{
    \PackageError{lwarp}
        {Package #1, or one which uses #1, must be loaded after lwarp}
       {Move \detokenize{\usepackage}{#1} after \detokenize{\usepackage}{lwarp}.
       Package #1 may also be loaded by something else, which must also be moved
        after lwarp.}
}
{}
}
```

\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.

```
\@ifclassloaded{memoir}
{\newcommand*{\LWR@notmemoirloadafter}[1]{}}
{\LetLtxMacro\LWR@notmemoirloadafter\LWR@loadafter}
```

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

```
\LetLtxMacro\LWR@notltjloadafter\LWR@loadafter

\@ifclassloaded{ltjarticle}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
\@ifclassloaded{ltjbook}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
\@ifclassloaded{ltjreport}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
\@ifclassloaded{ltjsarticle}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
\@ifclassloaded{ltjsbook}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
\@ifclassloaded{ltjsreport}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
\@ifclassloaded{ltjspf}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
\@ifclassloaded{ltjskiyou}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
\@ifclassloaded{ltjtarticle}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
\@ifclassloaded{ltjtbook}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
```

```
\@ifclassloaded{ltjtreport}{\renewcommand*{\LWR@notltjloadafter}[1]{}}{}
```

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

```
\newcommand*{\LWR@loadbefore}[1]{%
\@ifpackageloaded{#1}
{}
{
\PackageError{lwarp}
{Package #1 must be loaded before lwarp}
{Move \detokenize{\usepackage}{#1} before \detokenize{\usepackage}{lwarp}.}
}
}
```

`\LWR@checkloadbefore` {⟨*thispackagename*⟩} {⟨*packagename*⟩}

If package names match, error if it is loaded after lwarp.

```
\newcommand*{\LWR@checkloadbefore}[2]{%
    \edef\LWR@tempone{#1}%
    \ifdefstring{\LWR@tempone}{#2}{%
        \LWR@loadbefore{#1}%
    }{}%
}
```

`\LWR@loadnever` {⟨*badpackagename*⟩} {⟨*replacementpkgnames*⟩}

The first packages is not supported, so tell the user to use the second instead.

```
\newcommand*{\LWR@loadnever}[2]{%
\PackageError{lwarp}
{Package #1 is not supported by lwarp's HTML conversion.
Package(s) #2 may be useful instead}
{Package #1 might conflict with lwarp in some way,
or is superceded by another package.
For a possible alternative, see package(s) #2.}
}
```

`\LWR@checkloadnever` {⟨*thispackagename*⟩} {⟨*badpackagename*⟩} {⟨*replacementpkgnames*⟩}

If this package name is the bad packagename, suggest the replacements instead.

```
\newcommand*{\LWR@checkloadnever}[3]{%
    \edef\LWR@tempone{#1}%
    \ifdefstring{\LWR@tempone}{#2}{%
        \LWR@loadnever{#2}{#3}%
    }{}%
}
```

`\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.

```
\newcommand*{\LWR@earlyloadnever}[2]{%
\@ifpackageloaded{#1}{%
\PackageError{lwarp}
{Package #1 is not supported by lwarp's HTML conversion.
Package(s) #2 may be useful instead}
{Package #1 might conflict with lwarp in some way,
or is superceded by another package.
For a possible alternative, see package(s) #2.}
}{}%
}
```

`\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.

```
\newcommand*{\LWR@earlyclassloadnever}[2]{%
\@ifclassloaded{#1}{%
\PackageError{lwarp}
{Class #1 is not supported by lwarp's HTML conversion.
Class(es) #2 may be useful instead}
{Class #1 might conflict with lwarp in some way,
or is superceded by another class.
For a possible alternative, see class(es) #2.}
}{}%
}
```

21.2 Error for disallowed packages and classes loaded before lwarp

```
\LWR@earlyclassloadnever{jarticle}{ujarticle}
\LWR@earlyclassloadnever{jbook}{ujbook}
\LWR@earlyclassloadnever{jreport}{ujreport}
\LWR@earlyclassloadnever{tarticle}{utarticle}
\LWR@earlyclassloadnever{tbook}{utbook}
\LWR@earlyclassloadnever{treport}{utreport}
\LWR@earlyloadnever{ae}{cm-super, lmodern}
\LWR@earlyloadnever{aecompl}{cm-super, lmodern}
\LWR@earlyloadnever{aecc}{cm-super, lmodern}
\LWR@earlyloadnever{boxedminipage}{boxedminipage2e}
\LWR@earlyloadnever{caption2}{caption}
% \LWR@earlyloadnever{ccaption}{caption}% might be preloaded by memoir
```

The older CJK and CJKutf8 only work with xeCJK:

```
\@ifpackageloaded{xeCJK}{}{
    \LWR@earlyloadnever{CJK}{ctex, xeCJK}
    \LWR@earlyloadnever{CJKutf8}{ctex, xeCJK}
}
```

bxcjkatype is based on CJK:

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

hangul is not in TeXLive, and is not tested:

```
\LWR@earlyloadnever{hangul}{kotex, xetexko, luatexko}
```

Others:

```
\LWR@earlyloadnever{colortab}{colortbl}
\LWR@earlyloadnever{epsf}{graphicx}
\LWR@earlyloadnever{fancyheadings}{fancyhdr}
\LWR@earlyloadnever{glossary}{glossaries}
\LWR@earlyloadnever{hyper}{hyperref}
\LWR@earlyloadnever{picinpar}{floatflt, wrapfig}
\LWR@earlyloadnever{picins}{floatflt, wrapfig}
\LWR@earlyloadnever{sistyle}{siunitx}
\LWR@earlyloadnever{t1enc}{fontenc, inputenc, inputenx}
\LWR@earlyloadnever{ucs}{inputenc, inputencx}
\LWR@earlyloadnever{wasysym}{textcomp, amssymb, amsfonts, mnsymbol, fdsymbol}
```

21.3 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:

```
\LWR@loadafter{2in1}
\LWR@loadafter{2up}
\LWR@loadafter{a4}
\LWR@loadafter{a4wide}
\LWR@loadafter{a5comb}
\LWR@notmemoirloadafter{abstract}
\LWR@loadafter{academicons}
\LWR@loadafter{accsupp}
\LWR@loadafter{acro}
\LWR@loadafter{acronym}
\LWR@loadafter{adjmulticol}
\LWR@loadafter{addlines}
\LWR@loadafter{ae}
\LWR@loadafter{aecc}
\LWR@loadafter{afterpage}
\LWR@loadafter{algorithm2e}
\LWR@loadafter{algorithmicx}
\LWR@loadafter{alltt}
\LWR@loadafter{amsmath}
\LWR@loadafter{amsthm}
\LWR@loadafter{anonchap}
\LWR@loadafter{anysize}
\LWR@notmemoirloadafter{appendix}
\LWR@loadafter{ar}
\LWR@loadafter{arabicfront}
\LWR@notmemoirloadafter{array}
```

```
\LWR@loadafter{arydshln}
\LWR@loadafter{asymptote}
% \LWR@loadafter{atbegshi}% used by morewrites
\LWR@loadafter{attachfile}
\LWR@loadafter{attachfile2}
\LWR@loadafter{authblk}
\LWR@loadafter{axessibility}
\LWR@loadafter{axodraw2}
\LWR@loadafter{backref}
\LWR@loadafter{balance}
\LWR@loadafter{bbding}
\LWR@loadafter{bigdelim}
\LWR@loadafter{bigstrut}
\LWR@loadafter{bitpattern}
\LWR@loadafter{blowup}
\LWR@loadafter{booklet}
\LWR@loadafter{bookmark}
\LWR@notmemoirloadafter{booktabs}
\LWR@loadafter{bophook}
\LWR@loadafter{bounddvi}
\LWR@loadafter{boxedminipage}
\LWR@loadafter{boxedminipage2e}
\LWR@loadafter{breakurl}
\LWR@loadafter{breqn}
\LWR@loadafter{bsheaders}
\LWR@loadafter{bxpapersize}
\LWR@loadafter{bytefield}
\LWR@loadafter{cancel}
\LWR@loadafter{canoniclayout}
\LWR@loadafter{caption}
\LWR@loadafter{caption2}
\LWR@loadafter{cases}
% \LWR@loadafter{ccaption}% may be preloaded by memoir
\LWR@loadafter{changebar}
\LWR@notmemoirloadafter{changepage}
\LWR@loadafter{changes}
\LWR@loadafter{chappg}
\LWR@loadafter{chapterbib}
\LWR@loadafter{chemfig}
\LWR@loadafter{chemformula}
\LWR@loadafter{chemgreek}
\LWR@loadafter{chemmacros}
\LWR@loadafter{chemnum}
\LWR@loadafter{chkfloat}
\LWR@notmemoirloadafter{chngpage}
\LWR@loadafter{cite}
\LWR@loadafter{cmdtrack}
\LWR@loadafter{color}
\LWR@loadafter{colortbl}
\LWR@loadafter{continue}
\LWR@loadafter{copyrightbox}
\LWR@notmemoirloadafter{crop}
% ctex must be loaded before lwarp
\LWR@loadafter{cuted}
\LWR@loadafter{cutwin}
```

```
\LWR@loadafter{dblfloatfix}
\LWR@loadafter{dblfnote}
\LWR@notmemoirloadafter{dcolumn}
\LWR@loadafter{diagbox}
\LWR@loadafter{dingbat}
\LWR@loadafter{dprogress}
\LWR@loadafter{draftcopy}
\LWR@loadafter{draftfigure}
\LWR@loadafter{draftwatermark}
\LWR@loadafter{easy-todo}
\LWR@loadafter{ebook}
\LWR@loadafter{ed}
\LWR@loadafter{ellipsis}
\LWR@loadafter{embrac}
\LWR@loadafter{emptypage}
\LWR@loadafter{endfloat}
\LWR@loadafter{endheads}
\LWR@loadafter{endnotes}
\LWR@notmemoirloadafter{enumerate}
\LWR@loadafter{enumitem}
\LWR@notmemoirloadafter{epigraph}
\LWR@loadafter{epsfig}
\LWR@loadafter{epstopdf}
\LWR@loadafter{epstopdf-base}
\LWR@loadafter{errata}
\LWR@loadafter{eso-pic}
\LWR@loadafter{eurosym}
\LWR@loadafter{everypage}
\LWR@loadafter{everyshi}
\LWR@loadafter{extramarks}
\LWR@loadafter{fancybox}
\LWR@loadafter{fancyhdr}
\LWR@loadafter{fancyheadings}
\LWR@loadafter{fancyref}
\LWR@loadafter{fancytabs}
\LWR@loadafter{fancyvrb}
\LWR@loadafter{figcaps}
\LWR@loadafter{figsize}
\LWR@loadafter{fitbox}
\LWR@loadafter{fix2col}
\LWR@loadafter{fixme}
\LWR@loadafter{fixmetodonotes}
\LWR@loadafter{flafter}
\LWR@loadafter{flippdf}
\LWR@loadafter{float}
\LWR@loadafter{floatflt}
\LWR@loadafter{floatpag}
\LWR@loadafter{floatrow}
\LWR@loadafter{fltrace}
\LWR@loadafter{flushend}
\LWR@loadafter{fnbreak}
\LWR@loadafter{fncychap}
\LWR@loadafter{fnlineno}
\LWR@loadafter{fnpos}
\LWR@loadafter{fontawesome}
```

```
\LWR@loadafter{fontawesome5}
% fontenc must be loaded before lwarp
% fontspec must be loaded before lwarp
\LWR@loadafter{footmisc}
\LWR@loadafter{footnote}
\LWR@loadafter{footnotehyper}
\LWR@loadafter{footnoterange}
\LWR@loadafter{footnpag}
\LWR@loadafter{foreign}
\LWR@loadafter{forest}
\LWR@loadafter{framed}
\LWR@loadafter{ftnright}
\LWR@loadafter{fullminipage}
\LWR@loadafter{fullpage}
\LWR@loadafter{fullwidth}
\LWR@loadafter{fwlw}
\LWR@loadafter{gentombow}
% geometry is always loaded by lwarp, and lwarp-geometry is AtBeginDocument
\LWR@loadafter{gmeometric}
\LWR@loadafter{glossaries}
% \LWR@loadafter{graphics}% pre-loaded by xunicode
% \LWR@loadafter{graphicx}% pre-loaded by xunicode
\LWR@loadafter{gloss}
\LWR@loadafter{glossary}
\LWR@loadafter{grffile}
\LWR@loadafter{grid}
\LWR@loadafter{grid-system}
\LWR@loadafter{gridset}
\LWR@loadafter{hang}
\LWR@loadafter{hanging}
\LWR@loadafter{hypcap}
\LWR@loadafter{hypdestopt}
\LWR@loadafter{hypernat}
\LWR@loadafter{hyperref}
\LWR@loadafter{hyperxmp}
\LWR@loadafter{hyphenat}
\LWR@loadafter{idxlayout}
\LWR@loadafter{ifoddpage}
\LWR@loadafter{imakeidx}
\LWR@notmemoirloadafter{index}
% inputenc must be loaded before lwarp
% inputenx must be loaded before lwarp
\LWR@loadafter{intopdf}
\LWR@loadafter{karnaugh-map}
\LWR@loadafter{keyfloat}
\LWR@loadafter{layaureo}
\LWR@loadafter{layout}
\LWR@loadafter{leading}
\LWR@loadafter{letterspace}
\LWR@loadafter{lettrine}
\LWR@loadafter{lineno}
\LWR@loadafter{lips}
\LWR@loadafter{listings}
\LWR@loadafter{longtable}
\LWR@loadafter{lscape}
```

```
\LWR@loadafter{ltablex}
\LWR@loadafter{ltcaption}
\LWR@loadafter{ltxgrid}
\LWR@loadafter{ltxtable}
\LWR@loadafter{lua-check-hyphen}
\LWR@loadafter{lua-visual-debug}
\LWR@loadafter{luacolor}
\LWR@loadafter{luatodonotes}
\LWR@loadafter{magaz}
\LWR@notmemoirloadafter{makeidx}
\LWR@loadafter{marginfit}
\LWR@loadafter{marginfix}
\LWR@loadafter{marginnote}
\LWR@loadafter{marvosym}
\LWR@loadafter{mcaption}
\LWR@loadafter{mdframed}
\LWR@loadafter{memhfixc}
\LWR@loadafter{metalogo}
\LWR@loadafter{mhchem}
\LWR@loadafter{microtype}
\LWR@loadafter{midfloat}
\LWR@loadafter{midpage}
\LWR@loadafter{morefloats}
\LWR@notmemoirloadafter{moreverb}
% morewrites must be loaded before lwarp
\LWR@notmemoirloadafter{mparhack}
\LWR@loadafter{multicap}
%\LWR@loadafter{multicol}% loaded by ltxdoc
\LWR@loadafter{multicolrule}
\LWR@loadafter{multirow}
\LWR@loadafter{multitoc}
\LWR@loadafter{musicography}
\LWR@loadafter{nameauth}
\LWR@loadafter{nameref}
\LWR@loadafter{natbib}
\LWR@notmemoirloadafter{nccfancyhdr}
\LWR@notmemoirloadafter{needspace}
% newclude must be loaded before lwarp
\LWR@loadafter{newtxmath}
% newunicodechar must be loaded before lwarp
\LWR@notmemoirloadafter{nextpage}
\LWR@loadafter{nicefrac}
\LWR@loadafter{nomencl}
\LWR@loadafter{nonfloat}
\LWR@loadafter{nonumonpart}
\LWR@loadafter{nopageno}
\LWR@loadafter{notes}
\LWR@loadafter{notespages}
\LWR@loadafter{nowidow}
\LWR@loadafter{ntheorem}
\LWR@loadafter{octave}
\LWR@loadafter{overpic}
\LWR@loadafter{pagegrid}
\LWR@notmemoirloadafter{pagenote}
\LWR@loadafter{pagesel}
```

```
\LWR@loadafter{paralist}
\LWR@loadafter{parnotes}
\LWR@notmemoirloadafter{parskip}
\LWR@loadafter{pbox}
\LWR@loadafter{pdfcomment}
\LWR@loadafter{pdflscape}
\LWR@loadafter{pdfmarginpar}
\LWR@loadafter{pdfpages}
\LWR@loadafter{pdfprivacy}
\LWR@loadafter{pdfrender}
\LWR@loadafter{pdfsync}
\LWR@loadafter{pdftricks}
\LWR@loadafter{pdfx}
\LWR@loadafter{pfnote}
\LWR@loadafter{phfqit}
\LWR@loadafter{pifont}
\LWR@loadafter{placeins}
\LWR@loadafter{plarray}
\LWR@loadafter{plarydshln}
\LWR@loadafter{plextarray}
\LWR@loadafter{plextarydshln}
\LWR@loadafter{plcolortbl}
\LWR@loadafter{plextdelarray}
\LWR@loadafter{prelim2e}
\LWR@loadafter{prettyref}
\LWR@loadafter{preview}
\LWR@loadafter{psfrag}
\LWR@loadafter{psfragx}
\LWR@loadafter{pst-eps}
\LWR@loadafter{pstool}
\LWR@loadafter{pstricks}
% \LWR@loadafter{pxatbegshi}% may be used by morewrites
\LWR@loadafter{pxeveryshi}
\LWR@loadafter{pxftnright}
\LWR@loadafter{pxjahyper}
\LWR@loadafter{quotchap}
\LWR@loadafter{quoting}
\LWR@loadafter{ragged2e}
\LWR@loadafter{realscripts}
\LWR@loadafter{refcheck}
\LWR@loadafter{register}
\LWR@loadafter{relsize}
\LWR@loadafter{repeatindex}
\LWR@loadafter{resizegather}
\LWR@loadafter{rmpage}
\LWR@loadafter{romanbar}
\LWR@loadafter{romanbarpagenumber}
\LWR@loadafter{rotating}
\LWR@loadafter{rotfloat}
\LWR@loadafter{rviewport}
\LWR@loadafter{savetrees}
% scalefnt is loaded by babel-french
\LWR@loadafter{schemata}
\LWR@loadafter{scrextend}
\LWR@loadafter{scrhack}
```

```
\LWR@loadafter{scrlayer}
\LWR@loadafter{scrlayer-notecolumn}
\LWR@loadafter{scrlayer-scrpage}
\LWR@loadafter{scrpage2}
\LWR@loadafter{section}
\LWR@loadafter{sectionbreak}
\LWR@loadafter{sectsty}
\LWR@loadafter{semantic-markup}
\LWR@notmemoirloadafter{setspace}
\LWR@loadafter{shadow}
\LWR@notmemoirloadafter{showidx}
\LWR@loadafter{showkeys}
\LWR@loadafter{sidecap}
\LWR@loadafter{sidenotes}
\LWR@loadafter{SIunits}
\LWR@loadafter{siunitx}
\LWR@loadafter{soul}
\LWR@loadafter{soulpos}
\LWR@loadafter{soulutf8}
\LWR@loadafter{splitidx}
\LWR@loadafter{srcltx}
\LWR@loadafter{srctex}
\LWR@loadafter{stabular}
\LWR@notltjloadafter{stfloats}
\LWR@loadafter{struktex}
\LWR@loadafter{subcaption}
\LWR@loadafter{subfig}
\LWR@loadafter{subfigure}
\LWR@loadafter{supertabular}
\LWR@loadafter{t1inc}
\LWR@loadafter{tabls}
\LWR@notmemoirloadafter{tabularx}
\LWR@loadafter{tabulary}
\LWR@loadafter{tascmac}
\LWR@loadafter{textarea}
% \LWR@loadafter{textcomp}% maybe before lwarp with font packages
\LWR@loadafter{textfit}
\LWR@loadafter{textpos}
\LWR@loadafter{theorem}
\LWR@loadafter{thinsp}
\LWR@loadafter{threadcol}
\LWR@loadafter{threeparttable}
\LWR@loadafter{thumb}
\LWR@loadafter{thumbs}
\LWR@loadafter{tikz}
\LWR@loadafter{titleps}
\LWR@loadafter{titlesec}
\LWR@loadafter{titletoc}
\LWR@notmemoirloadafter{titling}
% \LWR@loadafter{tocbasic}% preloaded by koma-script classes
\LWR@notmemoirloadafter{tocbibind}
\LWR@loadafter{tocenter}
\LWR@notmemoirloadafter{tocloft}
\LWR@loadafter{tocstyle}
\LWR@loadafter{todo}
```

```
\LWR@loadafter{todonotes}
\LWR@loadafter{tram}
\LWR@loadafter{transparent}
\LWR@loadafter{trimclip}
\LWR@loadafter{trivfloat}
\LWR@loadafter{truncate}
\LWR@loadafter{turnthepage}
\LWR@loadafter{twoup}

% \LWR@loadafter{typearea}% preloaded by koma-script classes
\LWR@loadafter{typicons}
% \LWR@loadafter{ulem}% preloaded by ctexart and related classes
\LWR@loadafter{umoline}
\LWR@loadafter{underscore}
\LWR@loadafter{units}
\LWR@loadafter{unitsdef}
\LWR@loadafter{upref}
\LWR@loadafter{url}
\LWR@loadafter{uspace}
\LWR@loadafter{varioref}% no lwarp package provided
\LWR@notmemoirloadafter{verse}
\LWR@loadafter{versonotes}
\LWR@loadafter{vertbars}
\LWR@loadafter{vmargin}
\LWR@loadafter{vowel}
\LWR@loadafter{vpe}
\LWR@loadafter{vwcol}
\LWR@loadafter{wallpaper}
\LWR@loadafter{watermark}
\LWR@loadafter{widows-and-orphans}
\LWR@loadafter{wrapfig}
\LWR@loadafter{xbmks}
\LWR@loadafter{xcolor}
\LWR@loadafter{xechangebar}
\LWR@loadafter{xellipsis}
% xetexko-vertical must be loaded before lwarp
\LWR@loadafter{xfakebold}
\LWR@loadafter{xfrac}
\LWR@loadafter{xltabular}
\LWR@loadafter{xltxtra}
\LWR@loadafter{xmpincl}
\LWR@loadafter{xpiano}
\LWR@loadafter{xpinyin}
\LWR@loadafter{xtab}
% xunicode must be loaded before lwarp
\LWR@loadafter{xurl}
\LWR@loadafter{xy}
\LWR@loadafter{zwpagelayout}
```

22 MD5 hashing

The MD5 hash is used for `lateximage` filenames for SVG math.

```
\newcommand{\LWR@mdfive}[1]{%
\PackageError{lwarp}
{No MD5 macro was found.}
{Lwarp must find the macros pdfmdfivesum or mdfivesum.}
}
```

The default for pdfLaTeX, DVILaTeX, upLaTeX, etc:

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

For LuaLaTeX:

```
\ifLuaTeX
\RequirePackage{pdftexcmds}
\let\LWR@mdfive\pdf@mdfivesum
\fi
```

For XeLaTeX:

```
\ifXeTeX
\@ifundefined{pdffivesum}{}
    {\let\LWR@mdfive\pdfmdfivesum}
\@ifundefined{mdfivesum}{}
    {\let\LWR@mdfive\mdfivesum}
\fi
```

23 pdfLaTeX T1 and UTF-8 encoding

When using pdfLaTeX, lwarp requires T1 encoding, and recommends UTF-8 encoding.

If some other input encoding is already defined, lwarp will try to use it instead, and hope for the best.

XeLaTeX and LuaLaTeX are both UTF-8 by nature.

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

```
\newcommand*{\LWR@pdfencoding}{%
    \RequirePackage[T1]{fontenc}

    \@ifpackageloaded{inputenc}{}{
        \@ifpackageloaded{inputenx}{}{
            \RequirePackage[utf8]{inputenc}
        }
    }
}
```

```
\ifPDFTeX% pdflatex or dvi latex
    \LWR@pdfencoding
\fi

\ifpTeX
    \LWR@pdfencoding
\fi
```

24 Unicode input characters

for HTML & PRINT:

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.

```
\ifpTeX
\else
\RequirePackage{newunicodechar}

\newunicodechar{×}{\texttimes}

\ifPDFTeX% pdflatex or dvi latex
\newunicodechar{ff}{ff}% the first arguments are ligatures
\newunicodechar{fi}{fi}
\newunicodechar{fl}{fl}
\newunicodechar{ffi}{ffi}
\newunicodechar{ffl}{ffl}
\newunicodechar{---}{---}
\newunicodechar{--}{--}
\fi

\fi
```

25 Upright quotes

In PDFTEX, preserve upright quotes in verbatim text. upquote also loads textcomp.

```
\ifPDFTeX
\RequirePackage{upquote}
\fi

\ifpTeX
    \RequirePackage{upquote}
\fi
```

26 Miscellaneous tools

\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.

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

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

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

```
\newcommand*{\LWR@convertto}[2]{\strip@pt\dimexpr #2*65536/\number\dimexpr 1#1}
```

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

Prints an error if could not patch a macro.

```
\newcommand*{\LWR@patcherror}[2]{%
\PackageError{lwarp}
{Unable to patch package #1, macro #2}
{Please contact the author of the lwarp package.}
}
```

\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.

```
\newcommand{\LWR@isolate}[1]{#1}%

\@ifpackageloaded{ctexpatch}{
    \renewcommand{\LWR@isolate}[1]{\null#1\null}%
}{}

\@ifpackageloaded{xeCJK}{
    \renewcommand{\LWR@isolate}[1]{\null#1\null}%
}{}
```

\LWR@firstoffour {⟨*first*⟩} {⟨*second*⟩} {⟨*third*⟩} {⟨*fourth*⟩}

\LWR@secondoffour {⟨*first*⟩} {⟨*second*⟩} {⟨*third*⟩} {⟨*fourth*⟩}

\LWR@thirdoffour {⟨*first*⟩} {⟨*second*⟩} {⟨*third*⟩} {⟨*fourth*⟩}

\LWR@fourthoffour {⟨*first*⟩} {⟨*second*⟩} {⟨*third*⟩} {⟨*fourth*⟩}

Expands to the nth of the four arguments. Used for extra cross referencing.

```
\long\def\LWR@firstoffour#1#2#3#4{#1}
\long\def\LWR@secondoffour#1#2#3#4{#2}
\long\def\LWR@thirdoffour#1#2#3#4{#3}
\long\def\LWR@fourthoffour#1#2#3#4{#4}
```

27 Operating-System portability

Prog Unix
Prog Mac OS
Prog Linux
Prog MS-Windows
Prog Windows
Opt OSWindows

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.

27.1 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:

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

The literal `$` character:

```
\catcode`\$=12
\def\LWRdollar{$}
\catcode`\$=3
```

The literal `&` character:

```
\catcode`\&=12
\def\LWRamp{&}
\catcode`\&=4
```

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

```
\catcode`\&=0
&catcode`&\=12
&def&LWRbackslash{\}
&catcode`&\=0
\catcode`\&=4
```

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

```
\catcode‘\&=1
\catcode‘\{=12
\def\LWRleftbrace&{}
\catcode‘\{=1
\catcode‘\&=4
```

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

```
\catcode‘\&=2
\catcode‘\}=12
\def\LWRrightbrace{}&
\catcode‘\}=2
\catcode‘\&=4
```

The literal # character:

```
\catcode‘\#=12
\def\LWRhash{#}
\catcode‘\#=6
```

`\LWRopquote` The operating system's quote mark, UNIX default. For WINDOWS, see `\LWR@setOSWindows`, below.

```
\def\LWRopquote{’}
```

`\LWRopseq` The operating system's sequential execution command, UNIX default. For WINDOWS, see `\LWR@setOSWindows`, below.

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

27.2 Common portability code

Bool `usingOSWindows` Set if the `OSWindows` option is used, or if WINDOWS is automatically detected.

```
\newbool{usingOSWindows}
\boolfalse{usingOSWindows}
```

27.3 UNIX, LINUX, and MAC OS

`\OSPathSymbol` Symbol used to separate directories in a path.

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

27.4 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.

```
\newcommand*{\LWR@setOSWindows}
{
\booltrue{usingOSWindows}
\renewcommand*{\OSPathSymbol}{\@backslashchar}
\def\LWRopquote{"}
\def\LWRopseq{\space\LWRamp\space\space}
}
```

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

```
\ifwindows
\LWR@setOSWindows
\fi
```

28 Package options

Pkg kvoptions Allows key/value package options.

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

\lwarpsetup A user interface to set the keys:

```
\newcommand{\lwarpsetup}[1]{\setkeys{LWR}{#1}}
```

Bool warpingprint
Bool warpingHTML
Bool mathjax
Bool LWR@origmathjax

Set to true/false depending on the package option selections for print/HTML/EPUB output and mathsvg/mathjax.

LWR@origmathjax remembers the original setting to be restored by \displaymathnormal.

```
\newbool{warpingprint}
\newbool{warpingHTML}
\newbool{mathjax}
\newbool{LWR@origmathjax}
```

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

```
\booltrue{warpingprint}%
\boolfalse{warpingHTML}%
\boolfalse{mathjax}%
```

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

```
\DeclareVoidOption{warpprint}{%
\PackageInfo{lwarp}{Using option 'warpprint'}
\booltrue{warpingprint}%
\boolfalse{warpingHTML}%
}
```

Opt warpHTML Anything in the warpHTML environment will be generated for HTML output only.

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

```
\DeclareVoidOption{warpHTML}{%
\PackageInfo{lwarp}{Using option 'warpHTML'}%
\booltrue{warpingHTML}%
\boolfalse{warpingprint}%
}
```

Opt mathsvg Option mathsvg selects SVG math display: If the mathsvg option is given, boolean mathjax is false, and may be used for \ifbool tests.

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

}
```

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

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

}
```

Opt BaseJobname 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.

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

Opt ImagesDirectory
Default: \jobname-images

Option ImagesDirectory sets the name of the directory to use for the lateximage images.

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

Opt ImagesName
Default: image-

Option ImagesName sets the prefix to use for the lateximage images.

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

Opt makeindexStyle
Default: lwarp.ist

Selects a custom .ist file. A customized file should be based on lwarp.ist, and must retain the lines related to \hyperindexref.

```
\DeclareStringOption[lwarp.ist]{makeindexStyle}
```

Opt xindyStyle
Default: lwarp.xdy

Selects a custom .xdy file. A customized file should be based on lwarp.xdy, and must retain the line

```
(markup-locref :open "\hyperindexref{" :close "}")
```

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

Opt xindyLanguage
Default: english

Sets the xindy language to be assigned in *lwarpmk*'s configuration files. This is then used by *lwarpmk* while processing the index and glossary.

```
\DeclareStringOption[english]{xindyLanguage}
```

Opt xindyCodepage
Default: utf8

Sets the xindy codepage to be assigned in *lwarpmk*'s configuration files. This is then used by *lwarpmk* while processing the index.

```
\DeclareStringOption[utf8]{xindyCodepage}
```

Opt pdftotextEnc
Default: UTF-8

The option pdftotextEnc sets the encoding used by *pdftotext*. This is passed to *pdftotext* using its -enc option, and is used when converting LaTeX PDF output with HTML tags into a plain-text file with HTML tags.

```
\DeclareStringOption[UTF-8]{pdftotextEnc}
```

Opt lwarpmk

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:

```
\newbool{LWR@creatinglwarpmk}
\boolfalse{LWR@creatinglwarpmk}

\DeclareVoidOption{lwarpmk}{
\PackageInfo{lwarp}{Using option 'lwarpmk'}
\booltrue{LWR@creatinglwarpmk}
}
```

Opt OSWindows

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.

```
\DeclareVoidOption{OSWindows}{
\PackageInfo{lwarp}{Using option 'OSWindows'}
\LWR@setOSWindows
}
```

Opt HomeHTMLFilename
Default: \BaseJobname

The filename of the homepage. The default is the jobname. This option is stored into `\LWR@HomeHTMLFilename`, and later transferred into `\HomeHTMLFilename` for internal use.

```
\DeclareStringOption[]{HomeHTMLFilename}
```

Opt HTMLFilename
Default: <empty>

The filename prefix of web pages after the homepage. The default is empty, no prefix. This option is stored into `\LWR@HTMLFilename`, and later transferred into `\HTMLFilename` for internal use.

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

Opt PrintLatexCmd
Default: <automatic>

The shell commands to use to compile the print document.

```
\DeclareStringOption[]{PrintLatexCmd}
```

Opt HTMLLatexCmd
Default: <automatic>

The shell commands to use to compile the HTML document.

```
\DeclareStringOption[]{HTMLLatexCmd}
```

Opt PrintIndexCmd
Default: <empty>

The shell commands to use to compile the print indexes.

```
\DeclareStringOption[]{PrintIndexCmd}
```

Opt HTMLIndexCmd
Default: <empty>

The shell commands to use to compile the HTML indexes.

```
\DeclareStringOption[]{HTMLIndexCmd}
```

Opt LatexmkIndexCmd
Default: <empty>

The shell commands to by used by *latexmk* to compile the print indexes. Unlike `PrintIndexCmd` and `HTMLIndexCmd`, `LatexmkIndexCmd` does not include the filename, which will be provided by *latexmk*.

```
\DeclareStringOption[]{LatexmkIndexCmd}
```

Opt makeindex

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.

```
\DeclareBoolOption[false]{makeindex}
```

Opt xindy

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.

```
\DeclareBoolOption[false]{xindy}
```

Opt GlossaryCmd
Default: makeglossaries

The shell command to use to compile the glossary. The print or HTML version of the glossary filename will be appended to this command.

```
\DeclareStringOption[makeglossaries]{GlossaryCmd}
```

Opt latexmk

Option `latexmk` tells *lwarpmk* to use *latexmk* when compiling documents.

```
\DeclareBoolOption[false]{latexmk}
```

Opt dvips

Option `dvips` tells *lwarpmk* to use *dvips* when compiling DVI *latex* documents.

```
\DeclareBoolOption[false]{dvips}
```

Opt dvipdfm

Option `dvipdfm` tells *lwarpmk* to use *dvipdfm* when compiling DVI *latex* documents.

```
\DeclareBoolOption[false]{dvipdfm}
```

Opt dvipdfmx

Option `dvipdfmx` tells *lwarpmk* to use *dvipdfmx* when compiling DVI *latex* documents.

```
\DeclareBoolOption[false]{dvipdfmx}
```

Execute options

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

```
\ProcessKeyvalOptions*\relax
```

28.1 Additional options support

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

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

Defaults unless already over-ridden by the user:

```
\ifcsempty{LWR@HomeHTMLFilename}{
    \newcommand*{\HomeHTMLFilename}{\BaseJobname}
}{
    \csedef{HomeHTMLFilename}{\LWR@HomeHTMLFilename}
}

\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.

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

\LWR@sanitize {⟨*text*⟩}

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

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

Sanitize some string options to neutralize underscores.

```
\LWR@sanitize{\LWR@BaseJobname}
\edef\LWR@BaseJobname{\LWR@sanitized}

\LWR@sanitize{\LWR@ImagesDirectory}
\edef\LWR@ImagesDirectory{\LWR@sanitized}

\LWR@sanitize{\LWR@ImagesName}
\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.

```
\ifdefempty{\LWR@PrintIndexCmd}{
    \renewcommand{\LWR@PrintIndexCmd}{%
        makeindex  -s \LWR@makeindexStyle \space  \jobname.idx%
    }
    \ifbool{LWR@xindy}{
        \renewcommand{\LWR@PrintIndexCmd}{%
            xindy
            -M \LWR@xindyStyle \space
            -L \LWR@xindyLanguage \space
            -C \LWR@xindyCodepage \space
            \jobname.idx%
        }
    }{}
}{}

\ifdefempty{\LWR@HTMLIndexCmd}{
    \renewcommand{\LWR@HTMLIndexCmd}{%
        makeindex  -s \LWR@makeindexStyle \space  \jobname_html.idx%
    }
    \ifbool{LWR@xindy}{
        \renewcommand{\LWR@HTMLIndexCmd}{%
            xindy
            -M \LWR@xindyStyle \space
            -L \LWR@xindyLanguage \space
```

```
            -C \LWR@xindyCodepage \space
            \jobname_html.idx%
        }
    }{}
}{}

\ifdefempty{\LWR@LatexmkIndexCmd}{
    \renewcommand{\LWR@LatexmkIndexCmd}{%
        makeindex  -s \LWR@makeindexStyle%
    }
    \ifbool{LWR@xindy}{
        \renewcommand{\LWR@LatexmkIndexCmd}{%
            xindy
            -M \LWR@xindyStyle \space
            -L \LWR@xindyLanguage \space
            -C \LWR@xindyCodepage%
        }
    }{}
}{}
```

28.2 Conditional compilation

\warpprintonly {⟨*contents*⟩}

Only process the contents if producing printed output.

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

\warpHTMLonly {⟨*contents*⟩}

Only process the contents if producing HTML output.

```
\newcommand{\warpHTMLonly}[1]{\ifbool{warpingHTML}{#1}{}}
```

Pkg comment 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.

```
\RequirePackage{comment}
```

Use `comment_print.cut` for print mode, and `comment_html.cut` for HTML mode. This helps *latexmk* to more reliably know whether to recompile.

```
\ifbool{warpingHTML}{
\def\DefaultCutFileName{\def\CommentCutFile{comment_html.cut}}
}{}

\ifbool{warpingprint}{
\def\DefaultCutFileName{\def\CommentCutFile{comment_print.cut}}
}{}
```

Env warpall Anything in the warpall environment will be generated for print or HTML outputs.

```
\includecomment{warpall}
```

Env warpprint Anything in the warpprint environment will be generated for print output only.

Env warpHTML For HTML output:

```
\ifbool{warpingHTML}
{\includecomment{warpHTML}}
{\excludecomment{warpHTML}}%

\ifbool{warpingprint}
{\includecomment{warpprint}}
{\excludecomment{warpprint}}
```

Optionally generate a local copy of *lwarpmk*. Default to no.

```
\ifbool{LWR@creatinglwarpmk}
{\includecomment{LWR@createlwarpmk}}
{\excludecomment{LWR@createlwarpmk}}
```

29 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:

```
\begin{warpHTML}
```

Load fontspec if necessary:

```
\ifxetexorluatex
\@ifpackageloaded{fontspec}{}{
    \usepackage[no-math]{fontspec}
}
```

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

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

pdflatex only: Only pre-loaded if *pdflatex* is being used.

Pkg microtype

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.

```
\RequirePackage {microtype}

\microtypesetup{
    protrusion=false,
    expansion=false,
    tracking=false,
    kerning=false,
    spacing=false}

\DisableLigatures[f,q,t,T,Q]{encoding = *,family = *}

\fi

\end{warpHTML}
```

Pkg geometry 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.

for HTML output:

```
\begin{warpHTML}

\RequirePackage{geometry}
```

Avoid class and option conflict by changing settings after package load:

```
\geometry{
    paperheight=190in,%
    paperwidth=20in,%
    left=2in,right=6in,%
    top=1in,bottom=1in,%
}
```

ltjtarticle and other classes may require these to be reset by lwarp:

```
\setlength{\textheight}{0.8\paperheight}
\setlength{\textwidth}{0.7\paperwidth}

\@twosidefalse
\@mparswitchfalse

\end{warpHTML}
```

for HTML & PRINT:

```
\begin{warpall}
```

Pkg xparse

LaTeX3 command argument parsing

```
\RequirePackage{xparse}
```

Pkg calc

```
\RequirePackage{calc}
```

```
\end{warpall}
```

for HTML output:

```
\begin{warpHTML}
```

Pkg expl3

LaTeX3 programming

```
\RequirePackage{expl3}
```

Pkg gettitlestring

Used to emulate \nameref.

```
\RequirePackage{gettitlestring}
```

Pkg everyhook

everyhook is used to patch paragraph handling.

```
\@ifundefined{bxjs@everypar}{}{\let\everypar\bxjs@everypar}

\RequirePackage{everyhook}
\end{warpHTML}
```

for HTML & PRINT:

```
\begin{warpall}
```

Pkg filecontents

Used to write helper files, done in print mode.

Patched 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

```
\RequirePackage{filecontents}

\@ifpackagelater{filecontents}{2011/10/09}%
{}
{
\newwrite\fcwrite
\let\LWR@origfilec@ntents\filec@ntents
\def\filec@ntents{\def\chardef##1\write{\let\reserved@c\fcwrite}\LWR@origfilec@ntents}
}
```

```
\end{warpall}
```

for HTML output:

```
\begin{warpHTML}
```

Pkg xifthen

```
\RequirePackage{xifthen}
```

Pkg verbatim

```
\RequirePackage{verbatim}
```

Pkg refcount

Provides `\setcounterref`, `\setcounterpageref`, etc.

```
\RequirePackage{refcount}
```

Pkg newfloat

```
\RequirePackage{newfloat}
```

```
\end{warpHTML}
```

for HTML & PRINT:

```
\begin{warpall}
```

Pkg xstring

⚠ index

There was a short-term bug in xstring regarding `\IfInteger` which affected lwarp's index generation. The updated version is requested here.

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

Pkg environ

Used to encapsulate math environments for re-use in HTML `<alt>` text.

```
\RequirePackage{environ}
```

```
\end{warpall}
```

for HTML output:

```
\begin{warpHTML}
```

Pkg printlen

Used to convert lengths for image width/height options.

```
\RequirePackage{printlen}
```

`\LWR@printlength` {⟨*length*⟩}

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

```
\newrobustcmd*{\LWR@printlength}[1]{%
\begingroup%
\uselengthunit{PT}%
```

```
\renewcommand*{\unitspace}{}%
\ifdimless{#1}{10pt}{%
    \printlength{#1}%
}{%
    \rndprintlength{#1}%
}%
\endgroup%
}
```

```
\end{warpHTML}
```

for PRINT output:

```
\begin{warpprint}
```

Pkg varwidth

Used for print-mode `lateximage`:

```
\RequirePackage{varwidth}
```

```
\end{warpprint}
```

30 Loading packages

for HTML & PRINT:

```
\begin{warpall}
```

Remember the original `\RequirePackage`:

```
\LetLtxMacro\LWR@origRequirePackage\RequirePackage
```

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

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

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

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

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

```
\newcommand*{\LWR@nullifycomment}{%
    \PackageInfo{lwarp}%
        {Nullifying the comment environment before loading \LWR@strresulttwo,}%
    \let\comment\relax%
    \let\endcomment\relax%
}
```

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

Note that argument 4 is passed directly to \StrBetween.

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

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

The following should never be loaded:

```
\newcommand*{\LWR@checkloadfilename}[1]{%
\LWR@checkloadnever{#1}{ae}{latinmodern}
\LWR@checkloadnever{#1}{aecc}{latinmodern}
\LWR@checkloadnever{#1}{bitfield}{bytefield}
\LWR@checkloadnever{#1}{boxedminipage}{boxedminipage2e}
\LWR@checkloadnever{#1}{caption2}{caption}
\LWR@checkloadnever{#1}{ccaption}{caption}
\LWR@checkloadnever{#1}{colortab}{colortbl}
\LWR@checkloadnever{#1}{doublespace}{setspace}
\LWR@checkloadnever{#1}{epsf}{graphicx}
\LWR@checkloadnever{#1}{fancyheadings}{fancyhdr}
\LWR@checkloadnever{#1}{glossary}{glossaries}
\LWR@checkloadnever{#1}{hyper}{hyperref}
\LWR@checkloadnever{#1}{newthm}{ntheorem}
\LWR@checkloadnever{#1}{picinpar}{floatflt, wrapfig}
\LWR@checkloadnever{#1}{picins}{floatflt, wrapfig}
\LWR@checkloadnever{#1}{rplain}{fancyhdr}
\LWR@checkloadnever{#1}{si}{siunitx}
\LWR@checkloadnever{#1}{sistyle}{siunitx}
\LWR@checkloadnever{#1}{t1enc}{fontenc, inputenc, inputenx}
\LWR@checkloadnever{#1}{ucs}{inputenc, inputencx}
\LWR@checkloadnever{#1}{wasysym}{textcomp, amssymb, amsfonts, mnsymbol, fdsymbol}
```

The following should only be loaded before lwarp:

```
\LWR@checkloadbefore{#1}{ctex}
\LWR@checkloadbefore{#1}{fontspec}
\LWR@checkloadbefore{#1}{inputenc}
\LWR@checkloadbefore{#1}{inputenx}
\LWR@checkloadbefore{#1}{kotex}
\LWR@checkloadbefore{#1}{luatexja}
\LWR@checkloadbefore{#1}{luatexja-fontspec}
\LWR@checkloadbefore{#1}{luatexko}
\LWR@checkloadbefore{#1}{morewrites}
\LWR@checkloadbefore{#1}{newclude}
\LWR@checkloadbefore{#1}{newunicodechar}
\LWR@checkloadbefore{#1}{plext}
\LWR@checkloadbefore{#1}{xeCJK}
\LWR@checkloadbefore{#1}{xetexko}
\LWR@checkloadbefore{#1}{zxjatype}
}
```

\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.

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

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

```
\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`.

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

See if the package name was found:

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

Possible adjustments before loading the package. Maybe nullify the `comment` environment if the new package will be redefining it for a new purpose.

```
    \ifdefstring{\LWR@strresulttwo}{easyReview}{\LWR@nullifycomment}{}%
```

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

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

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

```
    \ifboolexpr{
        bool{warpingHTML} and
        test{\IfFileExists{lwarp-\LWR@strresulttwo.sty}}
    }%
    {% lwarp-* file found
        \ifdefvoid{\LWR@parsedrequirepackagenames}{%
            \edef\LWR@parsedrequirepackagenames{lwarp-\LWR@strresulttwo}%
        }{%
            \edef\LWR@parsedrequirepackagenames{%
                \LWR@parsedrequirepackagenames,lwarp-\LWR@strresulttwo%
            }%
        }%
    }%
    {%
```

Otherwise, use the current package name.

```
        \ifdefvoid{\LWR@parsedrequirepackagenames}{%
            \edef\LWR@parsedrequirepackagenames{\LWR@strresulttwo}%
```

```
        }{%
            \edef\LWR@parsedrequirepackagenames{%
                \LWR@parsedrequirepackagenames,\LWR@strresulttwo%
            }%
        }%
    }% no lwarp-* file
}% yes filename
}
```

\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.

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

Redirect up to twenty names:[16]

```
\renewcommand*{\LWR@requirepackagenames}{#2}%
\renewcommand*{\LWR@parsedrequirepackagenames}{}%
\LWR@lookforpackagename{1}%
\LWR@lookforpackagename{2}%
\LWR@lookforpackagename{3}%
\LWR@lookforpackagename{4}%
\LWR@lookforpackagename{5}%
\LWR@lookforpackagename{6}%
\LWR@lookforpackagename{7}%
\LWR@lookforpackagename{8}%
\LWR@lookforpackagename{9}%
\LWR@lookforpackagename{10}%
\LWR@lookforpackagename{11}%
\LWR@lookforpackagename{12}%
\LWR@lookforpackagename{13}%
\LWR@lookforpackagename{14}%
\LWR@lookforpackagename{15}%
\LWR@lookforpackagename{16}%
\LWR@lookforpackagename{17}%
\LWR@lookforpackagename{18}%
\LWR@lookforpackagename{19}%
\LWR@lookforpackagename{20}%
```

\RequirePackage depending on the options and version:

```
\IfValueTF{#1}%
{% options given
    \IfValueTF{#3}% version given?
    {\LWR@origRequirePackage[#1]{\LWR@parsedrequirepackagenames}[#3]}%
    {\LWR@origRequirePackage[#1]{\LWR@parsedrequirepackagenames}}%
}%
{% no options given
    \IfValueTF{#3}% version given?
    {\LWR@origRequirePackage{\LWR@parsedrequirepackagenames}[#3]}%
```

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

```
    {\LWR@origRequirePackage{\LWR@parsedrequirepackagenames}}%
}%
}
\LetLtxMacro\usepackage\RequirePackage

\end{warpall}
```

for HTML output:

```
\begin{warpHTML}
```

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

Uses the original package, including options.

```
\NewDocumentCommand{\LWR@ProvidesPackagePass}{m o}{
\PackageInfo{lwarp}{Using package `#1' and adding lwarp modifications, including options,}%
\IfValueTF{#2}
{\ProvidesPackage{lwarp-#1}[#2]}
{\ProvidesPackage{lwarp-#1}}
\DeclareOption*{\PassOptionsToPackage{\CurrentOption}{#1}}
\ProcessOptions\relax
\IfValueTF{#2}
{\LWR@origRequirePackage{#1}[#2]}
{\LWR@origRequirePackage{#1}}
```

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

```
\DeclareOption*{}%
\ProcessOptions\relax
}
```

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

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

```
\NewDocumentCommand{\LWR@ProvidesPackageDrop}{m o}{
\PackageInfo{lwarp}{Replacing package `#1' with the lwarp version, discarding options,}%
\IfValueTF{#2}
{\ProvidesPackage{lwarp-#1}[#2]}
{\ProvidesPackage{lwarp-#1}}
```

Ignore all options.

```
\DeclareOption*{}
```

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

```
% \ProcessOptions\relax% original LaTeX code
\let\ds@\@empty%         from the original \ProcessOptions
\edef\@curroptions{}%    lwarp modification to \ProcessOptions
\@process@ptions\relax%  from the original \ProcessOptions
```

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

31 Additional required packages

for HTML output:

```
\begin{warpHTML}
```

Pkg caption

```
\LWR@origRequirePackage{caption}
\AtBeginDocument{\RequirePackage{lwarp-caption}}%
\end{warpHTML}
```

32 File handles

Defines file handles for writes.

for HTML & PRINT:

```
\begin{warpall}
```

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

```
\newwrite\LWR@quickfile%
\end{warpall}
```

for HTML output:

```
\begin{warpHTML}
```

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

```
\newwrite\LWR@lateximagesfile
\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.

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

\include {⟨*filename*⟩}

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

```
\def\@include#1 {%
\immediate\openout\LWR@quickfile #1_html_inc.tex% lwarp
\immediate\write\LWR@quickfile{\string\input{#1.tex}}% lwarp
\immediate\closeout\LWR@quickfile% lwarp
\LWR@origclearpage% \changed
\if@filesw
    \immediate\write\@mainaux{\string\@input{#1_html_inc.aux}}% changed
\fi
\@tempswatrue
\if@partsw
    \@tempswafalse
    \edef\reserved@b{#1}%
    \@for\reserved@a:=\@partlist\do
    {\ifx\reserved@a\reserved@b\@tempswatrue\fi}%
\fi
\if@tempswa
    \let\@auxout\@partaux
    \if@filesw
        \immediate\openout\@partaux #1_html_inc.aux % changed
        \immediate\write\@partaux{\relax}%
    \fi
    \@input@{#1_html_inc.tex}% changed
    \LWR@origclearpage% changed
    \@writeckpt{#1}%
    \if@filesw
        \immediate\closeout\@partaux
    \fi
\else
    \deadcycles\z@
    \@nameuse{cp@#1}%
\fi
\let\@auxout\@mainaux%
}
```

```
\end{warpHTML}
```

34 Copying a file

for HTML output:
```
\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.

```
\newwrite\LWR@copyoutfile  % open the file to write to
\newread\LWR@copyinfile      % open the file to read from

\newcommand*{\LWR@copyfile}[2]{%
\LWR@traceinfo{LWR@copyfile: copying #1 to #2}

\immediate\openout\LWR@copyoutfile=#2
\openin\LWR@copyinfile=#1
\begingroup\endlinechar=-1
\makeatletter

\LWR@traceinfo{LWR@copyfile: about to loop}

\loop\unless\ifeof\LWR@copyinfile
  \LWR@traceinfo{LWR@copyfile: one line}
 \read\LWR@copyinfile to\LWR@fileline % Read one line and store it into \LWR@fileline
%   \LWR@fileline\par                  % print the content into the pdf
% print the content:
  \immediate\write\LWR@copyoutfile{\unexpanded\expandafter{\LWR@fileline}}%
\repeat
\immediate\closeout\LWR@copyoutfile
\LWR@traceinfo{LWR@copyfile: done}
\endgroup
}

\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:

```
\begin{warpall}
```

Bool `LWR@tracinglwarp`

True if tracing is turned on.

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

`\tracinglwarp`

Turns on the debug tracing messages.

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

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

```
\newcommand{\LWR@traceinfo}[1]{%
\ifbool{LWR@tracinglwarp}%
{%
    \typeout{*** lwarp: #1}%
}%
{}%
}
```

Bool HTMLDebugComments Add comments in HTML about closing s, sections, etc.
Default: false

```
\newbool{HTMLDebugComments}
\boolfalse{HTMLDebugComments}
```

If \tracinglwarp, show where preamble hooks occur:

```
\AfterEndPreamble{
\LWR@traceinfo{AfterEndPreamble}
}

\AtBeginDocument{
\LWR@traceinfo{AtBeginDocument}
}
```

```
\end{warpall}
```

36 Defining print and HTML versions of macros and environments

The following refers to defining objects inside lwarp, and 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 latexmage environment, which draws and remembers an image of the printed output, and also several other places. The copies of the print versions may also be used by the HTML versions, such as when the HTML version merely encloses the print version inside HTML tags.

The general structure for providing print and HTML versions of a macro or environment is as follows:

For a preexisting macro, not defined with xparse: 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}{...}% may also use xparse

\LWR@formatted{name}
\end{warpHTML}
```

\LWR@formatted{name} copies the original print version, then redefines \name to use either the print or HTML version depending on which mode lwarp is using. xparse may be used to define the new HTML version, even if the original did not use xparse.

For a preexisiting environment, not defined with xparse: The process is similar. Note the use of \LWR@formattedenv instead of \LWR@formatted.

```
\begin{warpHTML}
\newenvironment{LWR@HTML@name}{...}% may also use xparse

\LWR@formattedenv{name}
\end{warpHTML}
```

If the original used xparse: A copy must be made using a new name:

```
\begin{warpHTML}
\NewDocumentCommand{\LWR@print@name}{..}{..}% copy the original

\NewDocumentCommand{\LWR@HTML@name}{..}{..}% or use \newcommand

\LWR@formatted{name}
\end{warpHTML}
```

Similar for an environment, using \LWR@formattedenv. (\LWR@formatted and \LWR@formattedenv use \LetLtxMacro to copy the original print definiton, which may not work with macros and environments created by xparse, so the print version must be manually recreated in the lwarp source.)

For a new macro or environment, not using xparse for the print version:

```
\begin{warpall}
\newcommand{\name}{...}% NOT xparse!
\end{warpall}

\begin{warpHTML}
\newcommand{\LWR@HTML@name}{...}% may use xparse for HTML

\LWR@formatted{name}
\end{warpHTML}
```

Similar for an environment. The plain \name or environment name is used for the printed version, and is placed inside warpall. xparse may be used for the \LWR@HTML@<name> version.

For a new macro or environment, using xparse: It is possible to use xparse for an entirely new macro or environment by defining the \LWR@print@<name> version with xparse, along with \name defined without xparse to refer directly to the \LWR@print version:

```
\begin{warpall}
\NewDocumentCommand{\LWR@print@name}{...} {...}%      -or-
\NewDocumentEnvironment{\LWR@print@name}{...} {...} {...}

% Simply a call to \LWR@print@name:
\newcommand{\name}{\LWR@print@name}%        -or-
\newenvironment{name}{\LWR@print@name}{\endLWR@print@name}
\end{warpall}

\begin{warpHTML}
\NewDocumentCommand{\LWR@HTML@name}{...} {...}%      -or-
\NewDocumentEnvironment{\LWR@HTML@name}{...} {...} {...}

\LWR@formatted{name}%    -or-
\LWR@formattedenv{name}
\end{warpHTML}
```

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>.
- 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 arguments are not handled by the new \name, any star and other arguments are processed by the print or HTML version.

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.)

for HTML output:

```
\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`".

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

`\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`.

```
\newcommand*{\LWR@formatted}[1]{%
    \ifcsundef{LWR@print@#1}{%
        \expandafter\LetLtxMacro\csname LWR@print@#1\expandafter\endcsname%
            \csname#1\endcsname%
    }{}%
    \ifcsundef{#1}{%
        \expandafter\newrobustcmd\csname #1\endcsname{%
            \@nameuse{LWR@\LWR@formatting @#1}%
        }%
    }{%
        \expandafter\renewrobustcmd\csname #1\endcsname{%
            \@nameuse{LWR@\LWR@formatting @#1}%
        }%
    }%
}
```

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

An expandable version of `\LWR@formatted`.

```
\newcommand*{\LWR@expandableformatted}[1]{%
    \ifcsundef{LWR@print@#1}{%
        \expandafter\LetLtxMacro\csname LWR@print@#1\expandafter\endcsname%
            \csname#1\endcsname%
    }{}%
    \ifcsundef{#1}{%
        \expandafter\newcommand\csname #1\endcsname{%
            \@nameuse{LWR@\LWR@formatting @#1}%
        }%
```

```
    }{%
        \expandafter\renewcommand\csname #1\endcsname{%
            \@nameuse{LWR@\LWR@formatting @#1}%
        }%
    }%
}
```

`\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`.

```
\newcommand*{\LWR@formattedenv}[1]{%
    \ifcsundef{LWR@print@#1}{%
        \expandafter\LetLtxMacro\csname LWR@print@#1\expandafter\endcsname%
            \csname#1\endcsname%
        \csletcs{endLWR@print@#1}{end#1}%
    }{}%
    \DeclareDocumentEnvironment{#1}{}%
    {\@nameuse{LWR@\LWR@formatting @#1}}%
    {\@nameuse{endLWR@\LWR@formatting @#1}}%
}
```

`\LWR@expandableformattedenv` {⟨*environmentname*⟩}

An expandable version of `LWR@formattedenv`.

```
\newcommand*{\LWR@expandableformattedenv}[1]{%
    \ifcsundef{LWR@print@#1}{%
        \expandafter\LetLtxMacro\csname LWR@print@#1\expandafter\endcsname%
            \csname#1\endcsname%
        \csletcs{endLWR@print@#1}{end#1}%
    }{}%
    \DeclareExpandableDocumentEnvironment{#1}{}%
    {\@nameuse{LWR@\LWR@formatting @#1}}%
    {\@nameuse{endLWR@\LWR@formatting @#1}}%
}
```

```
\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:

```
\begin{warpall}
```

37.1 User-level controls

Bool FormatEPUB
Default: false

Changes HTML output for easy EPUB conversion via an external program. Removes per-file headers, footers, and nav. Adds footnotes per chapter/section.

```
\newbool{FormatEPUB}
\boolfalse{FormatEPUB}
```

Bool FormatWP
Default: false

Changes HTML output for easier conversion by a word processor. Removes headers and nav, prints footnotes per section, and also forces single-file output and turns off HTML debug comments.

```
\newbool{FormatWP}
\boolfalse{FormatWP}
```

Bool WPMarkFloats
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. [17]

```
\newbool{WPMarkFloats}
\boolfalse{WPMarkFloats}
```

Bool WPMarkMinipages
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.

```
\newbool{WPMarkMinipages}
\boolfalse{WPMarkMinipages}
```

Bool WPMarkTOC
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.

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

If set `false`*, the actual* TOC *is printed instead.*

```
\newbool{WPMarkTOC}
\booltrue{WPMarkTOC}
```

Bool `WPMarkLOFT` 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.*

```
\newbool{WPMarkLOFT}
\boolfalse{WPMarkLOFT}
```

Bool `WPMarkMath` Default: `false`

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.

```
\newbool{WPMarkMath}
\boolfalse{WPMarkMath}
```

Bool `WPTitleHeading` Default: `false`

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**.

See table 8 on page 175.

```
\newbool{WPTitleHeading}
\boolfalse{WPTitleHeading}
```

```
\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:

```
\begin{warpHTML}
```

```
\AtBeginDocument{
\ifbool{FormatWP}{
\@ifundefined{chapter}{
\ifbool{WPTitleHeading}{% part and section starting at h2
```

```
\renewcommand*{\LWR@tagtitle}{h1}
\renewcommand*{\LWR@tagtitleend}{/h1}
\renewcommand*{\LWR@tagpart}{h2}
\renewcommand*{\LWR@tagpartend}{/h2}
\renewcommand*{\LWR@tagsection}{h3}
\renewcommand*{\LWR@tagsectionend}{/h3}
\renewcommand*{\LWR@tagsubsection}{h4}
\renewcommand*{\LWR@tagsubsectionend}{/h4}
\renewcommand*{\LWR@tagsubsubsection}{h5}
\renewcommand*{\LWR@tagsubsubsectionend}{/h5}
\renewcommand*{\LWR@tagparagraph}{h6}
\renewcommand*{\LWR@tagparagraphend}{/h6}
\renewcommand*{\LWR@tagsubparagraph}{span class="subparagraph"}
\renewcommand*{\LWR@tagsubparagraphend}{/span}
}% WPTitleHeading
{% not WPTitleHeading, part and section starting at h1
\renewcommand*{\LWR@tagtitle}{div class="title"}
\renewcommand*{\LWR@tagtitleend}{/div}
\renewcommand*{\LWR@tagpart}{h1}
\renewcommand*{\LWR@tagpartend}{/h1}
\renewcommand*{\LWR@tagsection}{h2}
\renewcommand*{\LWR@tagsectionend}{/h2}
\renewcommand*{\LWR@tagsubsection}{h3}
\renewcommand*{\LWR@tagsubsectionend}{/h3}
\renewcommand*{\LWR@tagsubsubsection}{h4}
\renewcommand*{\LWR@tagsubsubsectionend}{/h4}
\renewcommand*{\LWR@tagparagraph}{h5}
\renewcommand*{\LWR@tagparagraphend}{/h5}
\renewcommand*{\LWR@tagsubparagraph}{h6}
\renewcommand*{\LWR@tagsubparagraphend}{/h6}
}% not WPTitleHeading
}% chapter undefined
{% chapter defined
\ifbool{WPTitleHeading}{}
{% not WPTitleHeading, part and chapter starting at h1
\renewcommand*{\LWR@tagtitle}{div class="title"}
\renewcommand*{\LWR@tagtitleend}{/div}
\renewcommand*{\LWR@tagpart}{h1}
\renewcommand*{\LWR@tagpartend}{/h1}
\renewcommand*{\LWR@tagchapter}{h2}
\renewcommand*{\LWR@tagchapterend}{/h2}
\renewcommand*{\LWR@tagsection}{h3}
\renewcommand*{\LWR@tagsectionend}{/h3}
\renewcommand*{\LWR@tagsubsection}{h4}
\renewcommand*{\LWR@tagsubsectionend}{/h4}
\renewcommand*{\LWR@tagsubsubsection}{h5}
\renewcommand*{\LWR@tagsubsubsectionend}{/h5}
\renewcommand*{\LWR@tagparagraph}{h6}
\renewcommand*{\LWR@tagparagraphend}{/h6}
\renewcommand*{\LWR@tagsubparagraph}{span class="subparagraph"}
\renewcommand*{\LWR@tagsubparagraphend}{/span}
}% not WPTitleHeading
}% chapter defined
}{}% FormatWP
}% AtBeginDocument
```

```
\end{warpHTML}
```

38 Remembering original formatting macros

for HTML output:

```
\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. Also nullify unused commands.

Some packages redefine \#, which is used to generate HTML, so the original must be remembered here.

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

\let\LWR@origcomma\,
\let\LWR@origthinspace\thinspace
\let\LWR@orignegthinspace\negthinspace
\let\LWR@origtilde~
\let\LWR@origenskip\enskip
\let\LWR@origquad\quad
\let\LWR@origqquad\qquad
\let\LWR@orighfil\hfil
\let\LWR@orighss\hss
\let\LWR@origllap\llap
\let\LWR@origrlap\rlap
\let\LWR@orighfilneg\hfilneg
\let\LWR@orighspace\hspace

\let\LWR@origrule\rule

\let\LWR@origmedskip\medskip
\let\LWR@origbigskip\bigskip

\let\LWR@origtextellipsis\textellipsis
\let\LWR@orig@textquotedbl\textquotedbl

\LetLtxMacro\LWR@origrmfamily\rmfamily
\LetLtxMacro\LWR@origsffamily\sffamily
\LetLtxMacro\LWR@origttfamily\ttfamily
\LetLtxMacro\LWR@origbfseries\bfseries
\LetLtxMacro\LWR@origmdseries\mdseries
\LetLtxMacro\LWR@origupshape\upshape
\LetLtxMacro\LWR@origslshape\slshape
\LetLtxMacro\LWR@origscshape\scshape

\@ifundefined{sishape}{
    \LetLtxMacro\LWR@origsishape\scshape
}{
    \LetLtxMacro\LWR@origsishape\sishape
}
```

```
\LetLtxMacro\LWR@origitshape\itshape
\LetLtxMacro\LWR@origem\em
\LetLtxMacro\LWR@orignormalfont\normalfont

\let\LWR@origonecolumn\onecolumn

\let\LWR@origsp\sp
\let\LWR@origsb\sb
\LetLtxMacro\LWR@origtextsuperscript\textsuperscript
\LetLtxMacro\LWR@orig@textsuperscript\@textsuperscript

\AtBeginDocument{
\LetLtxMacro\LWR@origtextsubscript\textsubscript
\LetLtxMacro\LWR@orig@textsubscript\@textsubscript
}

\LetLtxMacro\LWR@origunderline\underline


\let\LWR@orignewpage\newpage

\let\LWR@origpagestyle\pagestyle
\let\LWR@origthispagestyle\thispagestyle
\LetLtxMacro\LWR@origpagenumbering\pagenumbering

\let\LWR@orignewline\newline


\AtBeginDocument{% in case packages change definition
\let\LWR@orig@trivlist\@trivlist
\let\LWR@origtrivlist\trivlist
\let\LWR@origendtrivlist\endtrivlist
\LetLtxMacro\LWR@origitem\item
\LetLtxMacro\LWR@origitemize\itemize
\LetLtxMacro\LWR@endorigitemize\enditemize
\LetLtxMacro\LWR@origenumerate\enumerate
\LetLtxMacro\LWR@endorigenumerate\endenumerate
\LetLtxMacro\LWR@origdescription\description
\LetLtxMacro\LWR@endorigdescription\enddescription
\let\LWR@orig@mklab\@mklab
\let\LWR@origmakelabel\makelabel
\let\LWR@orig@donoparitem\@donoparitem
\LetLtxMacro\LWR@orig@item\@item
\let\LWR@orig@nbitem\@nbitem
}

\let\LWR@origpar\par

\LetLtxMacro\LWR@origfootnote\footnote
\let\LWR@orig@mpfootnotetext\@mpfootnotetext

\let\LWR@origclearpage\clearpage


\AtBeginDocument{% in case packages change definition
\LetLtxMacro\LWR@orighline\hline%
```

```
\LetLtxMacro\LWR@origcline\cline%
}

\end{warpHTML}
```

39 Accents

Native LaTeX accents such as `\"` will work, but many more kinds of accents are available when using Unicode-aware XeLaTeX and LuaLaTeX.

for HTML output:

```
\begin{warpHTML}
```

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

```
\AtBeginDocument{
```

The following are restored for print when inside a `lateximage`.

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

```
\LetLtxMacro\LWR@origt\t
```

For pdfLaTeX, additional work is required:

```
\ifPDFTeX% pdflatex or dvi latex
\LetLtxMacro\LWR@origequalaccent\=
\LetLtxMacro\LWR@origdotaccent\.
\LetLtxMacro\LWR@origu\u
\LetLtxMacro\LWR@origv\v
\LetLtxMacro\LWR@origc\c
\LetLtxMacro\LWR@origd\d
\LetLtxMacro\LWR@origb\b
```

The HTML redefinitions follow.

For pdfLaTeX, Unicode diacritical marks are used:

```
\renewcommand*{\=}[1]{#1\HTMLunicode{0305}}
\renewcommand*{\.}[1]{#1\HTMLunicode{0307}}
\renewcommand*{\u}[1]{#1\HTMLunicode{0306}}
\renewcommand*{\v}[1]{#1\HTMLunicode{030C}}
\renewcommand*{\c}[1]{#1\HTMLunicode{0327}}
\renewcommand*{\d}[1]{#1\HTMLunicode{0323}}
\renewcommand*{\b}[1]{#1\HTMLunicode{0331}}
\fi
```

For all engines, a Unicode diacritical tie is used:

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

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

```
\ifPDFTeX% pdflatex or dvi latex
\newcommand*{\LWR@restoreorigaccents}{%
\LetLtxMacro\=\LWR@origequalaccent%
\LetLtxMacro\.\LWR@origdotaccent%
\LetLtxMacro\u\LWR@origu%
\LetLtxMacro\v\LWR@origu%
\LetLtxMacro\t\LWR@origt%
\LetLtxMacro\c\LWR@origc%
\LetLtxMacro\d\LWR@origd%
\LetLtxMacro\b\LWR@origb%
}%
\else% XeLaTeX, LuaLaTeX:
\newcommand*{\LWR@restoreorigaccents}{%
\LetLtxMacro\t\LWR@origt%
}%
\fi%
}% AtBeginDocument

\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:

```
\excludecomment{LWRwriteconf}
```

Generate configuration files if print mode and not -pstool:

for PRINT output:

```
\begin{warpprint}
\fullexpandarg%
\IfSubStr*{\jobname}{-pstool}
    {
        \typeout{lwarp: jobname with -pstool is found,}%
        \typeout{lwarp: \space\space not generating configuration files.}%
    }
    {
        \typeout{lwarp: generating configuration files}
        \includecomment{LWRwriteconf}
    }
\end{warpprint}
```

40.2 <project>_html.tex

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

Config file:
```
\begin{LWRwriteconf}
\immediate\openout\LWR@quickfile=\jobname_html.tex
\immediate\write\LWR@quickfile{%
\detokenize{\PassOptionsToPackage}%
{warpHTML,BaseJobname=\jobname}{lwarp}%
}
\immediate\write\LWR@quickfile{%
\detokenize{\input}\string{\jobname.tex\string }%
}
\immediate\closeout\LWR@quickfile
\end{LWRwriteconf}
```

40.3 *lwarpmk* configuration files

Config file:
```
\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.)

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

40.3.1 Helper macros

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

```
\ifshellescape
    \def\LWR@shellescapecmd{--shell-escape }
\else
    \def\LWR@shellescapecmd{}
\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

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

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

Adds to the compilation command.

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

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

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

```
\newcommand*{\LWR@unknownengine}{%
    \PackageError{lwarp}
    {Unknown LaTeX engine.}
   {Lwarp only knows about pdflatex, dvi latex, xelatex, lualatex, and upLateX.}
}
```

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

Adds a *latexmk* variable assignment.

```
\newcommand*{\LWR@latexmkvar}[2]{%
    -e
    \LWRopquote%
    \LWRdollar #1=q/#2/%
    \LWRopquote
}
```

`\LWR@latexmkcmd` {⟨*latexmk options*⟩}

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

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

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

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

```
\newcommand*{\LWR@latexmkdvipdfm}[1]{%
    -pdfdvi \space
    \LWR@latexmkvar{dvipdf}{%
        #1
        \@percentchar O
        -o \@percentchar D
        \@percentchar S%
    }
}
```

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

```
\newcommand*{\LWR@compileuplatex}{
    \def\LWR@tempprintlatexcmd{%
        \LWR@compilecmd{uplatex}{}
        \LWR@addcompilecmd{dvipdfmx}{}
    }
    \def\LWR@tempHTMLlatexcmd{%
        \LWR@compilecmd{uplatex}{_html}
        \LWR@addcompilecmd{dvipdfmx}{_html}
    }
}
```

\LWR@PrintLatexCmd
\LWR@HTMLLatexCmd
If not set by the user, the following sets the command to use to compile the source 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.

```
\ifbool{LWR@latexmk}{
```

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

```
    \ifpdf
```

For *latexmk* with *pdflatex*:

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

For *latexmk* with *lualatex*:

```
            \ifLuaTeX
```

```
                \def\LWR@latexcmd{\LWR@latexmkcmd{-lualatex}}
            \else
                \LWR@unknownengine
            \fi
        \fi
    \else% \ifpdf
```

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

```
        \ifXeTeX
```

For *latexmk* with *xelatex*:

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

For *latexmk* with DVI *latex*:

```
            \ifbool{LWR@dvipdfm}{
                \def\LWR@latexcmd{%
                    \LWR@latexmkcmd{%
                        \LWR@latexmkdvipdfm{dvipdfm}%
                    }
                }
            }{
                \ifbool{LWR@dvipdfmx}{
                    \def\LWR@latexcmd{%
                        \LWR@latexmkcmd{%
                            \LWR@latexmkdvipdfm{dvipdfmx}%
                        }
                    }
                }{
                    \def\LWR@latexcmd{\LWR@latexmkcmd{-pdfps}}
                }
            }
        \fi
    \fi% \ifpdf
```

The final assignment if *latexmk*:

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

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

```
{% not latexmk
    \ifpdf
```

For *pdflatex* or *lualatex*:

```
        \ifPDFTeX
```

For *pdflatex*:

```
            \def\LWR@tempprintlatexcmd{\LWR@compilecmd{pdflatex}{}}
            \def\LWR@tempHTMLlatexcmd{\LWR@compilecmd{pdflatex}{_html}}
        \else
            \ifLuaTeX
```

For *lualatex*:

```
                \def\LWR@tempprintlatexcmd{\LWR@compilecmd{lualatex}{}}
                \def\LWR@tempHTMLlatexcmd{\LWR@compilecmd{lualatex}{_html}}
            \else
                \LWR@unknownengine
            \fi
        \fi
    \else% \ifpdf
```

For DVI *latex* or *xelatex*:

```
        \ifXeTeX
```

For *xelatex*:

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

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

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

For DVI *latex* with *dvipdfm*:

```
                \def\LWR@tempprintlatexcmd{%
                    \LWR@compilecmd{latex}{}
                    \LWR@addcompilecmd{dvipdfm}{}
                }
                \def\LWR@tempHTMLlatexcmd{%
                    \LWR@compilecmd{latex}{_html}
                    \LWR@addcompilecmd{dvipdfm}{_html}
                }
            }{
                \ifbool{LWR@dvipdfmx}{
```

For DVI *latex* with *dvipdfmx*:

```
                    \def\LWR@tempprintlatexcmd{%
                        \LWR@compilecmd{latex}{}
                        \LWR@addcompilecmd{dvipdfmx}{}
                    }
                    \def\LWR@tempHTMLlatexcmd{%
                        \LWR@compilecmd{latex}{_html}
                        \LWR@addcompilecmd{dvipdfmx}{_html}
                    }
```

```
                }{% dvips
```

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

```
                    \def\LWR@tempprintlatexcmd{%
                        \LWR@compilecmd{latex}{}
                        \LWR@addcompilecmd{dvips}{}
                        \LWR@addcompilecmd{ps2pdf}{}.ps
                    }
                    \def\LWR@tempHTMLlatexcmd{%
                        \LWR@compilecmd{latex}{_html}
                        \LWR@addcompilecmd{dvips}{_html}
                        \LWR@addcompilecmd{ps2pdf}{_html}.ps
                    }
                }
            }
        \fi% \ifXeTeX
    \fi% \ifpdf
}% latexmk
```

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

```
\@ifclassloaded{ujarticle}{\LWR@compileuplatex}{}
\@ifclassloaded{ujbook}{\LWR@compileuplatex}{}
\@ifclassloaded{ujreport}{\LWR@compileuplatex}{}
\@ifclassloaded{utarticle}{\LWR@compileuplatex}{}
\@ifclassloaded{utbook}{\LWR@compileuplatex}{}
\@ifclassloaded{utreport}{\LWR@compileuplatex}{}
```

Only make the setting permanent if the original was empty:

```
\ifdefempty{\LWR@PrintLatexCmd}{
    \def\LWR@PrintLatexCmd{\LWR@tempprintlatexcmd}
}{}
\ifdefempty{\LWR@HTMLLatexCmd}{
    \def\LWR@HTMLLatexCmd{\LWR@tempHTMLlatexcmd}
}{}
```

`\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.

```
\newcommand{\LWR@writeconf}[1]{
\ifcsdef{LWR@quickfile}{}{\newwrite{\LWR@quickfile}}
\immediate\openout\LWR@quickfile=#1
\immediate\write\LWR@quickfile{confversion = [[\LWR@lwarpconfversion]]}
\ifbool{usingOSWindows}{
    \immediate\write\LWR@quickfile{opsystem = [[Windows]]}
}{
    \immediate\write\LWR@quickfile{opsystem = [[Unix]]}
}
\immediate\write\LWR@quickfile{sourcename = [[\jobname]]}
```

```
\immediate\write\LWR@quickfile{homehtmlfilename = [[\HomeHTMLFilename]]}
\immediate\write\LWR@quickfile{htmlfilename = [[\HTMLFilename]]}
\immediate\write\LWR@quickfile{imagesdirectory = [[\LWR@ImagesDirectory]]}
\immediate\write\LWR@quickfile{imagesname = [[\LWR@ImagesName]]}
\immediate\write\LWR@quickfile{latexmk = [[\ifbool{LWR@latexmk}{true}{false}]]}
\immediate\write\LWR@quickfile{printlatexcmd = [[\LWR@PrintLatexCmd]]}
\immediate\write\LWR@quickfile{HTMLlatexcmd = [[\LWR@HTMLLatexCmd]]}
\immediate\write\LWR@quickfile{printindexcmd = [[\LWR@PrintIndexCmd]]}
\immediate\write\LWR@quickfile{HTMLindexcmd = [[\LWR@HTMLIndexCmd]]}
\immediate\write\LWR@quickfile{latexmkindexcmd = [[\LWR@LatexmkIndexCmd]]}
\immediate\write\LWR@quickfile{glossarycmd = [[\LWR@GlossaryCmd]]}
\immediate\write\LWR@quickfile{pdftotextenc = [[\LWR@pdftotextEnc]]}
\immediate\closeout\LWR@quickfile
}

\end{LWRwriteconf}
```

40.3.2 lwarpmk.conf

File lwarpmk.conf

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:

```
\begin{LWRwriteconf}

\AtBeginDocument{\LWR@writeconf{lwarpmk.conf}}

\end{LWRwriteconf}
```

40.3.3 <project>.lwarpmkconf

File project.lwarpmkconf

A project-specific configuration file for *lwarpmk*.

The makeindex and xindy options have already been handled for lwarp.conf.

Config file:

```
\begin{LWRwriteconf}

\AtBeginDocument{\LWR@writeconf{\jobname.lwarpmkconf}}

\end{LWRwriteconf}
```

40.4 lwarp.css

File lwarp.css

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:

```
\begin{LWRwriteconf}
\begin{filecontents*}{lwarp.css}
/*
  CSS stylesheet for the LaTeX lwarp package
  Copyright 2016-2018 Brian Dunn --- BD Tech Concepts LLC
*/


/* a fix for older browsers: */
header, section, footer, aside, nav, main,
    article, figure { display: block; }


A:link {color:#000080 ; text-decoration: none ; }
A:visited {color:#800000 ; }
A:hover {color:#000080 ; text-decoration: underline ;}
A:active {color:#800000 ; }

a.tocpart {display: inline-block ; margin-left: 0em ;
    font-weight: bold ;}
a.tocchapter {display: inline-block ; margin-left: 0em ;
    font-weight: bold ;}
a.tocsection {display: inline-block ; margin-left: 1em ;
    text-indent: -.5em ; font-weight: bold ; }
a.tocsubsection {display: inline-block ; margin-left: 2em ;
    text-indent: -.5em ; }
a.tocsubsubsection {display: inline-block ; margin-left: 3em ;
    text-indent: -.5em ; }
a.tocparagraph {display: inline-block ; margin-left: 4em ;
    text-indent: -.5em ; }
a.tocsubparagraph {display: inline-block ; margin-left: 5em ;
    text-indent: -.5em ; }
a.tocfigure {margin-left: 0em}
a.tocsubfigure {margin-left: 2em}
a.toctable {margin-left: 0em}
a.tocsubtable {margin-left: 2em}
a.toctheorem {margin-left: 0em}
a.toclstlisting {margin-left: 0em}

body {
    font-family: "DejaVu Serif", "Bitstream Vera Serif",
        "Lucida Bright", Georgia, serif;
    background: #FAF7F4 ;
    color: black ;
    margin:0em ;
    padding:0em ;
    font-size: 100% ;
    line-height: 1.2 ;
}

p {margin: 1.5ex 0em 1.5ex 0em ;}
table p {margin: .5ex 0em .5ex 0em ;}

/* Holds a section number */
span.sectionnumber { margin-right: 0em }

```

```
/* Inserted in front of index lines */
span.indexitem {margin-left: 0em}
span.indexsubitem {margin-left: 2em}
span.indexsubsubitem {margin-left: 4em}

div.hidden, span.hidden { display: none ; }

kbd, span.texttt {
    font-family: "DejaVu Mono", "Bitstream Vera Mono", "Lucida Console",
        "Nimbus Mono L", "Liberation Mono", "FreeMono", "Andale Mono",
        "Courier New", monospace;
    font-size: 100% ;
}

pre { padding: 3pt ; }

span.strong, span.textbf, div.strong, div.textbf { font-weight: bold; }

span.textit, div.textit { font-style: italic; }

span.textmd, div.textmd { font-weight: normal; }

span.textsc, div.textsc {
    font-variant: small-caps;
    font-variant-numeric: oldstyle-nums ;
}

span.textsi, div.textsi {
    font-style: italic ;
    font-variant: small-caps;
    font-variant-numeric: oldstyle-nums ;
}

span.textsl, div.textsl { font-style: oblique; }

span.textup, div.textup {
    font-style: normal;
    font-variant: normal;
    font-variant-numeric: normal ;
}

span.textrm, div.textrm {
    font-family: "DejaVu Serif", "Bitstream Vera Serif",
    "Lucida Bright", Georgia, serif;
}

span.textsf, div.textsf {
     font-family: "DejaVu Sans", "Bitstream Vera Sans",
        Geneva, Verdana, sans-serif ;
}

span.textcircled { border: 1px solid black ; border-radius: 1ex ; }

span.underline {
    text-decoration: underline ;
```

```
    text-decoration-skip: auto ;
}

span.overline {
    text-decoration: overline ;
    text-decoration-skip: auto ;
}


/* for vertical text: */
div.verticalrl { writing-mode: vertical-rl }
div.horizontaltb { writing-mode: horizontal-tb }


/* for diagbox */
div.diagboxtitleN { border-bottom: 1px solid gray }
div.diagboxtitleS { border-top: 1px solid gray }

div.diagboxE {
    padding-left: 2em ;
    text-align: right ;
}

div.diagboxW {
    padding-right: 2em ;
    text-align: left ;
}



/* For realscripts */
.supsubscript {
    display: inline-block;
    text-align:left ;
}

.supsubscript sup,
.supsubscript sub {
    position: relative;
    display: block;
    font-size: .5em;
    line-height: 1;
}

.supsubscript sup {
    top: .5em;
}

.supsubscript sub {
    top: .5em;
}

div.attribution p {
    text-align: right ;
    font-size: 80%
```

```
}

span.poemtitle {
  font-size: 120% ; font-weight: bold;
}

pre.tabbing {
    font-family: "Linux Libertine Mono O", "Lucida Console",
        "Droid Sans Mono", "DejaVu Mono", "Bitstream Vera Mono",
        "Liberation Mono", "FreeMono", "Andale Mono",
        "Nimbus Mono L", "Courier New", monospace;
}

blockquote {
    display: block ;
    margin-left: 2em ;
    margin-right: 2em ;
}

blockquotation {
    display: block ;
    margin-left: 2em ;
    margin-right: 2em ;
}

/* quotchap is for the quotchap package */
div.quotchap {
    display: block ;
    font-style: oblique ;
    overflow-x: auto ;
    margin-left: 2em ;
    margin-right: 2em ;
}

blockquote p, blockquotation p, div.quotchap p {
    line-height: 1.5;
    text-align: left ;
    font-size: .85em ;
}

/* qauthor is for the quotchap package */
div.qauthor {
  display: block ;
  text-align: right ;
  margin-left: auto ;
  margin-right: 2em ;
  font-size: 80% ;
  font-variant: small-caps;
}

div.qauthor p {
  text-align: right ;
}

div.epigraph, div.dictum {
```

```
  line-height: 1.2;
    text-align: left ;
    padding: 3ex 1em 0ex 1em ;
/*     margin: 3ex auto 3ex auto ; */ /* Epigraph centered */
    margin: 3ex 1em 3ex auto ; /* Epigraph to the right */
/*    margin: 3ex 1em 3ex 1em ; */ /* Epigraph to the left */
    font-size: .85em ;
    max-width: 27em ;
}

div.epigraphsource, div.dictumauthor {
    text-align:right ;
    margin-left:auto ;
/*     max-width: 50% ; */
    border-top: 1px solid #A0A0A0 ;
    padding-bottom: 3ex ;
    line-height: 1.2;
}

div.epigraph p, div.dictum p { padding: .5ex ; margin: 0ex ;}
div.epigraphsource p, div.dictumauthor p { padding: .5ex 0ex 0ex 0ex ; margin: 0ex ;}
div.dictumauthor { font-style:italic }


/* copyrightbox package */
div.copyrightbox { margin: .5ex .5em }
div.copyrightbox p {margin: 0px .5em ; padding: 0px}
div.copyrightboxnote {text-align: left ; font-size: 60%}


/* lettrine package: */
span.lettrine { font-size: 4ex ; float: left ; }
span.lettrinetext { font-variant: small-caps ; }

/* ulem, soul, umoline packages: */
span.uline {
    text-decoration: underline ;
    text-decoration-skip: auto ;
}

span.uuline {
    text-decoration: underline ;
    text-decoration-skip: auto ;
    text-decoration-style: double ;
}

span.uwave {
    text-decoration: underline ;
    text-decoration-skip: auto ;
    text-decoration-style: wavy ;
}

span.sout {
    text-decoration: line-through ;
}
```

```
1871
1872 span.oline {
1873     text-decoration: overline ;
1874     text-decoration-skip: auto ;
1875 }
1876
1877 span.xout {
1878     text-decoration: line-through ;
1879 }
1880
1881 span.dashuline {
1882     text-decoration: underline ;
1883     text-decoration-skip: auto ;
1884     text-decoration-style: dashed ;
1885 }
1886
1887 span.dotuline {
1888     text-decoration: underline ;
1889     text-decoration-skip: auto ;
1890     text-decoration-style: dotted ;
1891 }
1892
1893 span.letterspacing { letter-spacing: .2ex ; }
1894
1895 span.capsspacing {
1896     font-variant: small-caps ;
1897     letter-spacing: .1ex ;
1898 }
1899
1900 span.highlight { background: #F8E800 ; }
1901
1902
1903
1904
1905 html body {
1906   margin: 0 ;
1907   line-height: 1.2;
1908 }
1909
1910
1911 body div {
1912   margin: 0ex;
1913 }
1914
1915
1916 h1, h2, h3, h4, h5, h6, span.paragraph, span.subparagraph
1917 {
1918     font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
1919         "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
1920         "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
1921         "Times New Roman", serif;
1922     font-style: normal ;
1923     font-weight: bold ;
1924     text-align: left ;
1925 }
```

```
1926
1927 h1 {    /* title of the entire website, used on each page */
1928     text-align: center ;
1929     font-size: 2.5em ;
1930     padding: .4ex 0em 0ex 0em ;
1931 }
1932 h2 { font-size: 2.25em }
1933 h3 { font-size: 2em }
1934 h4 { font-size: 1.75em }
1935 h5 { font-size: 1.5em }
1936 h6 { font-size: 1.25em }
1937 span.paragraph {font-size: 1em ; font-variant: normal ;
1938     margin-right: 1em ; }
1939 span.subparagraph {font-size: 1em ; font-variant: normal ;
1940     margin-right: 1em ;  }
1941
1942 div.minisec {
1943     font-family: "DejaVu Sans", "Bitstream Vera Sans",
1944         Geneva, Verdana, sans-serif ;
1945     font-style: normal ;
1946     font-weight: bold ;
1947     text-align: left ;
1948 }
1949
1950 h1 {
1951   margin: 0ex 0em 0ex 0em ;
1952   line-height: 1.3;
1953   text-align: center ;
1954 }
1955
1956 h2 {
1957   margin: 1ex 0em 1ex 0em ;
1958   line-height: 1.3;
1959   text-align: center ;
1960 }
1961
1962 h3 {
1963   margin: 3ex 0em 1ex 0em ;
1964   line-height: 1.3;
1965 }
1966
1967 h4 {
1968   margin: 3ex 0em 1ex 0em ;
1969   line-height: 1.3;
1970 }
1971
1972 h5 {
1973   margin: 3ex 0em 1ex 0em ;
1974   line-height: 1.3;
1975 }
1976
1977 h6 {
1978   margin: 3ex 0em 1ex 0em ;
1979   line-height: 1.3;
1980 }
```

```
1981
1982
1983 div.titlepage {
1984   text-align: center ;
1985 }
1986
1987 .footnotes {
1988     text-align: left ;
1989     font-size: .85em ;
1990     margin: 3ex 2em 0ex 2em ;
1991     border-top: 1px solid silver ;
1992 }
1993
1994 .marginpar, .marginparblock {
1995     max-width: 50%;
1996     float: right ;
1997     clear: both ;
1998     text-align: left ;
1999     margin: 1ex 0.5em 1ex 1em ;
2000     padding: 1ex 0.5em 1ex 0.5em ;
2001     font-size: 85% ;
2002     border-top: 1px solid silver ;
2003     border-bottom: 1px solid silver ;
2004     overflow-x: auto ;
2005 }
2006
2007 .marginpar br { margin-bottom: 2ex ; }
2008
2009 div.marginblock, div.marginparblock {
2010     max-width:50%;
2011     float:right;
2012     text-align:left;
2013     margin: 1ex 0.5em 1ex 1em ;
2014     padding: 1ex 0.5em 1ex 0.5em ;
2015     overflow-x: auto;
2016 }
2017
2018 div.marginblock div.minipage,
2019 div.marginparblock div.minipage {
2020     display: block ;
2021     margin: 0pt auto 0pt auto ;
2022 }
2023
2024 div.marginblock div.minipage p ,
2025 div.marginparblock div.minipage p
2026     { font-size: 85%}
2027
2028 div.marginblock br ,
2029 div.marginparblock br
2030     { margin-bottom: 2ex ; }
2031
2032 div.bodycontainer {
2033     float: left ;
2034     width: 80% ;
2035 }
```

```
2036
2037 div.bodywithoutsidetoc div.bodycontainer {
2038     float: none ;
2039     width: 100% ;
2040 }
2041
2042 section.textbody div.footnotes{
2043     margin: 3ex 2em 0ex 2em ;
2044     border-bottom: 2px solid silver ;
2045 }
2046
2047 .footnoteheader {
2048     border-top: 2px solid silver ;
2049     margin-top: 3ex ;
2050     padding-top: 1ex ;
2051     font-weight: bold ;
2052 }
2053
2054 .mpfootnotes {
2055     text-align: left ;
2056     font-size: .85em ;
2057     margin-left: 1em ;
2058     border-top: 1px solid silver ;
2059 }
2060
2061 /* Remove footnote top border in the title page. */
2062 div.titlepage div.mpfootnotes {
2063     border-top: none ;
2064 }
2065
2066
2067
2068 ul, ol {
2069   margin: 1ex 1em 1ex 0em;
2070   line-height: 1.2;
2071 }
2072
2073 body dir, body menu {
2074   margin: 3ex 1em 3ex 0em;
2075   line-height: 1.2;
2076 }
2077
2078 li { margin: 0ex 0em 1ex 0em; }
2079
2080 html {
2081   margin: 0;
2082   padding: 0;
2083 }
2084
2085 .programlisting {
2086   font-family: "DejaVu Mono", "Bitstream Vera Mono", "Lucida Console",
2087         "Nimbus Mono L", "Liberation Mono", "FreeMono", "Andale Mono",
2088         "Courier New", monospace;
2089   margin: 1ex 0ex 1ex 0ex ;
2090   padding: .5ex 0pt .5ex 0pt ;
```

```
  overflow-x: auto;
}

section.textbody>pre.programlisting {
border-top: 1px solid silver ;
border-bottom: 1px solid silver ;
}


div.displaymath {
    text-align: center ;
}

div.displaymathnumbered {
    text-align: right ;
    margin-left: 5% ;
    margin-right: 5% ;
    min-width: 2.5in ;
}

@media all and (min-width: 400px) {
    div.displaymathnumbered {
        margin-left: 10% ;
        margin-right: 10% ;
    }
}

@media all and (min-width: 800px) {
    div.displaymathnumbered {
        margin-right: 20% ;
    }
}

@media all and (min-width: 1200px) {
    div.displaymathnumbered {
        margin-right: 30% ;
    }
}


.inlineprogramlisting {
  font-family: "DejaVu Mono", "Bitstream Vera Mono", "Lucida Console",
        "Nimbus Mono L", "Liberation Mono", "FreeMono", "Andale Mono",
        "Courier New", monospace;
  overflow-x: auto;
}

span.listinglabel {
    display: inline-block ;
    font-size: 70% ;
    width: 4em ;
    text-align: right ;
    margin-right: 2em ;
}

```

```
2146 div.abstract {
2147   margin: 2em 5% 2em 5% ;
2148   padding: 1ex 1em 1ex 1em ;
2149 /*  font-weight: bold ; */
2150   font-size: 90% ;
2151     text-align: left ;
2152 }
2153
2154 div.abstract dl {line-height:1.5;}
2155 div.abstract dt {color:#304070;}
2156
2157 div.abstracttitle{
2158     font-family: "URW Classico", Optima, "Linux Biolinum O",
2159         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
2160         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
2161     font-weight:bold;
2162     font-size:1.25em;
2163     text-align: center ;
2164 }
2165
2166 span.abstractrunintitle{
2167     font-family: "URW Classico", Optima, "Linux Biolinum O",
2168         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
2169         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
2170     font-weight:bold;
2171 }
2172
2173
2174 .verbatim {
2175     overflow-x: auto ;
2176 }
2177
2178 .alltt {
2179     overflow-x: auto ;
2180 }
2181
2182
2183 .bverbatim {
2184     margin: 1ex 0pt 1ex 0pt ;
2185     padding: .5ex 0pt .5ex 0pt ;
2186     overflow-x: auto ;
2187 }
2188
2189 .lverbatim {
2190     margin: 1ex 0pt 1ex 0pt ;
2191     padding: .5ex 0pt .5ex 0pt ;
2192     overflow-x: auto ;
2193 }
2194
2195 .fancyvrb {
2196     font-size:.85em ;
2197     margin: 3ex 0pt 3ex 0pt
2198 }
2199
2200 .fancyvrblabel {
```

```
    font-weight:bold;
    text-align: center ;
}


.verse {
    font-family: "Linux Libertine Mono O", "Lucida Console",
        "Droid Sans Mono", "DejaVu Mono", "Bitstream Vera Mono",
        "Liberation Mono", "FreeMono", "Andale Mono",
        "Nimbus Mono L", "Courier New", monospace;
    margin-left: 1em ;
}


div.singlespace { line-height: 1.2 ; }
div.onehalfspace { line-height: 1.5 ; }
div.doublespace { line-height: 2 ; }


/* Word processor format output: */
div.wpfigure { border: 1px solid red ; margin: .5ex ; padding: .5ex ; }
div.wptable { border: 1px solid blue ; margin: .5ex ; padding: .5ex ; }
div.wpminipage { border: 1px solid green ; margin: .5ex ; padding: .5ex ;}




/* Minipage environments, vertically aligned to top, center, bottom: */
.minipage, .fminipage, .fcolorminipage {
    /* display: inline-block ; */
        /* Mini pages which follow each other will be tiled. */
    margin: .25em .25em .25em .25em;
    padding: .25em .25em .25em .25em;
    display: inline-flex;
    flex-direction: column ;
    overflow: auto;
}

/* Paragraphs in the flexbox did not collapse their margins. */
/* Have not yet researched this. */
.minipage p {margin: .75ex 0em .75ex 0em ;}

.fboxBlock .minipage, .colorbox .minipage, .colorboxBlock .minipage,
.fcolorbox .minipage, .fcolorboxBlock .minipage
    {border: none ; background: none;}

.fbox, .fboxBlock { border: 1px solid black ; }

.fbox, .fboxBlock, .fcolorbox, .fcolorboxBlock, .colorbox, .colorboxBlock,
.fminipage, .fcolorminipage
    {display: inline-block}

.shadowbox, .shabox {
   border: 1px solid black;
    box-shadow: 3px 3px 3px #808080 ;
```

```
2256      border-radius: 0px ;
2257     padding: .4ex .3em .4ex .3em ;
2258     margin: 0pt .3ex 0pt .3ex ;
2259   display: inline-block ;
2260 }
2261
2262 .doublebox {
2263    border: 3px double black;
2264      border-radius: 0px ;
2265     padding: .4ex .3em .4ex .3em ;
2266     margin: 0pt .3ex 0pt .3ex ;
2267   display: inline-block ;
2268 }
2269
2270 .ovalbox, .Ovalbox {
2271    border: 1px solid black;
2272      border-radius: 1ex ;
2273     padding: .4ex .3em .4ex .3em ;
2274     margin: 0pt .3ex 0pt .3ex ;
2275   display: inline-block ;
2276 }
2277
2278 .Ovalbox { border-width: 2px ; }
2279
2280 .framebox {
2281    border: 1px solid black;
2282      border-radius: 0px ;
2283     padding: .3ex .2em 0ex .2em ;
2284     margin: 0pt .1ex 0pt .1ex ;
2285   display: inline-block ;
2286 }
2287
2288
2289 .mdframed {
2290     padding: 0ex ;
2291     margin: 2ex 0em 2ex 0em ;
2292 }
2293
2294 .mdframed p { padding: 0ex .5em 0ex .5em ; }
2295
2296 .mdframed dl { padding: 1ex .5em 0ex .5em ; }
2297
2298 .mdframedtitle {
2299     padding: .5ex 0pt 0pt 0pt ;
2300     border-radius: 10pt 10pt 0pt 0pt ;
2301     display: block ;
2302     margin-bottom: 1ex ;
2303 }
2304
2305 .mdframedsubtitle {
2306     display: block ;
2307 }
2308
2309 .mdframedsubsubtitle {
2310     display: block ;
```

```
}

.mdtheorem {
    padding: 0ex .5em 0ex .5em ;
    margin: 3ex 5% 3ex 5% ;
}


/* framed package */
.framed, pre.boxedverbatim, fcolorbox {
    margin: 3ex 0em 3ex 0em ;
   border: 1px solid black;
     border-radius: 0px ;
    padding: .3ex 1em 0ex 1em ;
  display: block ;
}

.shaded {
    margin: 3ex 0em 3ex 0em ;
    padding: .3ex 1em .3ex 1em ;
    display: block ;
}

.snugframed {
    margin: 3ex 0em 3ex 0em ;
   border: 1px solid black;
     border-radius: 0px ;
  display: block ;
}

.framedleftbar {
    margin: 3ex 0em 3ex 0em ;
   border-left: 3pt solid black;
     border-radius: 0px ;
    padding: .3ex .2em .3ex 1em ;
  display: block ;
}

.framedtitle {
    margin: 0em ;
    padding: 0em ;
    font-size: 130%
}

.framedtitle p { padding: .3em }



dl {
  margin: 1ex 2em 1ex 0em;
  line-height: 1.3;
}

dl dt {
    display: block ;
```

```
    float:left ;
    font-weight: bold;
    padding-right: 1em ;
}

dl dd { display: block ; }

dl dd:after { content: "" ; display: block ; clear: both }

dl dd p { margin-top: 0em; }

dd ul, dd ol, dd dl { clear: both ; padding-top: 1ex }


nav {
    font-family: "URW Classico", Optima, "Linux Biolinum O",
        "DejaVu Sans", "Bitstream Vera Sans",
        Geneva, Verdana, sans-serif ;
    margin-bottom: 4ex ;
}

nav p {
    line-height: 1.2 ;
    margin-top:.5ex ;
    margin-bottom:.5ex;
    font-size: .9em ;
}



img, img.hyperimage, img.borderimage {
    max-width: 600px;
    border: 1px solid silver;
    box-shadow: 3px 3px 3px #808080 ;
    padding: .5% ;
    margin: .5% ;
    background: none ;
}

img.inlineimage{
    padding: 0px ;
    box-shadow: none ;
    border: none ;
    background: none ;
    margin: 0px ;
    display: inline-block ;
    border-radius: 0px ;
}

img.logoimage{
    max-width: 300px ;
    box-shadow: 3px 3px 3px #808080 ;
    border: 1px solid black ;
    background:none ;
    padding:0 ;
```

```
    margin:.5ex ;
    border-radius: 10px ;
}


.section {
/*
    To have each section float relative to each other:
*/
/*
    display: block ;
    float: left ;
    position: relative ;
    background: white ;
    border: 1px solid silver ;
    padding: .5em ;
*/
    margin: 0ex .5em 0ex .5em ;
    padding: 0 ;
}


figure {
    margin: 5ex auto 5ex auto ;
    padding: 1ex 1em 1ex 1em ;
    overflow-x: auto ;
}


/* To automatically center images in figures: */
/*
figure img.inlineimage {
    margin: 0ex auto 0ex auto ;
    display: block ;
}
*/

/* To automatically center minipages in figures: */
/*
figure div.minipage, figure div.minipage div.minipage {
    margin: 1ex auto 1ex auto ;
    display: block ;
}
*/

figure figure { margin: 0pt }

figure div.minipage p { font-size: 85% ; }

figure.subfigure, figure.subtable  {
    display: inline-block ; margin: 3ex 1em 3ex 1em ;
}

div.figurecaption .minipage { margin:0 ; padding: 0 }

```

```
div.minipage figure { border: none ; box-shadow: none ; }
div.minipage figure.table { margin: 0ex }
div.minipage div.footnotes { margin: 1ex 2em 0ex 2em }

div.floatrow { text-align: center; }

div.floatrow figure { display: inline-block ; margin: 1ex 2% ; }

div.floatfoot { font-size: .85em ;
    border-top: 1px solid silver ; line-height: 1.2 ; }

div.figurecaption , .lstlistingtitle {
    font-size: .85em ;
    text-align: center ;
    font-weight: bold ;
    margin-top: 1ex ;
    margin-bottom: 1ex ;
}

figure.subfigure div.figurecaption, figure.subtable div.figurecaption  {
    border-bottom: none ; background: none ;
}

div.nonfloatcaption {
    margin: 1ex auto 1ex auto ;
    font-size: .85em ;
    text-align: center ;
    font-weight: bold ;
}

/* For a \RawCaption inside a minipage inside a figure's floatrow: */
figure div.floatrow div.minipage div.figurecaption {
    border: none ;
    background: none ;
}


/* For packages such as float, rotfloat, and algorithm2e: */

figure.boxed, figure.boxruled {
    border: 1px solid black ;
}

figure.ruled {
    border-top: 1px solid black ;
    border-bottom: 1px solid black ;
    border-left: 0px ;
    border-right: 0px ;
    border-radius: 0px ;
    background: none ;
    box-shadow: none ;
}

figure.ruled div.figurecaption, figure.boxruled div.figurecaption {
    border-top: 1px solid silver ;
```

```
    border-bottom: 1px solid silver ;
}


table {
    margin: 1ex auto 1ex auto ;
    border-collapse: separate ;
    border-spacing: 0px ;
    line-height: 1.3 ;
    }

tr.hline td {border-top: 1px solid #808080 ; margin-top: 0ex ;
    margin-bottom: 0ex ; } /* for \hline */

tr.tbrule td {border-top: 1px solid black ; margin-top: 0ex ;
    margin-bottom: 0ex ; } /* for \toprule, \bottomrule */

td {padding: .5ex .5em .5ex .5em ;}

table td.tdl { text-align: left ; vertical-align: middle ; }
table td.tdc { text-align: center ; vertical-align: middle ; }
table td.tdat { text-align: center ; vertical-align: middle ; padding: 0px ; margin: 0px ; }
table td.tdbang { text-align: center ; vertical-align: middle ; }
table td.tdr { text-align: right ; vertical-align: middle ; }
table td.tdp { text-align: left ; vertical-align: bottom ; }
table td.tdm { text-align: left ; vertical-align: middle ; }
table td.tdb { text-align: left ; vertical-align: top ; }
table td.tdP { text-align: center ; vertical-align: bottom ; }
table td.tdM { text-align: center ; vertical-align: middle ; }
table td.tdB { text-align: center ; vertical-align: top ; }

table td.tvertbarl { border-left: 1px solid black }
table td.tvertbarldouble { border-left: 4px double black }
table td.tvertbarr { border-right: 1px solid black }
table td.tvertbarrdouble { border-right: 4px double black }

table td.tvertbarldash { border-left: 1px dashed black }
table td.tvertbarldoubledash { border-left: 2px dashed black }
table td.tvertbarrdash { border-right: 1px dashed black }
table td.tvertbarrdoubledash { border-right: 2px dashed black }


/* for cmidrules: */
table td.tdrule {
    border-top: 1px solid #A0A0A0 ;
}

table td.tdrulel {
    border-top-left-radius:.5em ;
    border-top: 1px solid #A0A0A0 ;
}

table td.tdruler {
    border-top-right-radius:.5em ;
    border-top: 1px solid #A0A0A0 ;
```

```
}

table td.tdrulelr {
    border-top-left-radius:.5em ;
    border-top-right-radius:.5em ;
    border-top: 1px solid #A0A0A0 ;
}


/* Margins of paragraphs inside table cells: */
td.tdp p , td.tdprule p , td.tdP p , td.tdPrule p { padding-top: 1ex ;
    padding-bottom: 1ex ; margin: 0ex ; }
td.tdm p , td.tmbrule p , td.tdM p , td.tdMrule p { padding-top: 1ex ;
    padding-bottom: 1ex ; margin: 0ex ; }
td.tdb p , td.tdbrule p , td.tdB p , td.tdBrule p { padding-top: 1ex ;
    padding-bottom: 1ex ; margin: 0ex ; }

td.tdp , td.tdprule , td.tdP , td.tdPrule
    { padding: 0ex .5em 0ex .5em ; }
td.tdm , td.tdmrule , td.tdM , td.tdMrule
    { padding: 0ex .5em 0ex .5em ; }
td.tdb , td.tdbrule , td.tdB , td.tdBrule
    { padding: 0ex .5em 0ex .5em ; }


/* table notes: */
.tnotes {
    margin: 0ex 5% 1ex 5% ;
    padding: 0.5ex 1em 0.5ex 1em;
    font-size:.80em;
    text-align: left ;
}

.tnotes dl dt p {margin-bottom:0px;}

.tnoteitemheader {margin-right: 1em;}


/* for colortbl and cell color */
div.cellcolor {
    width: 100% ;
    padding: .5ex .5em .5ex .5em ;
    margin: -.5ex -.5em -.5ex -.5em ;
}


/* for bigdelim */
.ldelim, .rdelim { font-size: 200% }


/* center, flushleft, flushright environments */
div.center{text-align:center;}
div.center table {margin-left:auto;margin-right:auto;}
div.flushleft{text-align:left;}
div.flushleft table {margin-left:0em ; margin-right:auto;}
```

```
div.flushright{text-align:right;}
div.flushright table {margin-left:auto ; margin-right: 0em ;}


/* Fancybox */
div.Btrivlist table tr td {
    padding: .2ex 0em ;
}


/* program listing callouts: */
span.callout {
     font-family: "DejaVu Sans", "Bitstream Vera Sans",
        Geneva, Verdana, sans-serif ;
    border-radius: .5em;
    background-color:black;
    color:white;
    padding:0px .25em 0px .25em;
    margin: 0 ;
    font-weight: bold;
    font-size:.72em ;
}

div.programlisting pre.verbatim span.callout{
    font-size: .85em ;
}

span.verbatim {
    font-family: "DejaVu Mono", "Bitstream Vera Mono", "Lucida Console",
        "Nimbus Mono L", "Liberation Mono", "FreeMono", "Andale Mono",
        "Courier New", monospace;
}



div.published
{
    text-align: center ;
    font-variant: normal ;
    font-style: italic ;
    font-size: 1em ;
    margin: 1ex 0em 1ex 0em ;
}

div.subtitle
{
    text-align: center ;
    font-variant: normal ;
    font-style: italic ;
    font-size: 1.25em ;
    margin: 1ex 0em 1ex 0em ;
}

div.subtitle p { margin: 1ex ; }

```

```
div.author
{
    font-variant: normal ;
    font-style: normal ;
    font-size: 1em ;
    margin: 1ex 0em 1ex 0em ;
}

div.oneauthor {
    display: inline-block ;
    margin: 0ex 1em 0ex 1em ;
}

/*
div.author table {
    margin: 1ex auto 0ex auto ;
    background: none ;
}

div.author table tbody tr td { padding: .25ex ; }
*/

span.affiliation {font-size: .85em ; font-variant: small-caps; }

div.titledate {
    text-align: center ;
    font-size: .85em ;
    font-style: italic;
    margin: 1ex 0em 1ex 0em ;
}


nav.topnavigation{
    text-align: left ;
    padding: 0.5ex 1em 0.5ex 1em ;
/*     margin: 2ex 0em 3ex 0em ; */
    margin: 0 ;
    border-bottom: 1px solid silver ;
    border-top: 1px solid silver ;
    clear:both ;
}

nav.botnavigation{
    text-align: left ;
    padding: 0.5ex 1em 0.5ex 1em ;
/*     margin: 3ex 0em 2ex 0em ; */
    margin: 0 ;
    border-top: 1px solid silver ;
    border-bottom: 1px solid silver ;
    clear:both ;
}


header {
    line-height: 1.2 ;
```

```
    font-size: 1em ;
    border-bottom: 1px solid silver ;
    margin: 0px ;
    padding: 2ex 1em 2ex 1em ;
    text-align:left ;
}


footer {
    font-size: .85em ;
    line-height: 1.2 ;
    margin-top: 1ex ;
    border-top: 1px solid silver ;
    padding: 2ex 1em 2ex 1em ;
    clear:both ;
    text-align:left ;
}


a.linkhome { font-weight:bold ; font-size: 1em ;}


div.lateximagesource { padding: 0px ; margin: 0px ; display: none; }

img.lateximage{
    padding: 0pt ;
    margin: 0pt ;
    box-shadow: none ;
    border: none ;
    background: none ;
    max-width: 100% ;
    border-radius: 0ex ;
    border: none ;
}


/* The -1px right margin compensates for the 1px right border. */
/* Without this -1px margin, the body container appears below instead */
/* of floating to the side. */
div.sidetoccontainer {
    font-family: "DejaVu Serif", "Bitstream Vera Serif",
        "Lucida Bright", Georgia, serif;
    float: left ;
    width: 20%;
    margin: 0pt -1px 3ex 0pt ;
    border-right: 1px solid silver;
    border-bottom: 1px solid silver;
    background: #FAF7F4 ;
    font-size:.9em ;
    border-radius: 0px 0px 20px 0px ;
}

div.sidetoccontents {
    overflow-y: auto ;
    width: 100% ;
```

```
    text-align: left ;
}


nav.sidetoc p {line-height:1.2 ; margin: 1ex .5em 1ex .5em ;
    text-indent: 0 ; }

nav.sidetoc p a {color:black ; font-size: .7em ;}

div.sidetoctitle {font-size: 1.2em; font-weight:bold; text-align:center;
    border-bottom: 1px solid silver ;    }

nav.sidetoc a:hover {text-decoration: underline ; }



section.textbody { margin: 0ex 1em 0ex 1em ;}


div.multicolsheading { -webkit-column-span: all;
    -moz-column-span: all; column-span: all; }
div.multicols { -webkit-columns: 3 380px ;
    -moz-columns: 3 380px ; columns: 3 380px ; }
div.multicols p {margin-top: 0ex}


/* Used for xfrac and nicefrac: */
span.numerator {
    font-size: 60% ;
    vertical-align: .4em ;
}

span.denominator {
    font-size: 60%
}


/* Used for algorithm2e: */
div.alg2evline{
    margin-left: 1em ;
    padding-left: 1em ;
    border-left: 1px solid black ;
    border-radius: 0px 0px 0px 1ex ;
}

div.alg2evsline{
    margin-left: 1em ;
    padding-left: 1em ;
    border-left: 1px solid black ;
}

div.alg2enoline{
    margin-left: 1em ;
    padding-left: 1em ;
}
```

```
2861
2862 span.alg2elinenumber{
2863     margin-right: .5em ;
2864     font-size: 50% ;
2865     color: red ;
2866 }
2867
2868
2869 /* Used for algorithmicx: */
2870 span.floatright { float: right ; }
2871
2872
2873
2874
2875 /* Native LaTeX theorems: */
2876
2877 .theoremcontents { font-style: italic; margin-top: 3ex ; margin-bottom: 3ex ; }
2878 .theoremlabel { font-style: normal; font-weight: bold ; margin-right: .5em ; }
2879
2880
2881 /* theorem, amsthm, and ntheorem packages */
2882
2883 span.theoremheader,
2884 span.theoremheaderplain,
2885 span.theoremheaderdefinition,
2886 span.theoremheaderbreak,
2887 span.theoremheadermarginbreak,
2888 span.theoremheaderchangebreak,
2889 span.theoremheaderchange,
2890 span.theoremheadermargin
2891 {
2892     font-style:normal ; font-weight: bold ; margin-right: 1em ;
2893 }
2894
2895 span.amsthmnameplain,
2896 span.amsthmnamedefinition,
2897 span.amsthmnumberplain,
2898 span.amsthmnumberdefinition
2899 {
2900     font-style:normal ; font-weight: bold ;
2901 }
2902
2903
2904 span.amsthmnameremark,
2905 span.amsthmnumberremark
2906 {font-style:italic ; font-weight: normal ; }
2907
2908
2909 span.amsthmnoteplain,
2910 span.amsthmnotedefinition
2911 {font-style:normal ;}
2912
2913
2914 span.theoremheaderremark,
2915 span.theoremheaderproof,
```

```
span.amsthmproofname
{font-style:italic ; font-weight: normal ; margin-right: 1em ; }

span.theoremheadersc
{
    font-style:normal ;
    font-variant: small-caps ;
    font-weight: normal ;
    margin-right: 1em ;
}

.theoremendmark {float:right}

div.amsthmbodyplain, div.theorembodyplain, div.theorembodynonumberplain,
div.theorembodybreak, div.theorembodynonumberbreak,
div.theorembodymarginbreak,
div.theorembodychangebreak,
div.theorembodychange,
div.theorembodymargin
{
    font-style:italic;
    margin-top: 3ex ; margin-bottom: 3ex ;
}

div.theorembodydefinition, div.theorembodyremark, div.theorembodyproof,
div.theorembodyplainupright, nonumberplainuprightsc,
div.amsthmbodydefinition, div.amsthmbodyremark,
div.amsthmproof
{
    font-style: normal ;
    margin-top: 3ex ; margin-bottom: 3ex ;
}

span.amsthmnoteremark {}


/* For the notes package: */
div.notesimportantnote, div.noteswarningnote, div.notesinformationnote {
    clear: both ;
    margin: 2ex 2em 2ex 2em ;
    border: 1px solid silver ;
}

div.notesicon {
    float:left ;
    display: inline-block ;
    background: gold ;
    padding: 0ex 1em 0ex 1em ;
    margin-right: 1em ;
    font-weight: bold ;
}

div.notescontents { font-style: italic }


```

```
/*
For CSS LaTeX and related logos:
Based on spacing demonstrated by the metafont package.
*/

.latexlogofont {
    font-family: "Linux Libertine O", "Nimbus Roman No 9 L",
        "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
    font-variant: normal ;
}

.latexlogo {
    font-family: "Linux Libertine O", "Nimbus Roman No 9 L",
        "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
    font-size: 1.1em;
}

.latexlogosup {
  text-transform: uppercase;
  letter-spacing: .03em ;
  font-size: 0.7em;
  vertical-align: 0.25em;
  margin-left: -0.4em;
  margin-right: -0.15em;
}

.latexlogosub {
  text-transform: uppercase;
  vertical-align: -0.27ex;
  margin-left: -0.08em;
  margin-right: -0.07em;
  font-size: 1em;
}

.latexlogotwoe {
  text-transform: none ;
  font-variant-numeric: oldstyle-nums ;
}

.latexlogotwoesub {
  font-style:italic ;
  vertical-align: -0.27ex;
  margin-left: -0.11em;
  margin-right: -0.1em;
  font-size: 1em;
}

.xelatexlogo {
    font-family: "Linux Libertine O", "Nimbus Roman No 9 L",
        "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
    letter-spacing: .03em ;
    font-size: 1.1em;
}

.xelatexlogosub {
```

```
  vertical-align: -0.27ex;
  margin-left: -0.0667em;
  margin-right: -.05em;
  font-size: 1em;
  letter-spacing: .03em ;
}

.amslogo {
    font-family: "TeXGyreChorus","URW Chancery L",
        "Apple Chancery","ITC Zapf Chancery","Monotype Corsiva",
        "Linux Libertine O", "Nimbus Roman No 9 L",  "FreeSerif",
        "Hoefler Text", Times, "Times New Roman", serif ;
    font-style: italic ;
}

.lyxlogo {
    font-family: "URW Classico", Optima, "Linux Biolinum O",
        "DejaVu Sans", "Bitstream Vera Sans", Geneva,
        Verdana, sans-serif ;
}


/* Only display top and bottom navigation if a small screen: */
/* Hide the sidetoc if a small screen: */
nav.topnavigation { display:none; }
nav.botnavigation { display:none; }

/* Only display the sidetoc's webpage title if a small screen */
span.sidetocthetitle { display: none }

@media screen and (max-width: 50em) {
    div.sidetoccontainer {
        float: none ;
        width: 100% ;
        padding: 0 ;
        border-radius: 0 ;
        border-bottom: 1px solid black ;
        border-top: 1px solid black ;
        box-shadow: none ;
    }
    span.sidetocthetitle { display: inline }
    nav.botnavigation { display:block }
    div.bodycontainer { width: 100% }
    .marginpar {
        max-width: 100%;
        float: none;
        display:block ;
        margin: 1ex 1em 1ex 1em ;
    }
}

@media print {
    body {
        font-family: "Linux Libertine O",
        "DejaVu Serif", "Bitstream Vera Serif",
```

```
        "Liberation Serif", "Nimbus Roman No 9 L",
        "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
    }
    div.sidetoccontainer { display:none; }
    nav.topnavigation { display: none; }
    nav.botnavigation { display: none; }
    div.bodycontainer { width: 100% }
}

@media handheld {
    div.sidetoccontainer { display:none; }
    nav.topnavigation { display:block }
    nav.botnavigation { display:block }
    div.bodycontainer { width: 100% }
}

@media projection {
    div.sidetoccontainer { display:none; }
    nav.topnavigation { display:block }
    nav.botnavigation { display:block }
    div.bodycontainer { width: 100% }
}
\end{filecontents*}
% \end{Verbatim}% for syntax highlighting
\end{LWRwriteconf}
```

40.5 lwarp_sagebrush.css

File lwarp_sagebrush.css 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:

```
\begin{LWRwriteconf}
\begin{filecontents*}{lwarp_sagebrush.css}
@import url("lwarp.css") ;


A:link {color:#105030 ; text-decoration: none ; }
A:visited {color:#705030 ; text-shadow:1px 1px 2px #a0a0a0;}
A:hover {color:#006000 ; text-decoration: underline ; text-shadow:0px 0px 2px #a0a0a0;}
A:active {color:#00C000 ; text-shadow:1px 1px 2px #a0a0a0;}



h1, h2, h3, h4, h5, h6, span.paragraph, span.subparagraph
{
    font-family: "URW Classico", Optima, "Linux Biolinum O",
        "Linux Libertine O", "Liberation Serif",
        "Nimbus Roman No 9 L",  "FreeSerif",
        "Hoefler Text", Times, "Times New Roman", serif;
    font-variant: small-caps ;
    font-weight: normal ;
    color: #304070 ;
```

```
3127     text-shadow: 2px 2px 3px #808080;
3128 }
3129
3130 h1 {    /* title of the entire website, used on each page */
3131     font-variant: small-caps ;
3132     color: #304070 ;
3133     text-shadow: 2px 2px 3px #808080;
3134     background-color: #F7F7F0 ;
3135     background-image: linear-gradient(to bottom, #F7F7F0, #C0C0C4);
3136 }
3137
3138 h1 {
3139   border-bottom: 1px solid #304070;
3140 /*  border-top: 2px solid #304070; */
3141 }
3142
3143 h2 {
3144   border-bottom: 1px solid #304070;
3145 /*  border-top: 2px solid #304070; */
3146     background-color: #F7F7F0 ;
3147     background-image: linear-gradient(to bottom, #F7F7F0, #DAD0C0);
3148 }
3149
3150
3151
3152 div.abstract {
3153     background: #f5f5eb ;
3154     background-image: linear-gradient(to bottom, #f5f5eb, #C8C8B8);
3155
3156   border: 1px solid silver;
3157     border-radius: 1em ;
3158 }
3159
3160 div.abstract dl {line-height:1.5;}
3161 div.abstract dt {color:#304070;}
3162
3163 div.abstracttitle{
3164     font-family: "URW Classico", Optima, "Linux Biolinum O",
3165         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
3166         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
3167     font-weight:bold;
3168     font-variant: small-caps ;
3169     font-size:1.5em;
3170     border-bottom: 1px solid silver ;
3171     color: #304070 ;
3172     text-align: center ;
3173     text-shadow: 1px 1px 2px #808080;
3174 }
3175
3176 span.abstractrunintitle{
3177     font-family: "URW Classico", Optima, "Linux Biolinum O",
3178         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
3179         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
3180     font-weight:bold;
3181 }
```

```
3182
3183
3184 div.epigraph, div.dictum {
3185     background: #f5f5eb ;
3186     background-image: linear-gradient(to bottom, #f5f5eb, #C8C8B8);
3187
3188     border: 1px solid silver ;
3189     border-radius: 1ex ;
3190     box-shadow: 3px 3px 3px #808080 ;
3191 }
3192
3193
3194 .example {
3195     background-color: #f5f5eb ;
3196     background-image: linear-gradient(to bottom, #f5f5eb, #C8C8B8);
3197
3198 }
3199
3200 div.exampletitle{
3201     font-family: "URW Classico", Optima, "Linux Biolinum O",
3202         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
3203         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
3204     font-weight:bold;
3205     font-variant: small-caps ;
3206     border-bottom: 1px solid silver ;
3207     color: #304070 ;
3208     text-align: center ;
3209     text-shadow: 1px 1px 2px #808080;
3210 }
3211
3212
3213 .sidebar {
3214     background-color: #f5f5eb ;
3215     background-image: linear-gradient(to bottom, #f5f5eb, #C8C8B8);
3216
3217 }
3218
3219 div.sidebartitle{
3220     font-family: "URW Classico", Optima, "Linux Biolinum O",
3221         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
3222         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
3223     font-weight:bold;
3224     font-variant: small-caps ;
3225     border-bottom: 1px solid silver ;
3226     color: #304070 ;
3227     text-align: center ;
3228     text-shadow: 1px 1px 2px #808080;
3229 }
3230
3231
3232 .fancyvrblabel {
3233     font-family: "URW Classico", Optima, "Linux Biolinum O",
3234         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
3235         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
3236     font-weight:bold;
```

```
3237     font-variant: small-caps ;
3238     font-size: 1.5em ;
3239     color: #304070 ;
3240     text-align: center ;
3241     text-shadow: 1px 1px 2px #808080;
3242 }
3243
3244 div.minipage {
3245     background-color: #eeeee7 ;
3246     border: 1px solid silver ;
3247     border-radius: 1ex ;
3248 }
3249
3250 table div.minipage { background: none ; border: none ; }
3251
3252 div.framebox div.minipage {border:none ; background:none}
3253
3254 section.textbody > div.minipage {
3255     box-shadow: 3px 3px 3px #808080 ;
3256 }
3257
3258 div.fboxBlock div.minipage { box-shadow: none ; }
3259
3260 .framed .minipage , .framedleftbar .minipage {
3261     border: none ;
3262     background: none ;
3263     padding: 0ex ;
3264     margin: 0ex ;
3265 }
3266
3267 figure.figure .minipage, div.figurecaption .minipage { border: none; }
3268
3269 div.marginblock div.minipage ,
3270 div.marginparblock div.minipage
3271     { border: none; }
3272
3273 figure , div.marginblock {
3274     background-color: #eeeee7 ;
3275     border: 1px solid silver ;
3276     border-radius: 1ex ;
3277     box-shadow: 3px 3px 3px #808080 ;
3278 }
3279
3280 figure figure {
3281     border: 1px solid silver ;
3282     margin: 0em ;
3283     box-shadow: none ;
3284 }
3285
3286 /*
3287 div.figurecaption {
3288     border-top: 1px solid silver ;
3289     border-bottom: 1px solid silver ;
3290     background-color: #e8e8e8 ;
3291 }
```

```
*/


div.table {
    box-shadow: 3px 3px 3px #808080 ;
}

/*
.tnotes {
    background: #e8e8e8;
    border: 1px solid silver;
}
*/


nav.topnavigation{
    background-color: #b0b8b0 ;
    background-image: linear-gradient(to bottom,#e0e0e0,#b0b8b0) ;
}

nav.botnavigation{
    background-color: #b0b8b0 ;
    background-image: linear-gradient(to top,#e0e0e0,#b0b8b0) ;
}



header{
    background-color: #F7F7F0 ;
    background-image: linear-gradient(to top, #F7F7F0, #b0b8b0);
}

footer{
    background-color: #F7F7F0 ;
    background-image: linear-gradient(to bottom, #F7F7F0, #b0b8b0);
}



div.sidetoccontainer {
    background-color: #F7F7F0 ;
    background-image: linear-gradient(to bottom, #F7F7F0, #C0C0C0);
    box-shadow: 3px 3px 3px #808080 ;
    }

div.sidetoctitle {color: #304070 ; }

nav.sidetoc a:hover {
    color:#006000 ;
    text-decoration: none ;
    text-shadow:0px 0px 2px #a0a0a0;
}


@media screen and (max-width: 45em) {
```

```
    div.sidetoccontainer { border-radius: 0 ; }
}


\end{filecontents*}
% \end{Verbatim}% for syntax highlighting
\end{LWRwriteconf}
```

40.6 lwarp_formal.css

File lwarp_formal.css 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:

```
\begin{LWRwriteconf}
\begin{filecontents*}{lwarp_formal.css}
@import url("lwarp.css") ;



A:link {color:#802020 ; text-decoration:none; }
A:visited {color:#802020 ; text-shadow:none ;}
A:hover {color:#400000 ; text-shadow:none ;}
A:active {color:#C00000 ; text-shadow:none ;}


body {
    font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
        "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
        "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
        "Times New Roman", serif;
    background: #fffcf5;
}

span.textrm {
    font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
        "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
        "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
        "Times New Roman", serif;
}

span.textsf {
     font-family: "DejaVu Sans", "Bitstream Vera Sans",
        Geneva, Verdana, sans-serif ;
}



h1, h2, h3, h4, h5, h6, span.paragraph, span.subparagraph
{
    font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
        "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
        "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
```

```
3393         "Times New Roman", serif;
3394     color: #800000 ;
3395     text-shadow: none ;
3396 }
3397
3398 h1, h2 {
3399     background-color: #fffcf5 ;
3400     background-image: none ;
3401     border-bottom: 1px solid #808080;
3402 /*    border-top: 2px solid #808080; */
3403 }
3404
3405 div.abstracttitle {
3406     font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
3407         "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
3408         "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
3409         "Times New Roman", serif;
3410     color: black ;
3411     text-shadow: none ;
3412 }
3413
3414 span.abstractrunintitle {
3415     font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
3416         "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
3417         "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
3418         "Times New Roman", serif;
3419     color: black ;
3420     text-shadow: none ;
3421 }
3422
3423 div.abstract { font-size: 100% }
3424
3425 .sidebar {
3426     background: #fffcf5;
3427     background-image: none ;
3428   margin: 2em 5% 2em 5%;
3429   padding: 0.5em 1em;
3430   border: none ;
3431   border-top : 1px solid silver;
3432   border-bottom : 1px solid silver;
3433   font-size: 90% ;
3434 }
3435
3436 div.sidebartitle{
3437     font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
3438         "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
3439         "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
3440         "Times New Roman", serif;
3441     color: #800000 ;
3442     text-shadow: none ;
3443     border: none ;
3444 }
3445
3446 .example {
3447     background: #fffcf5;
```

```
3448     background-image: none ;
3449   margin: 2em 5% 2em 5%;
3450   padding: 0.5em 1em;
3451   border: none ;
3452   border-top : 1px solid silver;
3453   border-bottom : 1px solid silver;
3454 }
3455
3456 div.exampletitle{
3457     font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
3458         "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
3459         "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
3460         "Times New Roman", serif;
3461     color: #800000 ;
3462     text-shadow: none ;
3463     border: none ;
3464 }
3465
3466 div.fancyvrblabel{
3467     font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
3468         "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
3469         "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
3470         "Times New Roman", serif;
3471     color: #800000 ;
3472     text-shadow: none ;
3473     border: none ;
3474 }
3475
3476
3477
3478 figure {
3479     margin: 5ex 5% 5ex 5% ;
3480     padding: 1ex 1em 1ex 1em ;
3481     background-color: #fffcf5 ;
3482     overflow-x: auto ;
3483     border: none ;
3484 /*     border-top: 1px solid silver; */
3485 /*     border-bottom: 1px solid silver; */
3486 }
3487
3488
3489 div.figurecaption , .lstlisting {
3490     border: none ;
3491 /*     border-top: 1px solid silver ; */
3492 /*     border-bottom: 1px solid silver ; */
3493     background-color: #fffcf5 ;
3494 }
3495
3496 .tnotes {
3497     background: #fffcf5 ;
3498     border-top: 1px solid silver ;
3499     border-bottom: 1px solid silver ;
3500 }
3501
3502 .theorem {
```

```
3503         background: none ;
3504 }
3505
3506 .minipage {
3507     background-color: #fffcf5 ;
3508     border: none ;
3509 }
3510
3511 div.floatrow figure { border: none ; }
3512
3513 figure figure { border: none ; }
3514
3515
3516 nav.toc, nav.lof, nav.lot, nav.lol {
3517     font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
3518         "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
3519         "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
3520         "Times New Roman", serif;
3521 }
3522
3523 div.sidetoccontainer {
3524     font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
3525         "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
3526         "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
3527         "Times New Roman", serif;
3528     background-image: linear-gradient(to bottom, #fffcf5, #C0C0C0);
3529 }
3530
3531 div.sidetoctitle{
3532     color: #800000 ;
3533 }
3534
3535 header{
3536     background-color: #e0e0e0 ;
3537     background-image: linear-gradient(to top, #fffcf5, #b0b0b0);
3538     text-align:center ;
3539 }
3540
3541 footer{
3542     background-color: #e0e0e0 ;
3543     background-image: linear-gradient(to bottom, #fffcf5, #b0b0b0);
3544     padding: 2ex 1em 2ex 1em ;
3545     text-align:left ;
3546 }
3547
3548 nav.botnavigation {
3549     background: #dedcd5 ;
3550     border-top: 1px solid black ;
3551 }
3552 \end{filecontents*}
3553 % \end{Verbatim}% for syntax highlighting
3554 \end{LWRwriteconf}
```

40.7 sample_project.css

File sample_project.css 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:
```
\begin{LWRwriteconf}
\begin{filecontents*}{sample_project.css}
/* ( --- 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 --- ) */
\end{filecontents*}
% \end{Verbatim}% for syntax highlighting
\end{LWRwriteconf}
```

40.8 lwarp.ist

File lwarp.ist 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:
```
\begin{LWRwriteconf}
\begin{filecontents*}{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 "}"
page_compositor "."
\end{filecontents*}
% \end{Verbatim}% for syntax highlighting
\end{LWRwriteconf}
```

40.9 lwarp.xdy

File lwarp.xdy 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:

```
\begin{LWRwriteconf}
\begin{filecontents*}{lwarp.xdy}
(require "tex/inputenc/latin.xdy")
(merge-rule "\\PS *" "Postscript")
(require "texindy.xdy")
(require "page-ranges.xdy")
(require "book-order.xdy")
(define-location-class "arabic-page-numbers"
    ("arabic-numbers") :min-range-length 1)
(require "makeindex.xdy")
(define-attributes (("hyperindexref")))
(markup-locref :open "\hyperindexref{" :close "}")
(markup-locref :open "\hyperindexref{" :close "}" :attr "hyperpage")
(markup-locref :open "\textbf{\hyperindexref{" :close "}}" :attr "textbf")
(markup-locref :open "\textit{\hyperindexref{" :close "}}" :attr "textit")
(define-location-class-order ("roman-page-numbers"
                  "arabic-page-numbers"
                  "alpha-page-numbers"
                  "Roman-page-numbers"
                  "Alpha-page-numbers"
                  "see"
                  "seealso"))
\end{filecontents*}
% \end{Verbatim}% for syntax highlighting
\end{LWRwriteconf}
```

40.10 lwarp_one_limage.cmd

File lwarp_one_limage.cmd Used by lwarp to help make `latex`images 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:

```
\begin{LWRwriteconf}
\immediate\openout\LWR@quickfile=lwarp_one_limage.txt
\immediate\write\LWR@quickfile{%
    pdfseparate -f \LWRpercent 1 -l \LWRpercent 1 \LWRpercent 4_html.pdf %
   \LWR@ImagesDirectory\OSPathSymbol lateximagetemp-\LWRpercent\LWRpercent d.pdf%
}
\immediate\write\LWR@quickfile{%
   pdfcrop --hires  \LWR@ImagesDirectory\OSPathSymbol lateximagetemp-\LWRpercent 1.pdf %
    \LWR@ImagesDirectory\OSPathSymbol\LWRpercent 3.pdf%
}
\immediate\write\LWR@quickfile{%
   pdftocairo  -svg -noshrink  \LWR@ImagesDirectory\OSPathSymbol\LWRpercent 3.pdf %
    \LWR@ImagesDirectory\OSPathSymbol\LWRpercent 3.svg%
}
\immediate\write\LWR@quickfile{%
    del \LWR@ImagesDirectory\OSPathSymbol\LWRpercent 3.pdf%
}
\immediate\write\LWR@quickfile{%
    del \LWR@ImagesDirectory\OSPathSymbol lateximagetemp-\LWRpercent 1.pdf%
}
\immediate\write\LWR@quickfile{exit}
\immediate\closeout\LWR@quickfile
\end{LWRwriteconf}
```

40.11 `lwarp_mathjax.txt`

File `lwarp_mathjax.txt` Used by lwarp when using MathJax.

This must be present when compiling the project, but does not need to be present when distributing the resulting HTML files.

Config file:

```
\begin{LWRwriteconf}
\begin{filecontents*}{lwarp_mathjax.txt}
<!-- https://groups.google.com/forum/#!topic/
                                mathjax-users/jUtewUcE2bY -->
<script type="text/x-mathjax-config">
MathJax.Hub.Register.StartupHook("TeX AMSmath Ready",function () {
    var seteqsectionDefault = {name: "", num: 0};
    var seteqsections = {}, seteqsection = seteqsectionDefault;
    var TEX = MathJax.InputJax.TeX, PARSE = TEX.Parse;
    var AMS = MathJax.Extension["TeX/AMSmath"];
    TEX.Definitions.Add({
    macros: {
        seteqsection: "mySection",
        seteqnumber: "mySetEqNumber"
    }
    });
```

```
3660
3661    PARSE.Augment({
3662    mySection: function (name) {
3663        seteqsection.num = AMS.number;
3664        var n = this.GetArgument(name);
3665        if (n === "") {
3666            seteqsection = seteqsectionDefault;
3667        } else {
3668            if (!seteqsections["_"+n])
3669                seteqsections["_"+n] = {name:n, num:0};
3670            seteqsection = seteqsections["_"+n];
3671        }
3672        AMS.number = seteqsection.num;
3673    },
3674    mySetEqNumber: function (name) {
3675        var n = this.GetArgument(name);
3676        if (!n || !n.match(/^ *[0-9]+ *$/))
3677            n = "";
3678        else
3679            n = parseInt(n)-1;
3680        <!-- $ syntax highlighting -->
3681        if (n === "" || n < 1)
3682            TEX.Error
3683            ("Argument to "+name+" should be a positive integer");
3684        AMS.number = n;
3685    }
3686    });
3687    MathJax.Hub.Config({
3688    TeX: {
3689        equationNumbers: {
3690            formatTag: function (n) {
3691                <!-- if not numeric, don't include the chapter -->
3692                if (!n.match(/^ *[0-9]+ *$/ ))
3693                <!-- $ syntax highlighting -->
3694                    return "("+(n).replace(/^\./,"")+")" ;
3695                else
3696                  return "("+(seteqsection.name+"."+n).replace(/^\./,"")+")" ;
3697            },
3698            formatID: function (n) {
3699                n = (seteqsection.name+'.'+n).replace
3700                    (/[:"'<>&]/g,"").replace(/^\./,"");
3701                return 'mjx-eqn-' + n;
3702            }
3703        }
3704    }
3705    });
3706 });
3707 </script>
3708
3709 <!-- http://docs.mathjax.org/en/latest/options/ThirdParty.html -->
3710 <script type="text/x-mathjax-config">
3711   MathJax.Ajax.config.path["Contrib"] =
3712     "https://cdn.mathjax.org/mathjax/contrib";
3713 </script>
3714
```

```
<script type="text/x-mathjax-config">
MathJax.Hub.Config({
    TeX: {
    equationNumbers: {
        autoNumber: "AMS"
    }
    }
});
</script>

<!-- Alternative CDN provider: -->
<script type="text/javascript" async
src="https://cdnjs.cloudflare.com/ajax/libs/mathjax/2.7.4/MathJax.js?config=TeX-AMS_HTML-full">
</script>

<!-- No longer supported after April 30, 2017: -->
<!--
<script
 src="https://cdn.mathjax.org/mathjax/latest/MathJax.js?config=TeX-AMS_HTML-full">
</script>
-->

\end{filecontents*}
% \end{Verbatim}% for syntax highlighting
\end{LWRwriteconf}
```

40.12 `lwarpmk.lua` — `lwarpmk` option

Opt `lwarpmk` Creates a local copy of *lwarpmk*.

Prog `lwarpmk` Command-line utility to process lwarp files and images.

parallel processing `latexi`mages 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.

```
\begin{LWR@createlwarpmk}

\begin{filecontents*}{lwarpmk.lua}
#!/usr/bin/env texlua

-- Copyright 2016-2018 Brian Dunn


```

```
printversion = "v0.67"
requiredconfversion = "2" -- also at *lwarpmk.conf

function printhelp ()
print ("lwarpmk: Use lwarpmk -h or lwarpmk --help for help.") ;
end


function printusage ()
--
-- Print the usage of the lwarpmk command:
--
print ( [[

lwarpmk print [-p project]: Compile the print version if necessary.
lwarpmk print1 [-p project]: Forced single compile of the print version.
lwarpmk printindex [-p project]: Process print indexes.
lwarpmk printglossary [-p project]: Process the glossary for the print version.
lwarpmk html [-p project]: Compile the HTML version if necessary.
lwarpmk html1 [-p project]: Forced single compile of the HTML version.
lwarpmk htmlindex [-p project]: Process HTML indexes.
lwarpmk htmlglossary [-p project]: Process the glossary for the html version.
lwarpmk again [-p project]: Touch the source code to trigger recompiles.
lwarpmk limages [-p project]: Process the "lateximages" created by lwarp.sty.
lwarpmk pdftohtml [-p project]:
    For use with latexmk or a Makefile:
    Converts project_html.pdf to project_html.html and individual HTML files.
    Finishes the HTML conversion even if there was a compile error.
lwarpmk pdftosvg <list of file names>: Converts each PDF file to SVG.
lwarpmk epstopdf <list of file names>: Converts each EPS file to PDF.
lwarpmk clean [-p project]: Remove *.aux, *.toc, *.lof/t,
    *.idx, *.ind, *.log, *_html_inc.*, .gl*
lwarpmk cleanall [-p project]: Remove auxiliary files, project.pdf, *.html
lwarpmk cleanlimages: Removes all images from the "lateximages" directory.
lwarpmk -h: Print this help message.
lwarpmk --help: Print this help message.

]] )
-- printconf ()
end


function splitfile (destfile,sourcefile)
--
-- Split one large sourcefile into a number of files,
-- starting with destfile.
-- The file is split at each occurance of <!--|Start file|newfilename|*
--
print ("lwarpmk: Splitting " .. sourcefile .. " into " .. destfile) ;
local sfile = io.open(sourcefile)
io.output(destfile)
for line in sfile:lines() do
i,j,copen,cstart,newfilename = string.find (line,"(.*)|(.*)|(.*)|") ;
if ( (i~= nil) and (copen == "<!--") and (cstart == "Start file")) then
    -- split the file
```

```
    io.output(newfilename) ;
else
    -- not a splitpoint
    io.write (line .. "\n") ;
end
end -- do
io.close(sfile)
end -- function


function cvalueerror ( line, linenum , cvalue )
--
-- Incorrect value, so print an error and exit.
--
    print ("lwarpmk: ===")
    print ("lwarpmk: " .. linenum .. " : " .. line ) ;
    print (
        "lwarpmk: incorrect variable value \"" .. cvalue ..
        "\" in lwarpmk.conf.\n"
    ) ;
    print ("lwarpmk: ===")
--    printconf () ;
    os.exit(1) ;
end


function ignoreconf ()
-- Global argument index
argindex = 2
end

function loadconf ()
--
-- Load settings from the project's "lwarpmk.conf" file:
--
-- Default configuration filename:
local conffile = "lwarpmk.conf"
local confroot = "lwarpmk"
-- Global argument index
argindex = 2
-- Optional configuration filename:
if ( arg[argindex] == "-p" ) then
    argindex = argindex + 1
    confroot = arg[argindex]
    conffile = confroot..".lwarpmkconf"
    argindex = argindex + 1
end
-- Additional defaults:
confversion = "0"
opsystem = "Unix"
imagesdirectory = "lateximages"
imagesname = "image-"
latexmk = "false"
printlatexcmd = ""
HTMLlatexcmd = ""
```

```
printindexcmd = ""
HTMLindexcmd = ""
latexmkindexcmd = ""
-- to be removed:
-- indexprog = "makeindex"
-- makeindexstyle = "lwarp.ist"
-- xindylanguage = "english"
-- xindycodepage = "utf8"
-- xindystyle = "lwarp.xdy"
-- pdftotextenc = "UTF-8"
glossarycmd = "makeglossaries"
-- Verify the file exists:
if (lfs.attributes(conffile,"mode")==nil) then
    -- file not exists
    print ("lwarpmk: ===")
    print ("lwarpmk: File \"" .. conffile .."\" does not exist.")
    print ("lwarpmk: Move to the project's source directory,")
    print ("lwarpmk: recompile using pdflatex, xelatex, or lualatex,")
    print ("lwarpmk: then try using lwarpmk again.")
    if ( arg[argindex] ~= nil ) then
        print (
            "lwarpmk: (\"" .. confroot ..
            "\" does not appear to be a project name.)"
        )
    end
    print ("lwarpmk: ===")
    printhelp () ;
    os.exit(1) -- exit the entire lwarpmk script
else -- file exists
-- Read the file:
print ("lwarpmk: Reading " .. conffile ..".")
local cfile = io.open(conffile)
-- Scan each line, parsing each line as: name = [[string]]
local linenum = 0
for line in cfile:lines() do -- scan lines
linenum = linenum + 1
i,j,cvarname,cvalue = string.find (line,"([%w-_]*)%s*=%s*%[%[([^%]]*)%]%]") ;
-- Error if incorrect enclosing characters:
if ( i == nil ) then
    print ("lwarpmk: ===")
    print ("lwarpmk: " ..  linenum .. " : " .. line ) ;
    print ("lwarpmk: Incorrect entry in " .. conffile ..".\n" ) ;
    print ("lwarpmk: ===")
--    printconf () ;
    os.exit(1) ;
end -- nil
if ( cvarname == "confversion" ) then
    confversion = cvalue
elseif ( cvarname == "opsystem" ) then
    -- Verify choice of opsystem:
    if ( (cvalue == "Unix") or (cvalue == "Windows") ) then
        opsystem = cvalue
    else
        cvalueerror ( line, linenum , cvalue )
    end
```

```
elseif ( cvarname == "sourcename" ) then sourcename = cvalue
elseif ( cvarname == "homehtmlfilename" ) then homehtmlfilename = cvalue
elseif ( cvarname == "htmlfilename" ) then htmlfilename = cvalue
elseif ( cvarname == "imagesdirectory" ) then imagesdirectory = cvalue
elseif ( cvarname == "imagesname" ) then imagesname = cvalue
elseif ( cvarname == "latexmk" ) then latexmk = cvalue
elseif ( cvarname == "printlatexcmd" ) then printlatexcmd = cvalue
elseif ( cvarname == "HTMLlatexcmd" ) then HTMLlatexcmd = cvalue
elseif ( cvarname == "printindexcmd" ) then printindexcmd = cvalue
elseif ( cvarname == "HTMLindexcmd" ) then HTMLindexcmd = cvalue
elseif ( cvarname == "latexmkindexcmd" ) then latexmkindexcmd = cvalue
elseif ( cvarname == "glossarycmd" ) then glossarycmd = cvalue
elseif ( cvarname == "pdftotextenc" ) then pdftotextenc = cvalue
else
    print ("lwarpmk: ===")
    print ("lwarpmk: " .. linenum .. " : " .. line ) ;
    print (
        "lwarpmk: Incorrect variable name \"" .. cvarname .. "\" in " ..
        conffile ..".\n"
    ) ;
    print ("lwarpmk: ===")
--    printconf () ;
os.exit(1) ;
end -- cvarname
end -- do scan lines
io.close(cfile)
end -- file exists
-- Error if sourcename is "lwarp".
-- This could happen if a local copy of lwarp has recently been recompiled.
if sourcename=="lwarp" then
    print ("lwarpmk: ===")
   print ("lwarpmk: lwarp.sty has recently been recompiled in this directory,")
   print ("lwarpmk: and \"lwarpmk.conf\" is no longer set for your own project.")
   print ("lwarpmk: Recompile your own project using pdf/lua/xelatex <projectname>.")
   print ("lwarpmk: After a recompile, \"lwarpmk.conf\" will be set for your project,")
    print ("lwarpmk: and you may again use lwarpmk.")
    print ("lwarpmk: ===")
    os.exit(1)
end -- sourcename of "lwarp"
-- Select some operating-system commands:
if opsystem=="Unix" then  -- For Unix / Linux / Mac OS:
    rmname = "rm"
    mvname = "mv"
    cpname = "cp"
    touchnamepre = "touch"
    touchnamepost = ""
    newtouchname = "touch"
    dirslash = "/"
    opquote= "\'"
    cmdgroupopenname = " ( "
    cmdgroupclosename = " ) "
    seqname = " && "
    bgname = " &"
elseif opsystem=="Windows" then -- For Windows
    rmname = "DEL"
```

```
    mvname = "MOVE"
    cpname = "COPY"
    touchnamepre = "COPY /b"
    touchnamepost = "+,,"
    newtouchname = "echo empty >"
    dirslash = "\\"
    opquote= "\""
    cmdgroupopenname = ""
    cmdgroupclosename = ""
    seqname = " & "
    bgname = ""
else
    print ("lwarpmk: ===")
    print ("lwarpmk: Select Unix or Windows for opsystem." )
    print ("lwarpmk: ===")
    os.exit(1)
end --- for Windows
-- Warning if the operating system does not appear to be correct,
-- in case files were transferred to another system.
if ( (package.config:sub(1,1)) ~= dirslash ) then
    print ("lwarpmk: ===")
   print ("lwarpmk: It appears that lwarpmk.conf is for a different operating system." )
    print ("lwarpmk: To adjust lwarpmk.conf for the current operating system," )
    print ("lwarpmk:   recompile the original document using xe/lua/pdflatex." )
    print ("lwarpmk: ")
    print ("lwarpmk: lwarpmk shall attempt to continue...")
    print ("lwarpmk: ===")
end
-- Error if the configuration file's version is not current:
if ( confversion ~= requiredconfversion ) then
    print ("lwarpmk: ===")
   print ("lwarpmk: The configuration files lwarpmk.conf and "..sourcename..".lwarpmkconf" )
    print ("lwarpmk:   must be updated.  To update the configuration files," )
    print ("lwarpmk:   recompile "..sourcename..".tex using xe/lua/pdflatex," )
    print ("lwarpmk:   then use lwarpmk again.")
    print ("lwarpmk: ===")
    os.exit(1)
end
end -- loadconf


function executecheckerror ( executecommands , errormessage )
--
-- Execute an operating system call,
-- and maybe exit with an error message.
--
local err
err = os.execute ( executecommands )
if ( err ~= 0 ) then
    print ("lwarpmk: ===")
    print ("lwarpmk: " .. errormessage )
    print ("lwarpmk: ===")
    os.exit(1)
end
end -- executecheckerror
```

```
4022
4023
4024 function refreshdate ()
4025 os.execute(touchnamepre .. " " .. sourcename .. ".tex " .. touchnamepost)
4026 end
4027
4028
4029
4030 function reruntoget (filesource)
4031 --
4032 -- Scan the LaTeX log file for the phrase "Rerun to get",
4033 -- indicating that the file should be compiled again.
4034 -- Return true if found.
4035 --
4036 local fsource = io.open(filesource)
4037 for line in fsource:lines() do
4038 if ( string.find(line,"Rerun to get") ~= nil ) then
4039     io.close(fsource)
4040     return true
4041 end -- if
4042 end -- do
4043 io.close(fsource)
4044 return false
4045 end
4046
4047
4048
4049 function onetime (latexcmd, fsuffix)
4050 --
4051 -- Compile one time, return true if should compile again.
4052 -- fsuffix is "" for print, "_html" for HTML output.
4053 --
4054 print("lwarpmk: Compiling with: " .. latexcmd)
4055 executecheckerror (
4056     latexcmd ,
4057     "Compile error."
4058 )
4059 return (reruntoget(sourcename .. fsuffix .. ".log") ) ;
4060 end
4061
4062
4063 function manytimes (latexcmd, fsuffix)
4064 --
4065 -- Compile up to five times.
4066 -- fsuffix is "" for print, "_html" for HTML output
4067 --
4068 if onetime(latexcmd, fsuffix) == true then
4069 if onetime(latexcmd, fsuffix) == true then
4070 if onetime(latexcmd, fsuffix) == true then
4071 if onetime(latexcmd, fsuffix) == true then
4072 if onetime(latexcmd, fsuffix) == true then
4073 end end end end end
4074 end
4075
4076
```

```
function verifyfileexists (filename)
--
-- Exit if the given file does not exist.
--
if (lfs.attributes ( filename , "modification" ) == nil ) then
    print ("lwarpmk: ===")
    print ("lwarpmk: " .. filename .. " not found." ) ;
    print ("lwarpmk: ===")
    os.exit (1) ;
end
end



function pdftohtml ()
--
-- Convert <project>_html.pdf into HTML files:
--
-- Convert to text:
print ("lwarpmk: Converting " .. sourcename
    .."_html.pdf to " .. sourcename .. "_html.html")
os.execute("pdftotext  -enc " .. pdftotextenc .. "  -nopgbrk  -layout "
    .. sourcename .. "_html.pdf " .. sourcename .. "_html.html")
-- Split the result into individual HTML files:
splitfile (homehtmlfilename .. ".html" , sourcename .. "_html.html")
end


function removeaux ()
--
-- Remove auxiliary files:
-- All .aux files are removed since there may be many bbl*.aux files.
--
os.execute ( rmname .. " *.aux " ..
    sourcename ..".toc " .. sourcename .. "_html.toc " ..
    sourcename ..".lof " .. sourcename .. "_html.lof " ..
    sourcename ..".lot " .. sourcename .. "_html.lot " ..
    " *.idx " ..
    " *.ind " ..
    sourcename ..".ps " .. sourcename .."_html.ps " ..
    sourcename ..".log " .. sourcename .. "_html.log " ..
    sourcename ..".gl* " .. sourcename .. "_html.gl* " ..
    " *_html_inc.* "
    )
end

function checkhtmlpdfexists ()
--
-- Error if the HTML document does not exist.
-- The lateximages are drawn from the HTML PDF version of the document,
-- so "lwarpmk html" must be done before "lwarpmk limages".
--
local htmlpdffile = io.open(sourcename .. "_html.pdf", "r")
if ( htmlpdffile == nil ) then
    print ("")
```

```
4132     print ("lwarpmk: ===")
4133     print ("lwarpmk: The HTML version of the document does not exist.")
4134     print ("lwarpmk: Enter \"lwarpmk html\" to compile the HTML version.")
4135     print ("lwarpmk: ===")
4136     os.exit(1)
4137 end
4138 io.close (htmlpdffile)
4139 end -- checkhtmlpdfexists
4140
4141
4142 function warnlimages ()
4143 --
4144 -- Warning of a missing <sourcename>-images.txt file:
4145     print ("lwarpmk: ===")
4146     print ("lwarpmk: \"" .. sourcename .. "-images.txt\" does not exist.")
4147     print ("lwarpmk: Your project does not use SVG math or other lateximages,")
4148     print ("lwarpmk: or the file has been deleted somehow.")
4149     print ("lwarpmk: Use \"lwarpmk html\" to recompile your project")
4150     print ("lwarpmk: and recreate \"" .. sourcename .. "-images.txt\".")
4151    print ("lwarpmk: If your project does not use SVG math or other lateximages,")
4152    print ("lwarpmk: then \"" .. sourcename .. "-images.txt\" will never exist, and")
4153     print ("lwarpmk: \"lwarpmk limages\" will not be necessary.")
4154     print ("lwarpmk: ===")
4155 end -- warnlimages
4156
4157
4158 function checklimages ()
4159 --
4160 -- Check <sourcename>.txt to see if need to recompile first.
4161 -- If any entry has a page number of zero, then there were incorrect images.
4162 --
4163 print ("lwarpmk: Checking for a valid " .. sourcename .. "-images.txt file.")
4164 local limagesfile = io.open(sourcename .. "-images.txt", "r")
4165 if ( limagesfile == nil ) then
4166     warnlimages ()
4167     os.exit(1)
4168 end
4169 -- Track warning to recompile if find a page 0
4170 local pagezerowarning = false
4171 -- Scan <sourcename>.txt
4172 for line in limagesfile:lines() do
4173     -- lwimgpage is the page number in the PDF which has the image
4174     -- lwimghash is true if this filename is a hash
4175     -- lwimgname is the lateximage filename root to assign for the image
4176     i,j,lwimgpage,lwimghash,lwimgname = string.find (line,"|(.*)|(.*)|(.*)|")
4177     -- For each entry:
4178     if ( (i~=nil) ) then
4179         -- If the page number is 0, image references are incorrect
4180         --  and must recompile the soure document:
4181         if ( lwimgpage == "0" ) then
4182             pagezerowarning = true
4183         end
4184     end -- if i~=nil
4185 end -- do
4186 if ( pagezerowarning ) then
```

```
4187     print ("")
4188     print ("lwarpmk: ===")
4189    print ("lwarpmk: The document must be recompiled before creating the lateximages.")
4190    print ("lwarpmk: Enter \"lwarpmk html\" again, then try \"lwarpmk limages\" again.")
4191     print ("lwarpmk: ===")
4192     os.exit(1) ;
4193 end -- pagezerowarning
4194 end -- checklimages
4195
4196
4197 function createuniximage ( lwimgfullname )
4198 --
4199 -- Create one lateximage for Unix / Linux / Mac OS.
4200 --
4201 executecheckerror (
4202     cmdgroupopenname ..
4203     "pdfseparate -f " .. lwimgpage .. " -l " .. lwimgpage .. " " ..
4204         sourcename .."_html.pdf " ..
4205         imagesdirectory .. dirslash .."lateximagetemp-%d" .. ".pdf" ..
4206         seqname ..
4207     -- Crop the image:
4208     "pdfcrop  --hires  " .. imagesdirectory .. dirslash .. "lateximagetemp-" ..
4209         lwimgpage .. ".pdf " ..
4210         imagesdirectory .. dirslash .. lwimgname .. ".pdf" ..
4211         seqname ..
4212     -- Convert the image to svg:
4213    "pdftocairo -svg  -noshrink  " .. imagesdirectory .. dirslash .. lwimgname .. ".pdf " ..
4214         imagesdirectory .. dirslash .. lwimgname ..".svg" ..
4215         seqname ..
4216     -- Remove the temporary files:
4217    rmname .. " " .. imagesdirectory .. dirslash .. lwimgname .. ".pdf" .. seqname ..
4218    rmname .. " " .. imagesdirectory .. dirslash .. "lateximagetemp-" .. lwimgpage .. ".pdf" ..
4219     cmdgroupclosename .. " >/dev/null " .. bgname
4220     ,
4221     "File error trying to convert " .. lwimgfullname
4222 )
4223 -- Every 32 images, wait for completion at below normal priority,
4224 --  allowing other image tasks to catch up.
4225 numimageprocesses = numimageprocesses + 1
4226 if ( numimageprocesses > 32 ) then
4227     numimageprocesses = 0
4228     print ( "lwarpmk: waiting" )
4229     executecheckerror ( "wait" , "File error trying to wait.")
4230 end
4231 end -- createuniximage
4232
4233
4234 function createwindowsimage ( lwimgfullname )
4235 --
4236 -- Create one lateximage for Windows.
4237 --
4238 -- Every 32 images, wait for completion at below normal priority,
4239 --  allowing other image tasks to catch up.
4240 numimageprocesses = numimageprocesses + 1
4241 if ( numimageprocesses > 32 ) then
```

```
    numimageprocesses = 0
    thiswaitcommand = "/WAIT /BELOWNORMAL"
    print ( "lwarpmk: waiting" )
else
    thiswaitcommand = ""
end
-- Execute the image generation command
executecheckerror (
    "start /B " .. thiswaitcommand .. " \"\" lwarp_one_limage " ..
    lwimgpage .. " " ..
    lwimghash .. " " ..
    lwimgname .. " " ..
    sourcename .. " <nul >nul"
    ,
    "File error trying to create image."
)
end -- createwindowsimage


function createonelateximage ( line )
--
-- Given the next line of <sourcename>.txt, convert a single image.
--
-- lwimgpage is the page number in the PDF which has the image
-- lwimghash is true if this filename is a hash
-- lwimgname is the lateximage filename root to assign for the image
i,j,lwimgpage,lwimghash,lwimgname = string.find (line,"|(.*)|(.*)|(.*)|")
-- For each entry:
if ( (i~=nil) ) then
    -- Skip if the page number is 0:
    if ( lwimgpage == "0" ) then
        pagezerowarning = true
    else
        -- Skip is this image is hashed and already exists:
       local lwimgfullname = imagesdirectory .. dirslash .. lwimgname .. ".svg"
        if (
            (lwimghash ~= "true") or
            (lfs.attributes(lwimgfullname,"mode")==nil) -- file not exists
        )
        then -- not hashed or not exists:
            -- Print the name of the file being generated:
            print ( "lwarpmk: " .. lwimgname )
          -- Touch/create the dest so that only once instance tries to build it:
            executecheckerror (
                newtouchname .. " " .. lwimgfullname ,
                "File error trying to touch " .. lwimgfullname
            )
            -- Separate out the image into its own single-page pdf:
            if opsystem=="Unix" then
                createuniximage (lwimgfullname)
            elseif opsystem=="Windows" then
                createwindowsimage (lwimgfullname)
            end
        end -- not hashed or not exists
    end -- not page 0
```

```
4297 end -- not nil
4298 end -- createonelateximage
4299
4300
4301 function createlateximages ()
4302 --
4303 -- Create lateximages based on <sourcename>-images.txt:
4304 --
4305 -- See if the document must be recompiled first:
4306 checklimages ()
4307 -- See if the print version exists:
4308 checkhtmlpdfexists ()
4309 -- Attempt to create the lateximages:
4310 print ("lwarpmk: Creating lateximages.")
4311 local limagesfile = io.open(sourcename .. "-images.txt", "r")
4312 if ( limagesfile == nil ) then
4313     warnlimages ()
4314     os.exit(1)
4315 end
4316 -- Create the lateximages directory, ignore error if already exists
4317 err = os.execute("mkdir " .. imagesdirectory)
4318 -- For Windows, create lwarp_one_limage.cmd from lwarp_one_limage.txt:
4319 if opsystem=="Windows" then
4320     executecheckerror (
4321         cpname .. " lwarp_one_limage.txt lwarp_one_limage.cmd" ,
4322        "File error trying to copy lwarp_one_limage.txt to lwarp_one_limage.cmd"
4323     )
4324 end -- create lwarp_one_limage.cmd
4325 -- Track the number of parallel processes
4326 numimageprocesses = 0
4327 -- Track warning to recompile if find a page 0
4328 pagezerowarning = false
4329 -- Scan <sourcename>.txt
4330 for line in limagesfile:lines() do
4331     createonelateximage ( line )
4332 end -- do
4333 io.close(limagesfile)
4334 print ( "lwarpmk limages: ===")
4335 print ( "lwarpmk limages: Wait a moment for the images to complete" )
4336 print ( "lwarpmk limages:   before reloading the page." )
4337 print ( "lwarpmk limages: ===")
4338 print ( "lwarpmk limages: Done." )
4339 if ( pagezerowarning == true ) then
4340     print ( "lwarpmk limages: WARNING: Images will be incorrect." )
4341     print ( "lwarpmk limages:   Enter \"lwarpmk cleanlimages\", then" )
4342     print ( "lwarpmk limages:   recompile the document one more time, then" )
4343     print ( "lwarpmk limages:   repeat \"lwarpmk images\" again." )
4344 end -- pagezerowarning
4345 end -- function
4346
4347
4348 function convertepstopdf ()
4349 --
4350 -- Converts EPS files to PDF files.
4351 -- The filenames are arg[argindex] and up.
```

```
-- arg[1] is the command "pdftosvg".
--
ignoreconf ()
for i = argindex , #arg do
    if (lfs.attributes(arg[i],"mode")==nil) then
        print ("lwarpmk: File \"" .. arg[i] .. "\" does not exist.")
    else
        print ("lwarpmk: Converting \"" .. arg[i] .. "\"")
        os.execute ( "epstopdf " .. arg[i] )
    end -- if
end -- do
end --function


function convertpdftosvg ()
--
-- Converts PDF files to SVG files.
-- The filenames are arg[argindex] and up.
-- arg[1] is the command "pdftosvg".
--
ignoreconf ()
for i = argindex , #arg do
    if (lfs.attributes(arg[i],"mode")==nil) then
        print ("lwarpmk: File \"" .. arg[i] .. "\" does not exist.")
    else
        print ("lwarpmk: Converting \"" .. arg[i] .. "\"")
        os.execute ( "pdftocairo -svg " .. arg[i] )
    end -- if
end -- do
end --function


-- Force an update and conclude processing:
function updateanddone ()
print ("lwarpmk: Forcing an update of " .. sourcename ..".tex.")
refreshdate ()
print ("lwarpmk: " .. sourcename ..".tex is ready to be recompiled.")
print ("lwarpmk: Done.")
end -- function


-- Start of the main code: --


-- lwarpmk --version :

if (arg[1] == "--version") then
print ( "lwarpmk: " .. printversion )

else -- not --version


-- print intro:

print ("lwarpmk: " .. printversion .. "  Automated make for the LaTeX lwarp package.")
```

```
4407
4408
4409 -- lwarpmk print:
4410
4411 if arg[1] == "print" then
4412 loadconf ()
4413 if ( latexmk == "true" ) then
4414     print ("lwarpmk: Compiling with: " .. printlatexcmd)
4415     executecheckerror (
4416         printlatexcmd ,
4417         "Compile error."
4418     )
4419     print ("lwarpmk: Done.")
4420 else -- not latexmk
4421     verifyfileexists (sourcename .. ".tex") ;
4422     -- See if up to date:
4423     if (
4424         ( lfs.attributes ( sourcename .. ".pdf" , "modification" ) == nil ) or
4425         (
4426             lfs.attributes ( sourcename .. ".tex" , "modification" ) >
4427             lfs.attributes ( sourcename .. ".pdf" , "modification" )
4428         )
4429     ) then
4430         -- Recompile if not yet up to date:
4431         manytimes(printlatexcmd, "")
4432         print ("lwarpmk: Done.") ;
4433     else
4434         print ("lwarpmk: " .. sourcename .. ".pdf is up to date.") ;
4435     end
4436 end -- not latexmk
4437
4438
4439 -- lwarpmk print1:
4440
4441 elseif arg[1] == "print1" then
4442     loadconf ()
4443     verifyfileexists (sourcename .. ".tex") ;
4444     onetime(printlatexcmd, "")
4445     print ("lwarpmk: Done.") ;
4446
4447
4448 -- lwarpmk printindex:
4449 -- Compile the index then touch the source
4450 -- to trigger a recompile of the document:
4451
4452 elseif arg[1] == "printindex" then
4453 loadconf ()
4454 os.execute ( printindexcmd )
4455 print ("lwarpmk: -------")
4456 updateanddone ()
4457
4458
4459 -- lwarpmk printglossary:
4460 -- Compile the glossary then touch the source
4461 -- to trigger a recompile of the document:
```

```
4462
4463 elseif arg[1] == "printglossary" then
4464 loadconf ()
4465 print ("lwarpmk: Processing the glossary.")
4466
4467 os.execute(glossarycmd .. " " .. sourcename)
4468 updateanddone ()
4469
4470
4471 -- lwarpmk html:
4472
4473 elseif arg[1] == "html" then
4474 loadconf ()
4475 if ( latexmk == "true" ) then
4476     print ("lwarpmk: Compiling with: " .. HTMLlatexcmd)
4477     executecheckerror (
4478         HTMLlatexcmd ,
4479         "Compile error."
4480     )
4481     pdftohtml ()
4482     print ("lwarpmk: Done.")
4483 else -- not latexmk
4484     verifyfileexists ( sourcename .. ".tex" ) ;
4485     -- See if exists and is up to date:
4486     if (
4487       ( lfs.attributes ( homehtmlfilename .. ".html" , "modification" ) == nil ) or
4488         (
4489             lfs.attributes ( sourcename .. ".tex" , "modification" ) >
4490             lfs.attributes ( homehtmlfilename .. ".html" , "modification" )
4491         )
4492     ) then
4493         -- Recompile if not yet up to date:
4494         manytimes(HTMLlatexcmd, "_html")
4495         pdftohtml ()
4496         print ("lwarpmk: Done.")
4497     else
4498         print ("lwarpmk: " .. homehtmlfilename .. ".html is up to date.")
4499     end
4500 end -- not latexmk
4501
4502
4503 -- lwarpmk html1:
4504
4505 elseif arg[1] == "html1" then
4506     loadconf ()
4507     verifyfileexists ( sourcename .. ".tex" ) ;
4508     onetime(HTMLlatexcmd, "_html")
4509     pdftohtml ()
4510     print ("lwarpmk: Done.")
4511
4512
4513 -- lwarpmk pdftohtml:
4514 elseif arg[1] == "pdftohtml" then
4515     loadconf ()
4516     pdftohtml ()
```

```
4517
4518
4519 -- lwarpmk htmlindex:
4520 -- Compile the index then touch the source
4521 -- to trigger a recompile of the document:
4522
4523 elseif arg[1] == "htmlindex" then
4524 loadconf ()
4525 os.execute ( HTMLindexcmd )
4526 print ("lwarpmk: -------")
4527 updateanddone ()
4528
4529
4530 -- lwarpmk htmlglossary:
4531 -- Compile the glossary then touch the source
4532 -- to trigger a recompile of the document.
4533 -- The <sourcename>.xdy file is created by the glossaries package.
4534
4535 elseif arg[1] == "htmlglossary" then
4536 loadconf ()
4537 print ("lwarpmk: Processing the glossary.")
4538 os.execute(glossarycmd .. " " .. sourcename .. "_html")
4539 updateanddone ()
4540
4541
4542 -- lwarpmk limages:
4543 -- Scan the <sourcename>.txt file to create lateximages.
4544
4545 elseif arg[1] == "limages" then
4546 loadconf ()
4547 print ("lwarpmk: Processing images.")
4548 createlateximages ()
4549 print ("lwarpmk: Done.")
4550
4551
4552 -- lwarpmk again:
4553 -- Touch the source to trigger a recompile.
4554
4555 elseif arg[1] == "again" then
4556 loadconf ()
4557 updateanddone ()
4558
4559
4560 -- lwarpmk clean:
4561 -- Remove project.aux, .toc, .lof, .lot, .log, *.idx, *.ind, *_html_inc.*, .gl*
4562
4563 elseif arg[1] == "clean" then
4564 loadconf ()
4565 removeaux ()
4566 print ("lwarpmk: Done.")
4567
4568
4569 -- lwarpmk cleanall
4570 -- Remove project.aux, .toc, .lof, .lot, .log, *.idx, *.ind, *_html_inc.*, .gl*
4571 --    and also project.pdf, project.dvi, *.html
```

```
4572
4573 elseif arg[1] == "cleanall" then
4574 loadconf ()
4575 removeaux ()
4576 os.execute ( rmname .. " " ..
4577     sourcename .. ".pdf " .. sourcename .. "_html.pdf " ..
4578     sourcename .. ".dvi " .. sourcename .. "_html.dvi " ..
4579     "*.html"
4580     )
4581 print ("lwarpmk: Done.")
4582
4583
4584 -- lwarpmk cleanlimages
4585 -- Remove images from the imagesdirectory.
4586
4587 elseif arg[1] == "cleanlimages" then
4588 loadconf ()
4589 os.execute ( rmname .. " " .. imagesdirectory .. dirslash .. "*" )
4590 print ("lwarpmk: Done.")
4591
4592 -- lwarpmk epstopdf <list of file names>
4593 -- Convert EPS files to PDF using epstopdf
4594 elseif arg[1] == "epstopdf" then
4595 convertepstopdf ()
4596 print ("lwarpmk: Done.")
4597
4598
4599 -- lwarpmk pdftosvg <list of file names>
4600 -- Convert PDF files to SVG using pdftocairo
4601 elseif arg[1] == "pdftosvg" then
4602 convertpdftosvg ()
4603 print ("lwarpmk: Done.")
4604
4605
4606 -- lwarpmk with no argument :
4607
4608 elseif (arg[1] == nil) then
4609 printhelp ()
4610
4611
4612 -- lwarpmk -h or lwarpmk --help :
4613
4614 elseif (arg[1] == "-h" ) or (arg[1] == "--help") then
4615 printusage ()
4616
4617
4618 -- Unknown command:
4619
4620 else
4621 printhelp ()
4622 print ("\nlwarpmk: ****** Unknown command \""..arg[1].."\". ******\n")
4623 end
4624
4625 end -- not --version
4626 \end{filecontents*}
```

```
% \end{Verbatim}% for syntax highlighting
\end{LWR@createlwarpmk}
```

41 Stacks

for HTML output:

```
\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:

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

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

```
\newcommand*{\LWR@depthfinished}{-4}
\newcommand*{\LWR@depthpart}{-1}
\newcommand*{\LWR@depthchapter}{0}
\newcommand*{\LWR@depthsection}{1}
\newcommand*{\LWR@depthsubsection}{2}
\newcommand*{\LWR@depthsubsubsection}{3}
\newcommand*{\LWR@depthparagraph}{4}
\newcommand*{\LWR@depthsubparagraph}{5}
```

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

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

used by `\item`:

```
\newcommand*{\LWR@depthlistitem}{7}
```

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.

```
\newcommand*{\LWR@closeone}{}% top of the stack
\newcommand*{\LWR@closetwo}{}
\newcommand*{\LWR@closethree}{}
\newcommand*{\LWR@closefour}{}
\newcommand*{\LWR@closefive}{}
\newcommand*{\LWR@closesix}{}
\newcommand*{\LWR@closeseven}{}
\newcommand*{\LWR@closeeight}{}
\newcommand*{\LWR@closenine}{}
\newcommand*{\LWR@closeten}{}
\newcommand*{\LWR@closeeleven}{}
\newcommand*{\LWR@closetwelve}{}
```

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}
```

```
\newcommand*{\LWR@closedepthone}{\LWR@depthnone}% top of the stack
\newcommand*{\LWR@closedepthtwo}{\LWR@depthnone}
\newcommand*{\LWR@closedepththree}{\LWR@depthnone}
\newcommand*{\LWR@closedepthfour}{\LWR@depthnone}
\newcommand*{\LWR@closedepthfive}{\LWR@depthnone}
\newcommand*{\LWR@closedepthsix}{\LWR@depthnone}
\newcommand*{\LWR@closedepthseven}{\LWR@depthnone}
\newcommand*{\LWR@closedeptheight}{\LWR@depthnone}
\newcommand*{\LWR@closedepthnine}{\LWR@depthnone}
\newcommand*{\LWR@closedepthten}{\LWR@depthnone}
\newcommand*{\LWR@closedeptheleven}{\LWR@depthnone}
\newcommand*{\LWR@closedepthtwelve}{\LWR@depthnone}
```

41.4 Pushing and popping the stack

\pushclose {⟨*action*⟩} {⟨*depth*⟩}

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

```
\NewDocumentCommand{\pushclose}{m m}
{
\global\let\LWR@closetwelve\LWR@closeeleven
\global\let\LWR@closeeleven\LWR@closeten
\global\let\LWR@closeten\LWR@closenine
\global\let\LWR@closenine\LWR@closeeight
\global\let\LWR@closeeight\LWR@closeseven
\global\let\LWR@closeseven\LWR@closesix
\global\let\LWR@closesix\LWR@closefive
\global\let\LWR@closefive\LWR@closefour
\global\let\LWR@closefour\LWR@closethree
\global\let\LWR@closethree\LWR@closetwo
\global\let\LWR@closetwo\LWR@closeone
\global\let\LWR@closeone#1
\global\let\LWR@closedepthtwelve\LWR@closedeptheleven
\global\let\LWR@closedeptheleven\LWR@closedepthten
\global\let\LWR@closedepthten\LWR@closedepthnine
\global\let\LWR@closedepthnine\LWR@closedeptheight
\global\let\LWR@closedeptheight\LWR@closedepthseven
\global\let\LWR@closedepthseven\LWR@closedepthsix
\global\let\LWR@closedepthsix\LWR@closedepthfive
\global\let\LWR@closedepthfive\LWR@closedepthfour
\global\let\LWR@closedepthfour\LWR@closedepththree
\global\let\LWR@closedepththree\LWR@closedepthtwo
\global\let\LWR@closedepthtwo\LWR@closedepthone
\global\let\LWR@closedepthone#2
}
```

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

```
\newcommand*{\popclose}
{
\global\let\LWR@closeone\LWR@closetwo
\global\let\LWR@closetwo\LWR@closethree
\global\let\LWR@closethree\LWR@closefour
\global\let\LWR@closefour\LWR@closefive
\global\let\LWR@closefive\LWR@closesix
\global\let\LWR@closesix\LWR@closeseven
\global\let\LWR@closeseven\LWR@closeeight
\global\let\LWR@closeeight\LWR@closenine
\global\let\LWR@closenine\LWR@closeten
\global\let\LWR@closeten\LWR@closeeleven
\global\let\LWR@closeeleven\LWR@closetwelve
\global\let\LWR@closedepthone\LWR@closedepthtwo
\global\let\LWR@closedepthtwo\LWR@closedepththree
\global\let\LWR@closedepththree\LWR@closedepthfour
\global\let\LWR@closedepthfour\LWR@closedepthfive
\global\let\LWR@closedepthfive\LWR@closedepthsix
```

```
\global\let\LWR@closedepthsix\LWR@closedepthseven
\global\let\LWR@closedepthseven\LWR@closedeptheight
\global\let\LWR@closedeptheight\LWR@closedepthnine
\global\let\LWR@closedepthnine\LWR@closedepthten
\global\let\LWR@closedepthten\LWR@closedeptheleven
\global\let\LWR@closedeptheleven\LWR@closedepthtwelve
}

\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}{}
```

for HTML output:

```
\begin{warpHTML}
```

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

```
\NewDocumentCommand{\LWR@setexparray}{m m m}{%
    \begingroup%
    \let\par\relax%
    \xdef\LWR@thisexparrayname{#1#2}%
    \ifstrempty{#3}%
    {\csgdef{\LWR@thisexparrayname}{}}%
    {\csxdef{\LWR@thisexparrayname}{#3}}%
    \endgroup%
}
```

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

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

\end{warpHTML}
```

43 Localizing catcodes

for HTML & PRINT:

```
\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.

```
\newcommand{\StartDefiningTabulars}{%
\LWR@traceinfo{StartDefiningTabulars}%
\warpHTMLonly{\catcode`\&=\active}%
}
```

`\StopDefiningTabulars` Place after defining something with & in it.

```
\newcommand{\StopDefiningTabulars}{%
\LWR@traceinfo{StopDefiningTabulars}%
\warpHTMLonly{\catcode`\&=4}%
}
```

Bool `LWR@mathmacro` 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.

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

`\StartDefiningMath` Place before defining something with $ in it.

```
\newcommand{\StartDefiningMath}{%
\LWR@traceinfo{StartDefiningMath}%
\warpHTMLonly{\catcode`\$=\active}%
}
```

`\StopDefiningMath` Place after defining something with $ in it.

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

\end{warpall}
```

44 Localizing dynamic math

Inline SVG math usually uses a hash of its contents to generate `latexi`mages which are reusable for multiple instances with the same contents. If the contents may change for

each use, such as depending on the current value of a counter, then `\inlinemathother` must be used before the inline math expression, and `\inlinemathnormal` must be used after.

For MathJax, the inline math expression is usually printed for MathJax to interpret. When marked as dynamic math, the following inline math expression will be displayed as an unhashed inline SVG image instead.

For existing code and packages, it may be possible to patch macros after they have been defined, using the xpatch package, which is pre-loaded by lwarp:

```
\xpatchcmd{\macroname}
    {$math expression$}
    {\inlinemathother$math expression$\inlinemathnormal}
    {}
    {\typeout{Error patching macroname.}}
```

for HTML & PRINT:

```
\begin{warpall}
```

Bool `LWR@dynamicmath` Default: `false`

True to mark inline math which is dynamic in nature, thus should not be hashed for reuse.

```
\newbool{LWR@dynamicmath}
\boolfalse{LWR@dynamicmath}
```

`\inlinemathother`

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

```
\newcommand{\inlinemathother}{%
\LWR@traceinfo{inlinemathother}%
\booltrue{LWR@dynamicmath}%
}
```

`\inlinemathnormal`

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

```
\newcommand{\inlinemathnormal}{%
\LWR@traceinfo{inlinemathnormal}%
\boolfalse{LWR@dynamicmath}%
}
```

```
\end{warpall}
```

45 HTML entities

for HTML output:

```
\begin{warpHTML}
```

HTML Unicode entities:

```
\let\LWR@origampersand\&
```

`\HTMLentity` {⟨*entitytag*⟩}

```
\newcommand*{\HTMLentity}[1]{%
% \LWR@traceinfo{HTMLentity \detokenize{#1}}%
\begingroup%
\LWR@FBcancel%
\LWR@origampersand#1;%
\endgroup%
% \LWR@traceinfo{HTMLentity done}%
}
```

`\HTMLunicode` {⟨*hex_unicode*⟩}

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

`\&`

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

`\textless`

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

`\textgreater`

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

```
\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:

```
\begin{warpall}
```

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

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

\HTMLFilename The prefix for all generated HTML files other than the home page, defaulting to empty. See section 8.4.1.

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

\HomeHTMLFilename The filename of the home page, defaulting to the \BaseJobname. See section 8.4.1.

```
\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.

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

Bool FileSectionNames Selects how to create HTML file names.

Defaults to use section names in the filenames.

```
\newbool{FileSectionNames}
\booltrue{FileSectionNames}
```

```
\end{warpall}
```

for HTML output:

```
\begin{warpHTML}
```

Ctr LWR@htmlfilenumber Records the number of each HTML file as it is being created. Number 0 is the home page.

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

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

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

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

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

```
% \LWR@traceinfo{about to assign temp}%
\LWR@sanitize{#1}%
\LWR@traceinfo{about to compare with ??}%
\ifthenelse{\equal{\LWR@sanitized}{??}}%
```

```
    {\LWR@traceinfo{found ??}}%
    {\LWR@traceinfo{not found ??}}%
\LWR@traceinfo{about to compare with zero or empty}%
\ifthenelse{%
    \equal{\LWR@sanitized}{0}%
    \OR \equal{\LWR@sanitized}{}%
    \OR \equal{\LWR@sanitized}{??}%
}%
{%
    \LWR@traceinfo{LWR@htmlsectionfilename B \HomeHTMLFilename.html}%
    \HomeHTMLFilename.html%
}%
```

For a LaTeX section named “Index” or “index” without a prefix, create a filename with a leading underscore to avoid colliding with the HTML filename `index.html`:

```
{%
    \LWR@traceinfo{LWR@htmlsectionfilename C \LWR@sanitized}%
    \ifthenelse{%
        \equal{\HTMLFilename}{} \AND
        \equal{\LWR@sanitized}{Index} \OR
        \equal{\LWR@sanitized}{index}%
    }%
    {%
        \LWR@traceinfo{Prefixing the index name with an underscore.}%
        \_\LWR@sanitized.html%
    }%
```

Otherwise, create a filename with the chosen prefix:

```
    {%
        \HTMLFilename\LWR@isolate{\LWR@sanitized}.html%
    }%
}%
\LWR@traceinfo{LWR@htmlsectionfilename Z}%
}
```

`\LWR@htmlrefsectionfilename` {⟨*label*⟩}

Prints the filename for the given label

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

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

```
\begingroup%
\LWR@nullfonts%
\LWR@htmlsectionfilename{\LWR@htmlfileref{#1}}%
\endgroup%
\LWR@traceinfo{LWR@htmlrefsectionfilename: done}%
}

\end{warpHTML}
```

47 Homepage link

for HTML & PRINT:

```
\begin{warpall}
```

`\linkhomename` Holds the default name for the home link.

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

```
\end{warpall}
```

for HTML output:

```
\begin{warpHTML}
```

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

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

```
\end{warpHTML}
```

for PRINT output:

```
\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.

```
\AtBeginDocument{
\@ifundefined{hyperref}{
    \newcommand*{\LinkHome}{%
        \linkhomename\ --- page \pageref{\BaseJobname-page-LWRfirstpage}%
    }
}{
    \newcommand*{\LinkHome}{%
        \hyperref[\BaseJobname-page-LWRfirstpage]{\linkhomename}%
    }
}
}

\AfterEndPreamble{\label{\BaseJobname-page-LWRfirstpage}}
```

```
\end{warpprint}
```

for HTML output:

```
\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.

```
\newcommand*{\LWR@topnavigation}{
\LWR@htmlelementclassline{nav}{topnavigation}{\LinkHome}
}
```

\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.

```
\newcommand*{\LWR@botnavigation}{
\LWR@htmlelementclassline{nav}{botnavigation}{\LinkHome}
}
```

```
\end{warpHTML}
```

48 \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:

```
\begin{warpHTML}
```

\LWRPrintStack Prints the closedepth stack.

```
\newcommand*{\LWR@subprintstack}{
\LWR@closedepthone\ \LWR@closedepththwo\ \LWR@closedepththree\
\LWR@closedepthfour\ \LWR@closedepthfive\ \LWR@closedepthsix\
\LWR@closedepthseven\ \LWR@closedeptheight\ \LWR@closedepthnine\
\LWR@closedepthten\ \LWR@closedeptheleven\ \LWR@closedepthtwelve\
}

\newcommand*{\LWRPrintStack}{
\LWR@startpars
\LWR@subprintstack
}
```

```
\end{warpHTML}
```

for PRINT output:

```
\begin{warpprint}

\newcommand*{\LWRPrintStack}{}

\end{warpprint}
```

49 Closing stack levels

for HTML output:

```
\begin{warpHTML}
```

Close one nested level:

```
\newcommand*{\LWR@closeoneprevious}{%

\LWR@closeone

\popclose
}
```

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

```
\newcommand*{\LWR@closeprevious}[1]{
\LWR@traceinfo{LWR@closeprevious to depth #1, depths are \LWR@subprintstack}%
```

Close any pending paragraph:

```
\LWR@stoppars%
```

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

```
\whileboolexpr{test{\ifnumcomp{\LWR@closedepthone}{>}{#1}}}%
{%
    \LWR@traceinfo{LWR@closeprevious: closing out depth \LWR@closedepthone}%
    \LWR@closeoneprevious%
}%
\ifboolexpr{test{\ifnumcomp{\LWR@closedepthone}{=}{#1}}}%
{%
    \LWR@traceinfo{LWR@closeprevious: closing out depth \LWR@closedepthone}%
    \LWR@closeoneprevious%
}{}%
\LWR@traceinfo{LWR@closeprevious: done, depths are \LWR@subprintstack}%
}

\end{warpHTML}
```

50 PDF pages and styles

for HTML output:

```
\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.

```
\newcommand{\LWR@forcenewpage}{%
\LWR@traceinfo{LWR@forcenewpage}%
\ifinner\else%
\LWR@stoppars\LWR@orignewpage\LWR@startpars%
\fi%
}
```

\pagestyle, etc. are nullified for HTML output.

\pagestyle {⟨*style*⟩}

```
4915 \renewcommand*{\pagestyle}[1]{}
```

\thispagestyle {⟨*style*⟩}

```
4916 \renewcommand*{\thispagestyle}[1]{}
```

\markboth {⟨*left*⟩} {⟨*right*⟩}

```
4917 \renewcommand*{\markboth}[2]{}
```

\markright {⟨*right*⟩}

```
4918 \renewcommand*{\markright}[1]{}
```

\raggedbottom

```
4919 \renewcommand*{\raggedbottom}{}
```

\flushbottom

```
4920 \renewcommand*{\flushbottom}{}
```

\sloppy

```
4921 \renewcommand*{\sloppy}{}
```

\fussy

```
4922 \renewcommand*{\fussy}{}
```

\pagenumbering * {⟨*commands*⟩}

```
4923 \RenewDocumentCommand{\pagenumbering}{s m}{}
4924 \end{warpHTML}
```

51 HTML tags, spans, divs, elements

for HTML output:

```
4925 \begin{warpHTML}
```

51.1 Mapping LaTeX sections to HTML sections

```
\newcommand*{\LWR@tagtitle}{h1}
\newcommand*{\LWR@tagtitleend}{/h1}
\newcommand*{\LWR@tagpart}{h2}
\newcommand*{\LWR@tagpartend}{/h2}
\newcommand*{\LWR@tagchapter}{h3}
\newcommand*{\LWR@tagchapterend}{/h3}
\newcommand*{\LWR@tagsection}{h4}
\newcommand*{\LWR@tagsectionend}{/h4}
\newcommand*{\LWR@tagsubsection}{h5}
\newcommand*{\LWR@tagsubsectionend}{/h5}
\newcommand*{\LWR@tagsubsubsection}{h6}
\newcommand*{\LWR@tagsubsubsectionend}{/h6}
\newcommand*{\LWR@tagparagraph}{span class="paragraph"}
\newcommand*{\LWR@tagparagraphend}{/span}
\newcommand*{\LWR@tagsubparagraph}{span class="subparagraph"}
\newcommand*{\LWR@tagsubparagraphend}{/span}

\newcommand*{\LWR@tagregularparagraph}{p}
```

51.2 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.)

```
\providecommand*{\LWR@FBcancel}{}

\AtBeginDocument{%
\@ifundefined{frenchbsetup}%
{}%
{%
    \frenchbsetup{FrenchFootnotes=false}%

%
    \LetLtxMacro\LWR@FBcancel\NoAutoSpacing%
    \renewrobustcmd*{\FBcolonspace}{%
        \begingroup%
        \LWR@FBcancel%
        \LWR@origampersand{}nbsp;%
        \endgroup%
    }%
    \renewrobustcmd*{\FBthinspace}{%
        \begingroup%
        \LWR@FBcancel%
        \LWR@origampersand\LWR@origpound{}x202f;% \,
        \endgroup%
    }%
    \renewrobustcmd*{\FBguillspace}{%
        \begingroup%
        \LWR@FBcancel%
        \LWR@origampersand{}nbsp;% ~,  for \og xyz \fg{}
        \endgroup%
```

```
    }%
    \DeclareDocumentCommand{\FBmedkern}{}{%
        \begingroup%
        \LWR@FBcancel%
        \LWR@origampersand\LWR@origpound{}x202f;% \,
        \endgroup%
    }%
    \DeclareDocumentCommand{\FBthickkern}{}{%
        \begingroup%
        \LWR@FBcancel%
        \LWR@origampersand{}nbsp;% ~
        \endgroup%
    }%
    \renewrobustcmd*{~}{\HTMLentity{nbsp}}% was overwritten by babel-french
    \ifFBunicode%
    \else%
        \DeclareTextSymbol{\FBtextellipsis}{LY1}{133}%
        \DeclareTextCommandDefault{\FBtextellipsis}{\textellipsis\xspace}%
    \fi%
}%
}
```

51.3 HTML output formatting

Helps format the output HTML code for human readability.

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

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

51.4 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.

```
\newcommand*{\LWR@htmltagc}[1]{%
\LWR@traceinfo{LWR@htmltagc !\detokenize{#1}!}%
\begingroup%
\LWR@FBcancel%
\ifmmode\else\protect\LWR@origttfamily\fi%
\protect\LWR@origtextless%
\LWR@isolate{#1}%
\protect\LWR@origtextgreater%
\endgroup%
}
```

Env `LWR@nestspan` Disable `minipage`, `\parbox`, and HTML `<div>`s inside a `<span>`.

⚠ `\begin{LWR@nestspan}` must follow the opening `<span>` tag to allow a paragraph to start if the span is at the beginning of a new paragraph.

⚠ `\end{LWR@nestspan}` must follow the `</span>` or a `<p>` may appear inside the span.

```
\newcommand*{\LWR@nestspanitem}{%
\if@newlist\else{\LWR@htmltagc{br /}}\fi%
\LWR@origitem%
}

\newenvironment*{LWR@nestspan}
{%
\LWR@traceinfo{LWR@nestspan starting}%
\ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
{%
    \LWR@traceinfo{LWR@nestspan: inside a lateximage}%
}%
{% not in a lateximage
    \LWR@traceinfo{LWR@nestspan: NOT inside a lateximage}%
    \addtocounter{LWR@spandepth}{1}%
    \RenewDocumentEnvironment{minipage}{O{t} o O{t} m}{}{}%
    \RenewDocumentEnvironment{BlockClass}{o m}{}{}%
    \renewcommand{\BlockClassSingle}[2]{##2}%
    \renewcommand{\LWR@forcenewpage}{}%
    \renewcommand{\LWR@liststart}{%
        \let\item\LWR@nestspanitem%
    }%
    \renewcommand{\LWR@listend}{\LWR@htmltagc{br /}\LWR@htmltagc{br /}}%
}% not in a lateximage
\LWR@traceinfo{LWR@nestspan starting: done}%
}% starting env
{% ending env
\LWR@traceinfo{LWR@nestspan ending}%
\ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
{}%
{\addtocounter{LWR@spandepth}{-1}}%
\LWR@traceinfo{LWR@nestspan ending: done}%
}

\AfterEndEnvironment{LWR@nestspan}{\global\let\par\LWR@closeparagraph}
```

`\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.

```
\NewDocumentCommand{\LWR@htmlspan}{m +m}{%
\LWR@ensuredoingapar%
\LWR@htmltagc{#1}%
\begin{LWR@nestspan}%
#2%
\LWR@htmltagc{/#1}%
```

```
\end{LWR@nestspan}%
}
```

`\LWR@htmlspanclass` `[`⟨*style*⟩`]` `{`⟨*class*⟩`}` `{`⟨*text*⟩`}`

```
\NewDocumentCommand{\LWR@htmlspanclass}{o m +m}{%
\LWR@traceinfo{LWR@htmlspanclass |#1|#2|}%
\LWR@ensuredoingapar%
\LWR@subhtmlelementclass{span}[#1]{#2}%
\begin{LWR@nestspan}%
#3%
\LWR@htmltagc{/span}%
\LWR@traceinfo{LWR@htmlspanclass done}%
\end{LWR@nestspan}%
}
```

`\LWR@htmltag` `{`⟨*tag*⟩`}`

Print an HTML tag: <tag>

```
\newcommand*{\LWR@htmltag}[1]{%
% \LWR@traceinfo{LWR@htmltagb !\detokenize{#1}!}%
\LWR@htmltagc{#1}%
% \LWR@traceinfo{LWR@htmltagb: done}%
}
```

51.5 Block tags and comments

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

`\LWR@htmlopencomment`
`\LWR@htmlclosecomment`

```
\newcommand*{\LWR@htmlopencomment}{%
{%
% \LWR@traceinfo{LWR@htmlopencomment}%
\begingroup%
\LWR@FBcancel%
\ifmmode\else\protect\LWR@origttfamily\fi%
\LWR@print@mbox{\LWR@origtextless{}!-\/-}%
\endgroup%
}%
}

\newcommand*{\LWR@htmlclosecomment}{%
{%
% \LWR@traceinfo{LWR@htmlclosecomment}%
\begingroup%
\LWR@FBcancel%
\ifmmode\else\protect\LWR@origttfamily\fi%
```

```
\LWR@print@mbox{-\/-\LWR@origtextgreater}%
\endgroup%
}%
}
```

`\LWR@htmlcomment` {⟨*comment*⟩}

```
\newcommand{\LWR@htmlcomment}[1]{%
\LWR@htmlopencomment{}%
{%
\LWR@origttfamily% break ligatures
#1%
}%
\LWR@htmlclosecomment{}}
```

`\LWR@htmlblockcomment` {⟨*comment*⟩}

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

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

```
\newcommand*{\LWR@htmlblocktag}[1]{%
\LWR@stoppars%
\LWR@htmltag{#1}%
\LWR@startpars%
}
```

51.6 Div class and element class

`\LWR@subhtmlelementclass` {⟨*element*⟩} [⟨*style*⟩] {⟨*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.

```
\NewDocumentCommand{\LWR@subhtmlelementclass}{m O{} m}{%
\LWR@traceinfo{LWR@subhtmlelementclass !#1!#2!#3!}%
\ifblank{#2}%
{% empty option
    \LWR@htmltag{%
        #1 % space
        class=\textquotedbl#3\textquotedbl\ % space
    }%
}%
{% non-empty option
    \LWR@htmltag{%
        #1\LWR@indentHTML%
```

```
        class=\textquotedbl#3\textquotedbl\LWR@indentHTML%
        style=\textquotedbl#2\textquotedbl\LWR@orignewline%
    }%
}%
\LWR@traceinfo{LWR@subhtmlelementclass done}%
}
```

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

```
\NewDocumentCommand{\LWR@htmlelementclass}{m o m}{%
\LWR@stoppars%
\LWR@subhtmlelementclass{#1}[#2]{#3}%
\LWR@startpars%
}
```

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

```
\newcommand*{\LWR@htmlelementclassend}[2]{%
\LWR@stoppars%
\LWR@htmltag{/#1}%
\ifbool{HTMLDebugComments}{%
    \LWR@htmlcomment{End of #1 ``#2''}%
}{}%
\LWR@startpars%
}
```

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

```
\NewDocumentCommand{\LWR@htmldivclass}{o m}{%
\LWR@htmlelementclass{div}[#1]{#2}%
}
```

\LWR@htmldivclassend {⟨*class*⟩}

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

51.7 Single-line elements

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

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

```
\NewDocumentCommand{\LWR@htmlelementclassline}{m o m +m}{%
\LWR@stoppars
\LWR@subhtmlelementclass{#1}[#2]{#3}%
#4%
```

```
\LWR@htmltag{/#1}
\LWR@startpars
}
```

51.8 HTML5 semantic elements

`\LWR@htmlelement` {⟨*element*⟩}

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

`\LWR@htmlelementend` {⟨*element*⟩}

```
\newcommand*{\LWR@htmlelementend}[1]{%
\LWR@stoppars
\LWR@htmltag{/#1}
\LWR@startpars
}

\end{warpHTML}
```

51.9 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 89.

Env `BlockClass` [⟨*style*⟩] {⟨*class*⟩} High-level interface for `<div>` classes.

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

for PRINT output:

```
\begin{warpprint}
\NewDocumentEnvironment{BlockClass}{o m}{}{}%
\end{warpprint}
```

for HTML output:

```
\begin{warpHTML}
\NewDocumentEnvironment{LWR@print@BlockClass}{o m}{}{}%
\NewDocumentEnvironment{LWR@HTML@BlockClass}{o m}%
{%
    \LWR@origpar%
    \LWR@htmldivclass[#1]{#2}%
}
{\LWR@htmldivclassend{#2}}

```

```
\LWR@formattedenv{BlockClass}
\end{warpHTML}
```

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

for HTML & PRINT:

```
\begin{warpall}
\newcommand{\BlockClassSingle}[2]{#2}
\end{warpall}
```

for HTML output:

```
\begin{warpHTML}
\newcommand{\LWR@HTML@BlockClassSingle}[2]{%
\LWR@origpar%
\LWR@htmlelementclassline{div}{#1}{#2}%
}

\LWR@formatted{BlockClassSingle}
\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 PRINT output:

```
\begin{warpprint}
\NewDocumentCommand{\InlineClass}{D{(}{)}{} o m +m}{#4}%
\end{warpprint}
```

for HTML output:

```
\begin{warpHTML}
\NewDocumentCommand{\LWR@print@InlineClass}{D{(}{)}{} o m +m}{#4}%

\NewDocumentCommand{\LWR@HTML@InlineClass}{D{(}{)}{} o m +m}{%
\ifbool{FormatWP}{%
    \LWR@htmlspanclass[#1]{#3}{#4}%
}{%
    \LWR@htmlspanclass[#2]{#3}{#4}%
}%
}

\LWR@formatted{InlineClass}
\end{warpHTML}
```

Env `LWR@BlockClassWP` {⟨*WPstyle*⟩} {⟨*HTMLstyle*⟩} {⟨*class*⟩} Low-level interface for 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 PRINT output:

```
\begin{warpprint}
\NewDocumentEnvironment{LWR@BlockClassWP}{m m m}{}{}%
\end{warpprint}
```

for HTML output:

```
\begin{warpHTML}
\NewDocumentEnvironment{LWR@print@LWR@BlockClassWP}{m m m}{}{}%
\NewDocumentEnvironment{LWR@HTML@LWR@BlockClassWP}{m m m}%
{%
\LWR@stoppars%
\ifbool{FormatWP}%
{%
    \addtocounter{LWR@thisautoidWP}{1}%

    \LWR@htmltag{%
        div class=\textquotedbl#3\textquotedbl\ % space
        id=\textquotedbl%
            \LWR@print@mbox{autoidWP-\arabic{LWR@thisautoidWP}}%
        \textquotedbl%
        \ifblank{#1}{}{ style=\textquotedbl#1\textquotedbl}%
    }%
}% FormatWP
{% not FormatWP
    \LWR@htmltag{%
        div class=\textquotedbl#3\textquotedbl%
        \ifblank{#2}{}{ style=\textquotedbl#2\textquotedbl}%
    }%
}% not FormatWP
\LWR@startpars%
}
{\LWR@htmldivclassend{#3}}

\LWR@formattedenv{LWR@BlockClassWP}
\end{warpHTML}
```

51.10 Closing HTML tags

for HTML output:

```
\begin{warpHTML}
```

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

```
\newcommand*{\LWR@printclosepart}
    {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing part}}{}}
\newcommand*{\LWR@printclosechapter}
    {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing chapter}}{}}
\newcommand*{\LWR@printclosesection}
    {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing section}}{}}
\newcommand*{\LWR@printclosesubsection}
    {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing subsection}}{}}
\newcommand*{\LWR@printclosesubsubsection}
    {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing subsubsection}}{}}
\newcommand*{\LWR@printcloseparagraph}
    {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing paragraph}}{}}
\newcommand*{\LWR@printclosesubparagraph}
    {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing subparagraph}}{}}
```

Lists require closing HTML tags:

```
\newcommand*{\LWR@printcloselistitem}
    {\LWR@htmltag{/li}}
\newcommand*{\LWR@printclosedescitem}
    {\LWR@htmltag{/dd}}
\newcommand*{\LWR@printcloseitemize}
    {\LWR@htmltag{/ul}}
\newcommand*{\LWR@printcloseenumerate}
    {\LWR@htmltag{/ol}}
\newcommand*{\LWR@printclosedescription}
    {\LWR@htmltag{/dl}}

\end{warpHTML}
```

52 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 53 has high-level commands which allow paragraph-tag generation to start/stop. Even when allowed (`\LWR@doingstartpars`), tags are not generated until a LaTeX paragraph is being used (`\LWR@doingapar`). `LWR@lateximagedepth` is used to prevent nesting tags inside a `lateximage`. `LWR@spandepth` is used to prevent nesting paragraph tags inside a paragraph, which became important inside `\fbox` commands and other spans.

for HTML output:

```
\begin{warpHTML}
```

Ctr `LWR@spandepth` Do not create paragraph tags inside of an HTML span.

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

Bool `LWR@doingstartpars` Tells whether paragraphs may be generated.

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

Bool `LWR@doingapar` Tells whether have actually generated and are currently processing paragraph text.

```
\newbool{LWR@doingapar}
\global\boolfalse{LWR@doingapar}
```

`\LWR@ensuredoingapar` If are about to print something visible, and if allowed to start a new paragraph, ensure that are `LWR@doingapar`, so that paragraph tags are placed:

```
\newcommand*{\LWR@ensuredoingapar}{%
\ifbool{LWR@doingstartpars}%
{\global\booltrue{LWR@doingapar}}%
{}%
}
```

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

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

\LWR@openparagraph

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

See if paragraph handling is enabled:

```
\ifbool{LWR@doingstartpars}%
{% handling pars
```

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

```
    \ifboolexpr{
        test {\ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}} or
        test {\ifnumcomp{\value{LWR@spandepth}}{>}{0}}
    }% nested par tags?
```

If so: Do nothing if already started a lateximage page. Cannot nest a lateximage. Also do nothing if already inside a . Do not nest paragraph tags inside a .

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

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

```
    {% yes nest par tags
```

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

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

The opening paragraph tag:

```
        \LWR@htmltagc{\LWR@tagregularparagraph}\LWR@orignewline%
```

Now have started a paragraph.

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

At the endof each paragraph, generate closing tag and do regular /par stuff. (Attempting to use the everyhook cr hook for \LWR@closeparagraph does not work well.)

```
        \let\par\LWR@closeparagraph%
    }% end of yes nest par tags
}% end of handling pars
{}% not handling pars
}
```

\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.

```
\newcommand*{\LWR@closeparagraph@br}
{%
    \ifboolexpr{
        test {\ifnumcomp{\value{LWR@spandepth}}{>}{0}} and
        test {\ifnumcomp{\value{LWR@lateximagedepth}}{=}{0}} and
        not bool {LWR@intabularmetadata}
    }%
        {\unskip\LWR@htmltagc{br /}}%
        {}%
}
```

\LWR@closeparagraph

```
\newcommand*{\LWR@closeparagraph}
{%
% \LWR@traceinfo{LWR@closeparagraph}%
```

See if paragraph handling is enabled:

```
\ifbool{LWR@doingapar}%
```

If currently in paragraph mode:

```
{% handling pars
```

See if already started a lateximage or a :

```
    \ifboolexpr{
        test {\ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}} or
        test {\ifnumcomp{\value{LWR@spandepth}}{>}{0}}
    }%
```

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

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

If have not already started a lateximage or a :

```
    {% yes nest par tags
```

Print a closing tag and some extra vertical space.

```
        \leavevmode\LWR@orignewline%
        \LWR@htmltagc{/\LWR@tagregularparagraph}%
        \LWR@orignewline%
```

No longer doing a paragraph:

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

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

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

If parnotes is used, paragraph notes are inserted after ending the previous paragraph:

```
        \PN@parnotes@auto%
    }% end of yes nest par tags
}% end of handling pars
```

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

```
{% not handling pars
    \LWR@closeparagraph@br%
}% 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:

```
\ifboolexpr{%
    not bool {LWR@doingapar} and
    test {\ifnumcomp{\value{LWR@tabulardepth}}{>}{0}} and
    test {
        \ifnumcomp{\value{LWR@tabulardepth}}{=}{\value{LWR@tabularpardepth}}
    } and
    bool {LWR@intabularmetadata} and
    not bool {LWR@tableparcell} and
    test {\ifnumcomp{\value{LWR@lateximagedepth}}{=}{0}}
}%
{%
    \LWR@getmynexttoken%
}{%
    \LWR@origpar%
}%
}

\end{warpHTML}
```

53 Paragraph start/stop handling

These commands allow/disallow the generation of HTML paragraph tags.

Section 52 has the commands which actually generate the tags.

The everyhook package is used to generate the opening paragraph tags. The closing tags are generated by \par.

for HTML output:

```
\begin{warpHTML}
```

\LWR@startpars Begin handling HTML paragraphs. This allows an HTML paragraph to start, but one has not yet begun.

```
\newcommand*{\LWR@startpars}%
{%
% \LWR@traceinfo{LWR@startpars}%
```

Ignore if inside a span:

```
\ifnumcomp{\value{LWR@spandepth}}{>}{0}%
{}%
{%
```

See if currently handling HTML paragraphs:

```
    \ifbool{LWR@doingstartpars}%
```

If already in paragraph mode, do nothing.

```
    {}%
```

If not currently in paragraph mode:

```
    {%
```

At the start of each paragraph, generate an opening tag:

```
        \PushPreHook{par}{\LWR@openparagraph}%
```

At the end of each paragraph, generate closing tag then do regular /par actions:

```
        \let\par\LWR@closeparagraph

    }% an intentionally blank line
```

Are now handling paragraphs, but have not yet actually started one:

```
    \global\setbool{LWR@doingstartpars}{true}%
```

No <par> tag yet to undo:

```
    \global\boolfalse{LWR@doingapar}%
}% nestspan
% \LWR@traceinfo{LWR@startpars: done}%
}
```

\LWR@stoppars Stop handling HTML paragraphs. Any currently open HTML paragraph is closed, and no more will be opened.

```
\newcommand*{\LWR@stoppars}%
{%
```

Ignore if inside a span:

```
\ifnumcomp{\value{LWR@spandepth}}{>}{0}%
{}%
{%
```

See if currently handling HTML paragraphs:

```
    \ifbool{LWR@doingapar}%
```

if currently in an HTML paragraph:

```
    {%
```

Print a closing tag:

```
        \leavevmode\LWR@orignewline%
        \LWR@htmltagc{/\LWR@tagregularparagraph}%
        \LWR@orignewline%
```

No longer have an open HTML paragraph:

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

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

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

    }% an intentionally blank line
```

If was not in an HTML paragraph:

```
    {}%
```

See if currently allowing HTML paragraphs:

```
    \ifbool{LWR@doingstartpars}%
```

If so: clear the `par` hook to no longer catch paragraphs:

```
        {\ClearPreHook{par}}%
```

Else: Do nothing:

```
        {}%
```

No longer in paragraph mode:

```
    \global\setbool{LWR@doingstartpars}{false}%
```

No <p> tag to undo:

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

\end{warpHTML}
```

54 Indentfirst

Pkg indentfirst indentfirst redefines `\@afterindentfalse` to be `\@afterindenttrue`. This is reversed `\AtBeginDocument` here.

for HTML output:

```
\begin{warpHTML}

\AtBeginDocument{
\def\@afterindentfalse{\let\if@afterindent\iffalse}
\@afterindentfalse
}
\let\LWR@afterindent@syntaxhighlight\fi% syntax highlighting

\end{warpHTML}
```

55 Page headers and footers

for HTML & PRINT:

```
\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.

```
\newcommand{\LWR@firstpagetop}{} % for the home page alone
\newcommand{\LWR@pagetop}{} % for all other pages
\newcommand{\LWR@pagebottom}{}
```

`\HTMLFirstPageTop` {⟨*text and logos*⟩}

```
\newcommand{\HTMLFirstPageTop}[1]{%
    \renewcommand{\LWR@firstpagetop}{#1}%
}
```

`\HTMLPageTop` {⟨*text and logos*⟩}

```
\newcommand{\HTMLPageTop}[1]{%
    \renewcommand{\LWR@pagetop}{#1}%
}
```

`\HTMLPageBottom` {⟨*text and logos*⟩}

```
\newcommand{\HTMLPageBottom}[1]{%
    \renewcommand{\LWR@pagebottom}{#1}%
}

\end{warpall}
```

56 css

for HTML output:

```
\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.

```
\newcommand*{\LWR@currentcss}{lwarp.css}
```

`\CSSFilename` {⟨*new-css-filename.css*⟩} Assigns the css file to be used by the following HTML pages.

```
\newcommand*{\CSSFilename}[1]{%
\renewcommand*{\LWR@currentcss}{#1}%
\@onelevel@sanitize\LWR@currentcss%
}

\end{warpHTML}
```

for PRINT output:

```
\begin{warpprint}
\newcommand*{\CSSFilename}[1]{}
\end{warpprint}
```

57 Title, HTML meta author, HTML meta description

for HTML output:

```
\begin{warpHTML}
```

`\title` {⟨*title*⟩} Modified to remember `\thetitle`, which is used to set the HTML page titles.

```
\let\LWR@origtitle\title

\renewcommand*{\title}[1]{%
    \LWR@origtitle{#1}%
    \begingroup%
        \renewcommand{\thanks}[1]{}%
        \protected@xdef\thetitle{#1}%
    \endgroup%
}

\end{warpHTML}
```

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

`\HTMLTitle` {⟨*Titlename*⟩} The Title to place into an HTML meta tag. The default is to use the document `\title`'s setting.

```
\providecommand{\thetitle}{}

\newcommand{\theHTMLTitle}{\thetitle}

\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.

```
\providecommand{\theauthor}{}

\newcommand{\theHTMLAuthor}{\theauthor}

\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.

```
\newcommand{\LWR@currentHTMLDescription}{}

\newcommand{\HTMLDescription}[1]{%
\renewcommand{\LWR@currentHTMLDescription}{#1}
}

\end{warpall}
```

58 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.

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

If using memoir class, with which lwarp preloads footmisc, the `stable` option must be declared before lwarp is loaded:

```
\PassOptionsToPackage{stable}{footmisc}
\usepackage{lwarp}
. . .
```

Do not use a starred sectioning command. As an alternative, it may be possible to adjust `\secnumdepth` instead.

Several kinds of footnotes are used: in a regular page, in a minipage, or as thanks in the titlepage. Each of these is handle differently.

58.1 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@footnotes`. 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 58.4 for the implementation.

58.2 Minipage footnotes

See section 58.5 for how minipage footnotes are gathered. See section 88.3 for how minipage footnotes are placed into the document.

58.3 Titlepage thanks

See section 65.7 for titlepage footnotes.

58.4 Regular page footnote implementation

for HTML & PRINT:

```
\begin{warpall}
```

Ctr `FootnoteDepth` Default: 3

Determines how deeply to place footnotes in the HTML files, similar to `tocdepth`. The default of 3 places footnotes before each `\subsubsection` or higher. See table 9 for a table of LaTeX section headings.

```
\newcounter{FootnoteDepth}
\setcounter{FootnoteDepth}{3}
```

```
\end{warpall}
```

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

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.

```
\newbox\LWR@footnotes
```

Much of the following has unneeded print-mode formatting removed.

\@makefntext {⟨*text*⟩}

```
\long\def\@makefntext#1{\textsuperscript{\@thefnmark}~#1}
```

\@makefnmark

```
\def\@makefnmark{%
    \textsuperscript{\@thefnmark}%
}
```

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*⟩}

```
\long\def\LWR@footnotetext#1{%
\LWR@traceinfo{LWR@footnotetext}%
\global\setbox\LWR@footnotes=\vbox{%
```

Add to any current footnotes:

```
    \unvbox\LWR@footnotes%
```

Remember the footnote number for \ref:

```
    \protected@edef\@currentlabel{%
        \csname p@footnote\endcsname\@thefnmark%
    }% @currentlabel
```

Open a group:

```
    \color@begingroup%
```

Use HTML superscripts in the footnote even inside a lateximage:

```
    \renewrobustcmd{\textsuperscript}[1]{\LWR@htmlspan{sup}{##1}}%
```

Use paragraph tags if in a tabular data cell or a lateximage:

```
    \ifthenelse{%
        \boolean{LWR@doingstartpars} \AND%
        \cnttest{\value{LWR@lateximagedepth}}{=}{0}%
    }%
        {}%
        {\LWR@htmltagc{\LWR@tagregularparagraph}\LWR@orignewline}%
```

Append the footnote to the list:

```
    \@makefntext{#1}%
```

Closing paragraph tag:

```
    \ifthenelse{%
        \boolean{LWR@doingstartpars} \AND%
        \cnttest{\value{LWR@lateximagedepth}}{=}{0}%
    }%
        {\par}%
        {%
            \LWR@htmltagc{/\LWR@tagregularparagraph}%
            \LWR@orignewline%
        }%
```

Close the group:

```
    \color@endgroup%
}% vbox
```

Paragraph handling:

```
\LWR@ensuredoingapar%
}%
```

\@footnotetext {⟨*text*⟩}

```
\LetLtxMacro\@footnotetext\LWR@footnotetext
```

58.5 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.

```
\newbox\LWR@mpfootnotes
```

\@mpfootnotetext {⟨*text*⟩}

```
\long\def\@mpfootnotetext#1{%
\LWR@traceinfo{@mpfootnotetext}%
\global\setbox\LWR@mpfootnotes\vbox{%
    \unvbox\LWR@mpfootnotes%
    \reset@font\footnotesize%
    \hsize\columnwidth%
    \@parboxrestore%
    \protected@edef\@currentlabel%
        {\csname p@mpfootnote\endcsname\@thefnmark}%
    \color@begingroup%
```

Use paragraph tags if in a tabular data cell or a `lateximage`:

```
    \ifthenelse{%
        \boolean{LWR@doingstartpars} \AND%
        \cnttest{\value{LWR@lateximagedepth}}{=}{0}%
    }%
    {}%
    {\LWR@htmltagc{\LWR@tagregularparagraph}\LWR@orignewline}%

    \@makefntext{%
        \ignorespaces#1%
    }%
```

Don't add the closing paragraph tag if are inside a `lateximage`:

```
    \ifthenelse{\cnttest{\value{LWR@lateximagedepth}}{>}{0}}%
        {}%
        {%
            \LWR@htmltagc{/\LWR@tagregularparagraph}%
            \LWR@orignewline%
        }%
    \color@endgroup%
}% vbox
```

Paragraph handling:

```
\LWR@ensuredoingapar%
\LWR@traceinfo{@mpfootnotetext: done}%
}
```

\thempfootnote Redefined to remove the `\itshape`, which caused an obscure compiling error in some situations.

```
\AtBeginDocument{
\def\thempfootnote{\@alph\c@mpfootnote}
}
```

58.6 Printing pending footnotes

\LWR@printpendingfootnotes Enclose the footnotes in a class, print, then clear.

```
\newcommand*{\LWR@printpendingfootnotes}{%
\ifvoid\LWR@footnotes\else
    \LWR@forcenewpage
    \begin{BlockClass}{footnotes}
    \LWR@origmedskip
    \unvbox\LWR@footnotes
    \setbox\LWR@footnotes=\vbox{}
    \end{BlockClass}
\fi
}
```

\LWR@maybeprintpendingfootnotes {⟨*depth*⟩} Used to print footnotes before sections only if formatting for an EPUB or word processor:

```
\newcommand*{\LWR@maybeprintpendingfootnotes}[1]{%
\ifboolexpr{
    not test{\ifnumcomp{#1}{>}{\value{FootnoteDepth}}} or
    bool{FormatEPUB} or
    bool{FormatWP}
}%
{\LWR@printpendingfootnotes}%
{}%
}
```

\LWR@printpendingmpfootnotes Enclose the minipage footnotes in a class, print, then clear.

```
\newcommand*{\LWR@printpendingmpfootnotes}{%
\ifvoid\LWR@mpfootnotes\else
    \LWR@forcenewpage
    \begin{BlockClass}{footnotes}
    \LWR@print@vspace*{\baselineskip}
    \unvbox\LWR@mpfootnotes
    \setbox\LWR@mpfootnotes=\vbox{}
    \end{BlockClass}
\fi
}
```

```
\end{warpHTML}
```

59 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 instead of a . A line break will occur in the text where the \marginBlock occurs.

for HTML output:

```
\begin{warpHTML}
```

\marginpar [⟨*left*⟩] {⟨*right*⟩}

```
\renewcommand{\marginpar}[2][]{%
\ifbool{FormatWP}%
{%
\begin{LWR@BlockClassWP}{width:2in; float:right; margin:10pt}{}{marginblock}
#2
\end{LWR@BlockClassWP}
}%
{%
    \LWR@htmlspanclass{marginpar}{#2}%
}%
}
```

\marginparBlock [⟨*left*⟩] {⟨*right*⟩}

For use when the marginpar will be more than one paragraph, and/or contains more than simple text.

HTML version.

```
\newcommand{\marginparBlock}[2][]{%
\ifbool{FormatWP}%
{%
\begin{LWR@BlockClassWP}{width:2in; float:right; margin:10pt}{}{marginblock}
#2
\end{LWR@BlockClassWP}
}%
{%
\begin{BlockClass}[width:2in; float:right; margin:10pt]{marginparblock}
#2
\end{BlockClass}
}%
}
```

\reversemarginpar

```
\renewcommand*{\reversemarginpar}{}
```

\normalmarginpar

```
\renewcommand*{\normalmarginpar}{}
```

```
\end{warpHTML}
```

for PRINT output:

```
\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.

```
\LetLtxMacro\marginparBlock\marginpar
```

```
\end{warpprint}
```

60 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:

```
\begin{warpall}
```

Ctr FileDepth {⟨*section depth*⟩} determines how deeply to break into new HTML files, similar to `tocdepth`. The default of -5 produces one large HTML file.

```
\newcounter{FileDepth}
\setcounter{FileDepth}{-5}
```

Bool CombineHigherDepths Combile higher-level sections together into one file?

```
\newbool{CombineHigherDepths}
\booltrue{CombineHigherDepths}
```

\FilenameLimit Maximum length of the generated filenames.

```
\newcommand*{\FilenameLimit}{80}
```

```
\end{warpall}
```

for HTML output:

```
\begin{warpHTML}
```

\LWR@thisfilename The currently-active filename or number. At first, this is the homepage.

```
\AtBeginDocument{
\ifbool{FileSectionNames}%
    {\newcommand*{\LWR@thisfilename}{\HomeHTMLFilename}}
    {\newcommand*{\LWR@thisfilename}{0}}
}
```

\LWR@thisnewfilename The filename being sanitized.

```
\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.

```
\NewDocumentCommand{\LWR@simplifyname}{s m}{%
\IfBooleanTF{#1}{%
    \StrSubstitute{\LWR@thisnewfilename}%
        {\detokenize{#2}}%
        {\detokenize{-}}[\LWR@thisnewfilename]%
}{%
    \StrSubstitute{\LWR@thisnewfilename}%
        {#2}%
        {\detokenize{-}}[\LWR@thisnewfilename]%
}
}
```

\LWR@simplifycustom User-defined filename simplifications. Redefine with \newcommand.

```
\newcommand*{\LWR@simplifycustom}{}
```

\FilenameSimplify * {⟨*phrase*⟩} Assign a user-defined filename simplification. Appends to \LWR@simplifycustom.

```
\NewDocumentCommand{\FilenameSimplify}{s m}{%
\IfBooleanTF{#1}{%
    \appto{\LWR@simplifycustom}{%
        \LWR@simplifyname*{#2}%
    }%
}{%
    \appto{\LWR@simplifycustom}{%
        \LWR@simplifyname{#2}%
    }%
}%
}
```

\LWR@filenamenoblanks {⟨*filename*⟩}

Convert blanks into dashes, removes short words, store result in \LWR@thisfilename.

Also see \LWR@nullfonts for nullified macros.

```
\newcommand*{\LWR@filenamenoblanks}[1]{%
\begingroup
```

Locally temporarily disable direct-formatting commands, not used in filenames:

```
\LWR@nullfonts%
\renewcommand*{\LWR@htmltagc}[1]{}%

\edef\LWR@thisnewfilename{#1}%
```

Replaces common macros with hyphens. (\& is done by \LWR@nullfonts.)

```
\RenewDocumentCommand{\LWR@subsingledollar}{s m m m}{}%
\LWR@simplifyname{\_}
\LWR@simplifyname{\#}
\LWR@simplifyname{\textbackslash}
\LWR@simplifyname{\protect}
\LWR@simplifyname{\ }
\LWR@simplifyname{\textless}
\LWR@simplifyname{\textgreater}

\edef\LWR@thisnewfilename{\detokenize\expandafter{\LWR@thisnewfilename}}%

\LWR@traceinfo{LWR@filenamenoblanks edef: !\LWR@thisnewfilename!}%
\fullexpandarg%
```

Convert spaces into hyphens:

```
\LWR@simplifyname*{ }
```

Convert punctutation into hyphens:

```
\LWR@simplifyname*{!}
\LWR@simplifyname*{,}
\LWR@simplifyname*{'}
\LWR@simplifyname*{+}
\LWR@simplifyname*{,}
\LWR@simplifyname*{/}
\LWR@simplifyname*{:}
\LWR@simplifyname*{;}
\LWR@simplifyname*{=}
\LWR@simplifyname*{?}
\LWR@simplifyname*{@}
\LWR@simplifyname*{^}
\LWR@simplifyname*{&}
\LWR@simplifyname*{"}
\LWR@simplifyname*{<}
\LWR@simplifyname*{>}

\LWR@simplifyname{\LWRbackslash}
```

Braces are removed entirely to avoid extra dashes in the result.

```
\StrSubstitute{\LWR@thisnewfilename}%
    {\LWRleftbrace}{}[\LWR@thisnewfilename]%
\StrSubstitute{\LWR@thisnewfilename}%
    {\LWRrightbrace}{}[\LWR@thisnewfilename]%
```

```
\LWR@simplifyname{\LWRpercent}
\LWR@simplifyname{\LWRdollar}

\LWR@simplifyname*{|}
\LWR@simplifyname*{^}
\LWR@simplifyname*{~}
\LWR@simplifyname*{[}
\LWR@simplifyname*{]}
\LWR@simplifyname*{‘}
```

Convert short words:

```
\LWR@simplifyname*{-s-}
\LWR@simplifyname*{-S-}
\LWR@simplifyname*{-a-}
\LWR@simplifyname*{-A-}
\LWR@simplifyname*{-an-}
\LWR@simplifyname*{-AN-}
\LWR@simplifyname*{-to-}
\LWR@simplifyname*{-TO-}
\LWR@simplifyname*{-by-}
\LWR@simplifyname*{-BY-}
\LWR@simplifyname*{-of-}
\LWR@simplifyname*{-OF-}
\LWR@simplifyname*{-and-}
\LWR@simplifyname*{-AND-}
\LWR@simplifyname*{-for-}
\LWR@simplifyname*{-FOR-}
\LWR@simplifyname*{-the-}
\LWR@simplifyname*{-THE-}
```

Convert custom words:

```
\LWR@simplifycustom%
```

Convert multiple hyphens:

```
\LWR@simplifyname*{-----}
\LWR@simplifyname*{----}
\LWR@simplifyname*{---}
\LWR@simplifyname*{--}
```

If pdfLaTeX and not `utf8` encoding, don't try to convert emdash, endash:

```
\ifPDFTeX% pdflatex or dvi latex
\ifdefstring{\inputencodingname}{utf8}{%
\LWR@simplifyname*{---}
%     emdash
\LWR@simplifyname*{--}
%     endash
}{}%
\else% not PDFTeX
\LWR@simplifyname*{---}
\LWR@simplifyname*{--}
\fi%
```

If starts with a dash, remove the leading dash:

```
\IfBeginWith{\LWR@thisnewfilename}{\detokenize{-}}{%
    \StrGobbleLeft{\LWR@thisnewfilename}{1}[\LWR@thisnewfilename]%
}{}%
```

If ends with a dash, remove the trailing dash:

```
\IfEndWith{\LWR@thisnewfilename}{\detokenize{-}}{%
    \StrGobbleRight{\LWR@thisnewfilename}{1}[\LWR@thisnewfilename]%
}{}%
```

Limits the length of the filename:

```
\StrLeft{\LWR@thisnewfilename}{\FilenameLimit}[\LWR@thisnewfilename]%
```

Return the global result:

```
\global\let\LWR@thisfilename\LWR@thisnewfilename%
\endgroup%
\LWR@traceinfo{LWR@filenamenoblanks: result is \LWR@thisfilename}%
}
```

Ctr `LWR@previousautopagelabel` Remembers which autopage label was most recently generated. Used to avoid duplicates.

```
\newcounter{LWR@previousautopagelabel}
\setcounter{LWR@previousautopagelabel}{-1}
```

File `*_html.aux` A new entry in the `*_html.aux` file is used to help cross-references:

```
\newlabel{autopage-<nnn>}{{<x>}{<y>}}
```

`\LWR@newautopagelabel` {⟨*pagenumber counter*⟩}

```
\newcommand*{\LWR@newautopagelabel}[1]{%
\ifnumequal{\value{LWR@previousautopagelabel}}{\value{page}}%
{}% no action if this autopage label has already been defined
{%
    \label{autopage-\arabic{#1}}%
    \setcounter{LWR@previousautopagelabel}{\value{page}}
}%
}
```

60.1 Customizing MathJax

`\LWR@customizedMathJax` Additional MathJax definitions to be added to the start of each HTML page.

```
\newcommand*{\LWR@customizedMathJax}{}
```

\CustomizeMathJax MathJax does not have preexisting support every possible math function. Additional MathJax function definitions may be defined. These will be declared at the start of each HTML page, and thus will have a global effect.

Examples:

```
\CustomizeMathJax{
    \newcommand{\expval}[1]{\langle#1\rangle}
    \newcommand{\abs}[1]{\lvert#1\rvert}
}
\CustomizeMathJax{\newcommand{\arsinh}{\text{arsinh}}}
\CustomizeMathJax{\newcommand{\arcosh}{\text{arcosh}}}
\CustomizeMathJax{\newcommand{\NN}{\mathbb{N}}}
```

```
\newcommand*{\CustomizeMathJax}[1]{%
   \appto{\LWR@customizedMathJax}{%
       \(#1\)\par
   }%
}
```

\LWR@customizeMathJax

```
\newcommand{\LWR@customizeMathJax}{%
\ifbool{mathjax}{

\LWR@stoppars
\LWR@htmlcomment{Nullify \textbackslash{}ensuremath, footnotes for MathJax:}

\(\newcommand\ensuremath[1]{##1}\)

\(\newcommand\footnote[2][]{\text{( Footnote ##1 )}}\)

\(\newcommand\footnotemark[1][]{\text{( Footnote ##1 )}}\)

\LWR@htmlcomment{Additional customizations for MathJax:}

\LWR@customizedMathJax

\LWR@startpars
}{}
}

\end{warpHTML}
```

for PRINT output:

```
\begin{warpprint}
```

\CustomizeMathJax The print-mode version:

```
\newcommand*{\CustomizeMathJax}[1]{}
```

\FilenameSimplify * {⟨*expression*⟩}

```
\NewDocumentCommand{\FilenameSimplify}{s m}{}
```

```
\end{warpprint}
```

for HTML output:

```
\begin{warpHTML}
```

`\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.

```
\newcommand*{\LWR@newhtmlfile}[1]{
\LWR@traceinfo{LWR@newhtmlfile}
```

At the bottom of the ending file:

```
\LWR@htmlelementclassend{section}{textbody}
\LWR@htmlelementclassend{div}{bodycontainer}
\LWR@htmlelementclassend{div}{bodyandsidetoc}

\LWR@printpendingfootnotes

```

No footer between files if EPUB:

```
\ifbool{FormatEPUB}
{}
{
    \LWR@htmlelement{footer}

    \LWR@pagebottom

    \LWR@htmlelementend{footer}
}
```

No bottom navigation if are finishing the home page or formatting for EPUB or a word-processor.

```
\ifthenelse{\boolean{FormatEPUB}\OR\boolean{FormatWP}}
    {}
    {\ifnumcomp{\value{LWR@htmlfilenumber}}{>}{0}{\LWR@botnavigation}{}}
```

End of this HTML file:

```
\LWR@stoppars
\LWR@htmltag{/body}\LWR@orignewline
\LWR@htmltag{/html}\LWR@orignewline
\LWR@traceinfo{LWR@newhtmlfile: about to LWR@orignewpage}
\LWR@orignewpage

\addtocounter{LWR@htmlfilenumber}{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.

```
\ifbool{FileSectionNames}%
{%
```

Convert the section name to a filename with blanks and common words removed. The resulting filename is in `\LWR@thisfilename`.

```
    \LWR@filenamenoblanks{#1}%
```

Create a macro name from the MD5 hash of the file name, to detect duplicates:

```
    \edef\LWR@hashedname{\LWR@mdfive{\LWR@thisfilename}}%
```

If the macro name is not yet defined, this filename is unique.

```
    \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.

```
        \csdef{LWR@filename\LWR@hashedname}{}%
    }{%
```

If the filename is not unique, create an error.

```
        \PackageError{lwarp}
            {
                Section name ``#1'', at the line number listed below,
                generates the filename ``\LWR@thisfilename'',
                which appears to be a duplicate.
               There is a previous section with an identical or similar name.%
            }
            {
                Lwarp sanitizes most symbols and a few common short words
                when generating file names, and this may cause a conflict.
            }
    }
}
```

If using file numbers instead of names, the name is set to the next file number.

```
{\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.

```
\LWR@traceinfo{LWR@newhtmlfile: about to print start file}%
```

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

```
\begingroup%
\LWR@nullfonts%
\LWR@htmlblockcomment{%
|Start file|%
\LWR@htmlsectionfilename{\LWR@thisfilename}|%
}
\endgroup%
```

At the top of the starting file:

```
\LWR@stoppars

```

Start a new file with the given section name:

```
\LWR@filestart[#1]

```

Track the page numbers:

```
\setcounter{LWR@latestautopage}{\value{page}}%
\LWR@newautopagelabel{LWR@latestautopage}%
```

No navigation between files if formatting for an EPUB or word processor:

```
\ifthenelse{\boolean{FormatEPUB}\OR\boolean{FormatWP}}
    {}
    {\LWR@topnavigation}

```

No header if between files if formatting for an EPUB or word processor:

```
\ifthenelse{\boolean{FormatEPUB}\OR\boolean{FormatWP}}
    {}
    {
        \LWR@htmlelement{header}

        \LWR@pagetop

        \LWR@htmlelementend{header}
    }

```

The container for the sidetoc and text body:

```
\LWR@htmlelementclass{div}{bodyandsidetoc}
```

No sideTOC if formatting for an EPUB or word processor:

```
\ifthenelse{\boolean{FormatEPUB}\OR\boolean{FormatWP}}
    {}
    {\LWR@sidetoc}

```

Start of the <textbody>:

```
\LWR@htmlelementclass{div}{bodycontainer}
\LWR@htmlelementclass{section}{textbody}

```

Print title only if there is one. Skip if formatting for an EPUB or word processor:

```
\ifthenelse{\boolean{FormatEPUB}\OR\boolean{FormatWP}}%
    {}%
    {%
        \ifcsvoid{thetitle}{}{%
            \LWR@printthetitle%
        }%
    }%
```

Keep paragraph tags disabled for now:

```
\LWR@stoppars

```

If using MathJax, disable \ensuremath by printing a nullified definition at the start of each file, and add further customizations:

```
\LWR@customizeMathJax

\LWR@traceinfo{LWR@newhtmlfile: done}
}

\end{warpHTML}
```

61 Sections

Sections 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

⚠ accents in filenames

Section names and the resulting filenames with accented characters are partially supported, depending on the ability of *pdflatex* to generate characters and *pdftotext* 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.

XeLaTeX and LuaLaTeX directly support accented section and file names.

for HTML output:

```
\begin{warpHTML}
```

61.1 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.

```
\newbool{LWR@forcinghtmlpage}
\boolfalse{LWR@forcinghtmlpage}

\newcommand*{\ForceHTMLPage}{%
\global\booltrue{LWR@forcinghtmlpage}%
}
```

\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.

```
\newbool{LWR@forcinghtmltoc}
\boolfalse{LWR@forcinghtmltoc}

\newcommand*{\ForceHTMLTOC}{%
\global\booltrue{LWR@forcinghtmltoc}%
}

\end{warpHTML}
```

for PRINT output:

```
\begin{warpprint}
\newcommand*{\ForceHTMLPage}{}
\newcommand*{\ForceHTMLTOC}{}
\end{warpprint}
```

for HTML output:

```
\begin{warpHTML}
```

61.2 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.

```
\newbool{LWR@mainmatter}
\DeclareDocumentCommand{\mainmatter}{}{%
\booltrue{LWR@mainmatter}%
}
```

\frontmatter Declare the front matter section of the document, using arabic numbering for the internal numbering. Does not reset the page number.

```
\DeclareDocumentCommand{\frontmatter}{}{%
\boolfalse{LWR@mainmatter}%
}
```

`\backmatter` Declare the back matter section of the document. Does not reset the page number.

```
\DeclareDocumentCommand{\backmatter}{}{%
\boolfalse{LWR@mainmatter}
}
```

61.3 Sectioning support macros

`\LWR@sectionumber` {⟨*section type*⟩}

Typeset a section number and its trailing space with CSS formatting:

```
\newcommand*{\LWR@sectionnumber}[1]{%
\InlineClass{sectionnumber}{#1}%
}
```

`autosec` A tag used by the TOC and index.

`\LWR@createautosec` {⟨*section type*⟩}

Create an autosection tag.

```
\newcommand*{\LWR@createautosec}[1]{%
\LWR@htmltag{%
    #1 % space
    id=\textquotedbl\LWR@print@mbox{autosec-\arabic{page}}\textquotedbl%
}%
}
```

`\LWR@pushoneclose` {⟨*depth*⟩} {⟨*printclose*⟩} 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.

```
\NewDocumentCommand{\LWR@pushoneclose}{m m}{%
\LWR@traceinfo{LWR@pushoneclose #1}%
    \pushclose{#2}{#1}%
}
```

`\LWR@startnewdepth` {⟨*depth*⟩} {⟨*printclose*⟩}

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.

```
\NewDocumentCommand{\LWR@startnewdepth}{m m}{%
```

Close any stacked sections up to this new one.

```
\LWR@closeprevious{#1}%
```

Push a new section depth:

```
\LWR@pushoneclose{#1}{#2}%
}
```

Ctr `LWR@prevFileDepth` Remembers the previous `LWR@FileDepth`.

Initialized to a deep level so that any section will trigger a new HTML page after the home page.

```
\newcounter{LWR@prevFileDepth}
\setcounter{LWR@prevFileDepth}{\LWR@depthsubparagraph}
```

`\@seccntformat` {⟨*sectiontype*⟩}

```
\def\@seccntformat#1{\csname the#1\endcsname\quad}
```

`\simplechapterdelim` Used by tocbibind and anonchap.

```
\newcommand*{\simplechapterdelim}{}
```

`\@chapcntformat` {⟨*sectiontype*⟩}

`\let` to `\@seccntformat` by default, but may be redefined by `\simplechapter` and `\restorechapter` from tocbibind or anonchap.

```
\let\@chapcntformat\@seccntformat
```

`\@partcntformat` {⟨*sectiontype*⟩}

`\let` to `\@seccntformat` by default, but may be redefined by ctex.

```
\let\@partcntformat\@seccntformat
```

`\@partnameformat` Prints “Part” for part sections.

Nullified by ctex.

```
\newcommand*{\@partnameformat}{\LWR@isolate{\partname}~}%
```

Ctr `LWR@currentautosec` 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.

```
\newcounter{LWR@currentautosec}
\setcounter{LWR@currentautosec}{1}
```

\LWR@section * [⟨*TOC name*⟩] {⟨*name*⟩} {⟨*sectiontype*⟩}

The common actions for the high-level sectioning commands.

```
\DeclareDocumentCommand{\LWR@section}{m m m m}{%
\LWR@traceinfo{LWR@section: starting}%
\LWR@stoppars%
```

Cancel special `minipage` horizontal space interaction:

```
\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:

```
\LWR@traceinfo{LWR@section: testing whether to start a new HTML file}%
\IfBooleanT{#1}{\LWR@traceinfo{LWR@section: starred}}%
\ifbool{LWR@forcinghtmlpage}{\LWR@traceinfo{LWR@section: forcinghtmlpage}}{}%
\ifthenelse{%
    \(%
        \(\NOT\equal{#1}{\BooleanTrue}\)\OR%
        \(\cnttest{\@nameuse{LWR@depth#4}}{=}{\LWR@depthpart}\)\OR%
        \(\boolean{LWR@forcinghtmlpage}\)%
    \)%
    \AND%
    \cnttest{\@nameuse{LWR@depth#4}}{<=}{\value{FileDepth}}%
    \AND%
    \(%
        \NOT\boolean{CombineHigherDepths}\OR%
        \cnttest{\@nameuse{LWR@depth#4}}{<=}{\value{LWR@prevFileDepth}}%
    \)%
    \AND%


    \(% phantomsection
        \NOT\isempty{#3}%
        \OR%
        \(\NOT\equal{#1}{\BooleanTrue}\)%
    \)%
}%
```

If so: start a new HTML file:

```
{% new file
    \LWR@traceinfo{LWR@section: new HTML file}%
```

See if there was an optional TOC name entry:

```
    \IfNoValueTF{#2}%
```

If no optional entry

```
        {\LWR@newhtmlfile{#3}}%
```

If yes an optional entry

```
        {\LWR@newhtmlfile{#2}}%
}% new file
```

Else: No new HTML file:

```
{% not new file
```

Generate a new LaTeX page so that TOC and index page number points to the section:

```
    \LWR@traceinfo{LWR@section: not a new HTML file, about to LWR@orignewpage}%
    \LWR@orignewpage%
}% not new file

```

Remember this section's name for `\nameref`:

```
\IfValueT{#3}{%
    \LWR@traceinfo{LWR@section: about to LWR@setlatestname}%
    \IfValueTF{#2}{\LWR@setlatestname{#2}}{\LWR@setlatestname{#3}}%
}%
```

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.

```
\ifbool{HTMLDebugComments}{%
    \begingroup%
    \LWR@nullfonts%
    \IfBooleanTF{#1}% starred
    {\LWR@htmlcomment{Opening #4*}}%
    {%
        \IfNoValueTF{#2}% short TOC
            {\LWR@htmlcomment{Opening #4 ``#3''}}%
            {\LWR@htmlcomment{Opening #4 ``#2''}}%
    }\LWR@orignewline%
    \endgroup%
}{}
```

For inline sections paragraph and subparagraph, start a new paragraph now:

```
\ifthenelse{%
    \cnttest{\@nameuse{LWR@depth#4}}{>=}{\LWR@depthparagraph}%
```

```
}%
    {\LWR@startpars}%
    {}%
```

Create the opening tag with an autosec:

```
\LWR@traceinfo{LWR@section: about to LWR@createautosec}%
\LWR@createautosec{\@nameuse{LWR@tag#4}}%

\setcounter{LWR@currentautosec}{\value{page}}%
```

Check if starred:

```
\IfBooleanTF{#1}%
{%
    \LWR@traceinfo{LWR@section: starred}%
```

Starred, but also forcing a TOC entry, so add unnumbered TOC name or regular name:

```
    \ifbool{LWR@forcinghtmltoc}%
    {%
        \addcontentsline{toc}{#4}{%
            \IfValueTF{#2}{\LWR@isolate{#2}}{\LWR@isolate{#3}}%
        }%
    }%
    {}%
}% starred
```

Not starred, so step counter and add to TOC:

```
{% not starred
```

Only add a numbered TOC entry if section number is not too deep:

```
    \ifthenelse{%
        \cnttest{\@nameuse{LWR@depth#4}}{<=}{\value{secnumdepth}}%
    }%
    {% if secnumdepth
```

If in the main matter, step the counter and add the TOC entry. For `article` class, lwarp assumes that all is mainmatter.

```
        \LWR@traceinfo{LWR@section: about to test main matter}%
        \ifbool{LWR@mainmatter}%
        {%
            \LWR@traceinfo{LWR@section: yes mainmatter}%
            \refstepcounter{#4}%
```

Add main matter numbered TOC entry with the TOC name or the regular name:

```
            \LWR@traceinfo{LWR@section: about to addcontentsline}%
            \addcontentsline{toc}{#4}%
            {%
```

```
                \protect\numberline{%
                    \@nameuse{pre#4name}%
                    \@nameuse{the#4}%
                    \@nameuse{post#4name}%
                }%
                {%
                    \ignorespaces%
             \IfValueTF{#2}{\LWR@isolate{#2}}{\LWR@isolate{#3}}\protect\relax%
                }%
            }%
            \LWR@traceinfo{LWR@section: finished addcontentsline}%
        }% end of if main matter
```

If not main matter, add unnumbered TOC name or regular name:

```
        {% not main matter
            \LWR@traceinfo{LWR@section: no main matter}%
            \addcontentsline{toc}{#4}{%
                \IfValueTF{#2}{\LWR@isolate{#2}}{\LWR@isolate{#3}}%
            }%
        }% end of not main matter
    }% end of secnumdepth
```

Deeper than secnumdepth, so add an unnumbered TOC entry:

```
    {%
        \addcontentsline{toc}{#4}{%
            \IfValueTF{#2}{\LWR@isolate{#2}}{\LWR@isolate{#3}}%
        }%
    }%
```

For part, print “Part”:

```
    \ifbool{LWR@mainmatter}%
    {%
        \ifthenelse{%
            \(\cnttest{\@nameuse{LWR@depth#4}}{<=}%
                {\value{secnumdepth}}\) \AND%
            \(\cnttest{\@nameuse{LWR@depth#4}}{=}{\LWR@depthpart}\)%
        }%
            {\@partnameformat}%
            {}%
```

Print the section number:

```
        \LWR@traceinfo{LWR@section: about to print section number}%
        \ifthenelse{%
            \cnttest{\@nameuse{LWR@depth#4}}{<=}{\value{secnumdepth}}%
        }%
            {%
                \ifstrequal{#4}{part}%
                {\protect\LWR@sectionnumber{\@partcntformat{#4}}}%
                {%
                    \ifstrequal{#4}{chapter}%
                        {\protect\LWR@sectionnumber{\@chapcntformat{#4}}}%
```

```
                        {\protect\LWR@sectionnumber{\@seccntformat{#4}}}%
                }%
            }%
            {}%
        \LWR@traceinfo{LWR@section: finished print section number}%
    }{}%
}% end of not starred
```

Print the section name:

```
\LWR@traceinfo{LWR@section: about to print the section name}%
\LWR@isolate{#3}%
```

Close the heading tag, such as /H2:

```
\LWR@traceinfo{LWR@section: about to close the heading tag}%
\LWR@htmltag{\@nameuse{LWR@tag#4end}}%
\LWR@orignewline%
```

Generate a LaTeX label:

```
\LWR@traceinfo{LWR@section: about to create the LaTeX label}%
\setcounter{LWR@latestautopage}{\value{page}}%
\LWR@newautopagelabel{LWR@currentautosec}\LWR@orignewline%
```

Start paragraph handing unless is an inline paragraph or subparagraph:

```
\ifthenelse{%
    \cnttest{\@nameuse{LWR@depth#4}}{<}{\LWR@depthparagraph}%
}%
    {\LWR@startpars}%
    {}%
```

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.

```
\ifthenelse{%
    \NOT\equal{#1}{\BooleanTrue}\OR%
    \cnttest{\@nameuse{LWR@depth#4}}{=}{\LWR@depthpart}%
}%
    {% not starred
        \setcounter{LWR@prevFileDepth}{\@nameuse{LWR@depth#4}}%
    }% not starred
    {}%
```

Reset to defaults if not a phantomsection:

```
\ifstrempty{#3}%
```

```
    {}%
    {%
        \global\boolfalse{LWR@forcinghtmlpage}%
        \global\boolfalse{LWR@forcinghtmltoc}%
    }%
%
\LWR@traceinfo{LWR@section: done}%
}
```

61.4 Pre- and post- sectioning names

\prepartname
\postpartname

Usually null, but is used by uj* and ut* Japanese classes.

```
\providecommand*{\prepartname}{}
\providecommand*{\postpartname}{}
```

\prechaptername
\postchaptername

Usually null, but is used by uj* and ut* Japanese classes.

```
\providecommand*{\prechaptername}{}
\providecommand*{\postchaptername}{}
```

\presectionname
\postsectionname

Always null, but provided here for algorithmic simplicity in \LWR@section.

```
\providecommand*{\presectionname}{}
\let\postsectionname\presectionname

\let\presubsectionname\presectionname
\let\postsubsectionname\postsectionname

\let\presubsubsectionname\presectionname
\let\postsubsubsectionname\postsectionname

\let\preparagraphname\presectionname
\let\postparagraphname\postsectionname

\let\presubparagraphname\presectionname
\let\postsubparagraphname\postsectionname
```

61.5 \section and friends

\part * [⟨*TOC name*⟩] {⟨*name*⟩}

```
\newcommand{\part@preamble}{}% for koma-script

\DeclareDocumentCommand{\part}{s o m}{%
\LWR@maybeprintpendingfootnotes{\LWR@depthpart}%
\LWR@stoppars%

```

```
\LWR@startnewdepth{\LWR@depthpart}{\LWR@printclosepart}%

\LWR@section{#1}{#2}{#3}{part}%

\part@preamble% for koma-script
\renewcommand{\part@preamble}{}%
}
```

\chapter * [⟨*TOC name*⟩] [⟨*heading name*⟩] {⟨*name*⟩}

```
\let\@printcites\relax% for quotchap package

\newcommand{\chapter@preamble}{}% for koma-script

\@ifundefined{chapter}
{}
{%
\DeclareDocumentCommand{\chapter}{s o o m}{%
\IfValueTF{#2}{
\LWR@traceinfo{chapter #2}%
}{
\LWR@traceinfo{chapter #4}%
}
\LWR@maybeprintpendingfootnotes{\LWR@depthchapter}%
\LWR@stoppars%

\LWR@startnewdepth{\LWR@depthchapter}{\LWR@printclosechapter}%

\LWR@section{#1}{#2}{#4}{chapter}%

\@printcites% for quotchap package

\chapter@preamble% for koma-script
\renewcommand{\chapter@preamble}{}%
}
}
```

\section * [⟨*TOC name*⟩] [⟨*heading name*⟩] {⟨*name*⟩}

```
\DeclareDocumentCommand{\section}{s o o m}{%
\LWR@traceinfo{section: starting}%
\LWR@maybeprintpendingfootnotes{\LWR@depthsection}%
\LWR@stoppars%

\LWR@startnewdepth{\LWR@depthsection}{\LWR@printclosesection}%

\LWR@section{#1}{#2}{#4}{section}%
}
```

\subsection * [⟨*TOC name*⟩] {⟨*name*⟩}

```
\DeclareDocumentCommand{\subsection}{s o m}{%
\LWR@maybeprintpendingfootnotes{\LWR@depthsubsection}%
```

```
\LWR@stoppars%

\LWR@startnewdepth{\LWR@depthsubsection}{\LWR@printclosesubsection}%

\LWR@section{#1}{#2}{#3}{subsection}%
}
```

\subsubsection * [⟨*TOC name*⟩] {⟨*name*⟩}

```
\DeclareDocumentCommand{\subsubsection}{s o m}{%
\LWR@maybeprintpendingfootnotes{\LWR@depthsubsubsection}%
\LWR@stoppars%

\LWR@startnewdepth{\LWR@depthsubsubsection}%
{\LWR@printclosesubsubsection}%

\LWR@section{#1}{#2}{#3}{subsubsection}%
}
```

\paragraph * [⟨*TOC name*⟩] {⟨*name*⟩}

```
\DeclareDocumentCommand{\paragraph}{s o m}{%
\LWR@maybeprintpendingfootnotes{\LWR@depthparagraph}%
\LWR@stoppars%

\LWR@startnewdepth{\LWR@depthparagraph}{\LWR@printcloseparagraph}%

\LWR@section{#1}{#2}{#3}{paragraph}%
}
```

\subparagraph * [⟨*TOC name*⟩] {⟨*name*⟩}

```
\DeclareDocumentCommand{\subparagraph}{s o m}{%
\LWR@maybeprintpendingfootnotes{\LWR@depthsubparagraph}%
\LWR@stoppars%

\LWR@startnewdepth{\LWR@depthsubparagraph}{\LWR@printclosesubparagraph}%

\LWR@section{#1}{#2}{#3}{subparagraph}%
}

\end{warpHTML}
```

62 Starting a new file

for HTML & PRINT:

```
\begin{warpall}
```

\HTMLLanguage Default language for the HTML lang tag.

```
\newcommand*{\LWR@currentHTMLLanguage}{en-US}

\newcommand*{\HTMLLanguage}[1]{%
\renewcommand*{\LWR@currentHTMLLanguage}{#1}%
}
```

\theHTMLTitleSeparator May be used inside \theHTMLTitleSection to separate the website's overall HTML title and the particular page's section name.

```
\ifPDFTeX% pdflatex or dvi latex
    \ifdefstring{\inputencodingname}{utf8}{%
        \newcommand*{\theHTMLTitleSeparator}{ --- }% EMdash
    }{%
        \newcommand*{\theHTMLTitleSeparator}{ - }% hyphen
    }%
\else%
    \ifpTeX
        \newcommand*{\theHTMLTitleSeparator}{ - }% hyphen
    \else
        \newcommand*{\theHTMLTitleSeparator}{ --- }% EMdash
    \fi%
\fi%
```

\HTMLTitleBeforeSection Sets the HTML page's meta `title` tag to show the website title before the section name.

```
\newcommand*{\HTMLTitleBeforeSection}{%
    \def\theHTMLTitleSection{%
        \theHTMLTitle\theHTMLTitleSeparator\theHTMLSection%
    }%
}
```

\HTMLTitleAfterSection Sets the HTML page's meta `title` tag to show the section name before the website title.

```
\newcommand*{\HTMLTitleAfterSection}{%
    \def\theHTMLTitleSection{%
        \theHTMLSection\theHTMLTitleSeparator\theHTMLTitle%
    }%
}
```

\theHTMLTitleSection Forms the HTML page's meta `title` tag. The default is to show the website title before the section name.

```
\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.

```
\newcommand*{\theHTMLSection}{}

\end{warpall}
```

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

`\LWR@filestart` `[`⟨*sectionname*⟩`]` Creates the opening HTML tags.

```
\newcommand*{\LWR@filestart}[1][]{%
\LWR@traceinfo{LWR@filestart !#1!}%
```

Locally temporarily disable direct-formatting commands:

```
\begingroup%
\LWR@nullfonts%
```

Save the section name for use while creating the HTML meta `title` tag:

```
\edef\theHTMLSection{#1}%
```

Create the page's HTML header:

```
\LWR@htmltag{!DOCTYPE html}\LWR@orignewline
```

The language is user-adjustable:

```
\LWR@htmltag{%
   html lang=\LWR@orig@textquotedbl\LWR@currentHTMLLanguage\LWR@orig@textquotedbl%
}\LWR@orignewline
```

Start of the meta data:

```
\LWR@htmltag{head}\LWR@orignewline
```

Charset is fixed at UTF-8:

```
\LWR@htmltag{%
    meta charset=\LWR@orig@textquotedbl{}UTF-8\LWR@orig@textquotedbl\ /%
}\LWR@orignewline
```

Author:

```
\ifthenelse{\equal{\theHTMLAuthor}{}}%
    {}%
    {%
        \LWR@htmltag{%
        meta name=\LWR@orig@textquotedbl{}author\LWR@orig@textquotedbl\ % space
         content=\LWR@orig@textquotedbl\theHTMLAuthor\LWR@orig@textquotedbl\ /%
        }\LWR@orignewline%
    }%
```

lwarp is the generator:

```
\LWR@htmltag{%
    meta % space
    name=\LWR@orig@textquotedbl{}generator\LWR@orig@textquotedbl\ % space
```

```
   content=\LWR@orig@textquotedbl{}LaTeX lwarp package\LWR@orig@textquotedbl\ /%
}\LWR@orignewline%
```

If there is a description, add it now:

```
\ifdefempty{\LWR@currentHTMLDescription}{}{%
    \LWR@htmltag{%
      meta name=\LWR@orig@textquotedbl{}description\LWR@orig@textquotedbl\ % space
      content=\LWR@orig@textquotedbl\LWR@currentHTMLDescription\LWR@orig@textquotedbl\ /%
    }\LWR@orignewline
}%
```

Mobile-friendly viewport:

```
\LWR@htmltag{%
    meta % space
    name=\LWR@orig@textquotedbl{}viewport\LWR@orig@textquotedbl\ % space
   content=\LWR@orig@textquotedbl{}width=device-width, initial-scale=1.0\LWR@orig@textquotedbl\ /%
}\LWR@orignewline
```

IE patch:

```
\LWR@htmltag{!-\/-[if lt IE 9]}\LWR@orignewline
\LWR@htmltag{%
    script % space
    src=\LWR@orig@textquotedbl{}%
        http://html5shiv.googlecode.com/svn/trunk/html5.js%
    \LWR@orig@textquotedbl%
}%
\LWR@htmltag{/script}\LWR@orignewline
\LWR@htmltag{![endif]-\/-}\LWR@orignewline
```

The page's title, if there is one. A section name is also added if given.

```
\ifthenelse{\equal{\theHTMLTitle}{}}%
    {}%
    {%
        \LWR@htmltag{title}%
        \ifdefempty{\theHTMLSection}%
            {\theHTMLTitle}%
            {\theHTMLTitleSection}%
        \LWR@htmltag{/title}\LWR@orignewline%
    }%
```

The page's stylesheet:

```
\LWR@htmltag{%
    link % space
    rel=\LWR@orig@textquotedbl{}stylesheet\LWR@orig@textquotedbl\ % space
    type=\LWR@orig@textquotedbl{}text/css\LWR@orig@textquotedbl\ % space
    href=\LWR@orig@textquotedbl\LWR@currentcss\LWR@orig@textquotedbl\ /%
}%
\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.

```
\ifbool{mathjax}%
{%
    \begingroup%
    \LWR@restoreoriglists%
    \boolfalse{LWR@verbtags}%

        \verbatiminput{lwarp_mathjax.txt}%

    \booltrue{LWR@verbtags}%
    \endgroup%
    \LWR@stoppars%
}% end of mathjax
{}%
```

End of the header:

```
\LWR@htmltag{/head}\LWR@orignewline
```

Start of the body:

```
\LWR@htmltag{body}\LWR@orignewline
\endgroup
\LWR@traceinfo{LWR@filestart: done}
}

\end{warpHTML}
```

63 Starting HTML output

for HTML output:

```
\begin{warpHTML}
```

`\LWR@LwarpStart` Executed at the beginning of the entire document.

The use of `\textquotedbl` instead of " improves compatibility with xeCJK.

```
\catcode`\$=\active
\newcommand*{\LWR@LwarpStart}
{%
\LWR@traceinfo{LWR@lwarpStart}
```

If formatting for a word processor, force filedepth to single-file only, force HTML debug comments off.

```
\ifbool{FormatWP}{%
    \setcounter{FileDepth}{-5}%
    \boolfalse{HTMLDebugComments}%
}{}
```

Expand and detokenize \HomeHTMLFilename and \HTMLFilename:

```
\edef\LWR@strresult{\HomeHTMLFilename}
\edef\HomeHTMLFilename{\detokenize\expandafter{\LWR@strresult}}
\edef\LWR@strresult{\HTMLFilename}
\edef\HTMLFilename{\detokenize\expandafter{\LWR@strresult}}
```

Force onecolumn and empty page style:

```
\LWR@origonecolumn%
\LWR@origpagestyle{empty}%
```

No black box for overfull lines:

```
\overfullrule=0pt
```

Reduce chance of line overflow when HTML tags are added:

```
\LWR@print@footnotesize%
```

In PDF output, don't allow line breaks to interfere with HTML tags:

```
\LWR@print@raggedright%
\LetLtxMacro{\\}{\LWR@endofline}%
```

Spread the lines for *pdftotext* to read them well:

```
\linespread{1.3}%
```

For *pdftotext* to reliably identify paragraph splits:

```
\setlength{\parindent}{0pt}
\setlength{\parskip}{2ex}
```

For the lateximage record file:

```
\immediate\openout\LWR@lateximagesfile=\BaseJobname-images.txt
```

Removes space around the caption in the HTML:

```
\setlength{\belowcaptionskip}{0ex}
\setlength{\abovecaptionskip}{0ex}
```

Redefine the plain page style to be empty when used by index pages:

```
\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:

```
\let\LWR@origcaption\caption
```

Labels: `\ltx@label` is used in amsmath environments and is also patched by cleveref.

Label in HTML

```
\let\LWR@origltx@label\ltx@label
\let\ltx@label\LWR@htmlmathlabel
```

Logos:

```
\LetLtxMacro\TeX\LWR@TeX
\LetLtxMacro\LaTeX\LWR@LaTeX
\LetLtxMacro\LuaTeX\LWR@LuaTeX
\LetLtxMacro\LuaLaTeX\LWR@LuaLaTeX
\LetLtxMacro\XeTeX\LWR@XeTeX
\LetLtxMacro\XeLaTeX\LWR@XeLaTeX
\LetLtxMacro\ConTeXt\LWR@ConTeXt
```

Not yet started any paragraph handling:

```
\global\boolfalse{LWR@doingapar}
\global\boolfalse{LWR@doingstartpars}
```

Document and page settings:

```
\mainmatter
\LWR@origpagenumbering{arabic}
```

Start a new HTML file and a header:

```
\LWR@traceinfo{LWR@lwarpStart: Starting new file.}
\LWR@filestart
\LWR@traceinfo{LWR@lwarpStart: Generating first header.}
\LWR@htmltag{header}\LWR@orignewline
\LWR@startpars
\LWR@firstpagetop
\LWR@stoppars
\LWR@htmltag{/header}\LWR@orignewline
```

```
\LWR@htmlelementclass{div}{bodywithoutsidetoc}
\LWR@htmlelementclass{div}{bodycontainer}
\LWR@traceinfo{LWR@lwarpStart: Generating textbody.}
\LWR@htmlelementclass{section}{textbody}
```

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.

```
\LWR@patchlists
```

Ensure that math mode is active to call lwarp's patches:

```
\catcode`\$=\active
```

Required for \nameref to work with SVG math:

```
\immediate\write\@mainaux{\catcode‘\string$\active}%
\LetLtxMacro\LWR@syntaxhighlightone$% balance for editor syntax highlighting
```

Allow HTML paragraphs to begin:

```
\LWR@startpars
```

If using MATHJAX, disable \ensuremath by printing a nullified definition at the start of each file, and add further customizations:

```
\LWR@customizeMathJax
```

First autopage label in case a figure occurs early.

```
\setcounter{LWR@latestautopage}{\value{page}}%
\LWR@newautopagelabel{LWR@currentautosec}%

\LWR@traceinfo{LWR@lwarpStart: done}
}
\catcode‘\$=3% math shift until lwarp starts

\end{warpHTML}
```

64 Ending HTML output

for HTML output:

```
\begin{warpHTML}
```

\LWR@requesttoc {⟨*boolean*⟩} {⟨*suffix*⟩} Requests that a TOC, LOF, or LOTbe generated.

```
\newcommand*{\LWR@requesttoc}[2]{%
\ifbool{#1}
{
    \expandafter\newwrite\@nameuse{tf@#2}
    \immediate\openout \@nameuse{tf@#2} \jobname.#2\relax
}{}
}
```

\LWR@LwarpEnd Final stop of all HTML output:

```
\newcommand*{\LWR@LwarpEnd}
{
\LWR@stoppars
\LWR@closeprevious{\LWR@depthfinished}
```

At the bottom of the ending file:

Close the textbody:

```
\LWR@htmlelementclassend{section}{textbody}
\LWR@htmlelementclassend{div}{bodycontainer}
\LWR@htmlelementclassend{div}{bodyandsidetoc}
```

Print any pending footnotes:

```
\LWR@printpendingfootnotes
```

Create the footer:

```
\LWR@htmlelement{footer}

\LWR@pagebottom

\LWR@htmlelementend{footer}
```

No bottom navigation if are finishing the home page, or if formatting for an EPUB or word processor.

Presumably has a table-of-contents.

```
\ifthenelse{\boolean{FormatEPUB}\OR\boolean{FormatWP}}
    {}
    {
        \ifnumcomp{\value{LWR@htmlfilenumber}}{>}{0}{\LWR@botnavigation}{}
    }

\LWR@stoppars% final stop of all paragraphs
```

Finish the HTML file:

```
\LWR@htmltag{/body}\LWR@orignewline
\LWR@htmltag{/html}\LWR@orignewline
```

Seems to be required sometimes:

```
\LWR@orignewpage
```

For `lateximage` commands:

```
\immediate\closeout\LWR@lateximagesfile
}

\end{warpHTML}
```

65 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 65.8.

affiliation

lwarp provides for the \author macro an additional \affiliation macro to provide an affiliation and other additional information for each author in the title page. The affiliation information is removed when using titlingpage's \theauthor in the main text.

reusing titlepage information

The titling package maintains the definitions of \thetitle, \theauthor, etc., after the title has been typeset. These commands are to be used to refer to the document's title and author, etc., in the main text. These definitions have the \thanks and \affiliation removed, and for \author the \and is replaced to generate a simple

⚠ \theauthor, authblk

inline list of authors separated by commas. Note: \theauthor does not 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.

65.1 Setting the title, etc.

The following provide setting commands for both HTML and print outputs.

\author
\and

{⟨*author*⟩} While using \maketitle and print mode, the author is treated as a single-column tabular and the \and feature finishes the current tabular then starts a new one for the next author. Each author thus is placed into its own tabular, and an affiliation may be placed on its own line such as

```
\author{Name \\ Affiliation \and Second Name \\ Second Affiliation}
```

For HTML, the entire author block is placed inside a of class author, and each individual author is inside a of class oneauthor.

\@title
\@author
\@date

\@title, \@author, etc. store the values as originally assigned, including any \thanks, \and, or \affiliation. These are low-level macros intended to be used by other macros only inside a titlepage or titlingpage, and are used by \maketitle. The author is printed inside a single-column tabular, which becomes multiple single-column tabulars if multiples authors are included. For HTML, these tabulars become side-by-side s of class oneauthor, all of which are combined into one of class author.

\printtitle
\printauthor
\printdate

\printtitle, etc. are user-level macros intended to be used in custom titlepage or titlingpage environments in cases where \maketitle is not desired. These commands preserve the \thanks, etc., and should not be used in the main text.

\thetitle
\theauthor
\thedate

\thetitle, \theauthor, and \thedate are available if titling has been loaded, and are sanitized user-level versions from which have been removed the \thanks and \affiliation, and \and is changed for inline text usage. The author is printed inline without \affiliation or \thanks, with \and placing commas between multiple authors. Thus, these commands are to be used in the main text whenever the user wishes to refer to the document's title and such. One practical use for this is to place the authors at the bottom of each HTML page, such as:

\HTMLPageBottom {⟨*text*⟩}

```
\HTMLPageBottom{
\begin{center}\textcopyright~2016 \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.

65.2 \if@titlepage

for HTML & PRINT:

```
\begin{warpall}
```

\if@titlepage

Some classes do not provide \if@titlepage. In this case, provide it and force it false.

```
\ifcsvoid{@titlepagefalse}{
    \newif\if@titlepage
    \@titlepagefalse
}{}

\end{warpall}
```

65.3 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:
```
\begin{warpall}
\providerobustcmd{\affiliation}[1]{}
\end{warpall}
```

for PRINT output:
```
\begin{warpprint}

\AtBeginEnvironment{titlepage}{
\renewrobustcmd{\affiliation}[1]{\\ \textsc{\small#1}}
}

\AtBeginDocument{
\@ifpackageloaded{titling}{
\AtBeginEnvironment{titlingpage}{
\renewrobustcmd{\affiliation}[1]{\\ \textsc{\small#1}}
}
}{}% titling loaded
}% AtBeginDocument

\end{warpprint}
```

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

Env `titlepage` Sets up a of class `titlepage`. Provided even for memoir class, since it is used by `\maketitle`.

```
\DeclareDocumentEnvironment{titlepage}{}
{
\renewrobustcmd{\affiliation}[1]{\\ \InlineClass{affiliation}{##1}}
\LWR@printpendingfootnotes
\LWR@forcenewpage
\BlockClass{titlepage}
}
{
\endBlockClass
\LWR@printpendingfootnotes
}

\end{warpHTML}
```

65.4 Printing the thanks

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

`\printthanks` Forces the `\thanks` to be printed.

This is necessary in a `titlingpage` environment when `\maketitle` was not used.

```
\newcommand*{\printthanks}{\@thanks}

\end{warpall}
```

65.5 Printing the title, etc. in HTML

The following are for printing the title, etc. in a `titlepage` or a `titlingpage` in HTML:

for HTML output:

```
\begin{warpHTML}
```

`\printtitle`

```
\newcommand*{\printtitle}
{%
    \LWR@stoppars%
    \LWR@htmltag{\LWR@tagtitle}%
    \@title%
    \LWR@htmltag{\LWR@tagtitleend}%
    \LWR@startpars%
}
```

`\LWR@printthetitle` A private version which prints the title without footnotes, used to title each HTML page.

```
\newcommand*{\LWR@printthetitle}
{%
    \LWR@stoppars%
    \LWR@htmltag{\LWR@tagtitle}%
    \thetitle%
    \LWR@htmltag{\LWR@tagtitleend}%
    \LWR@startpars%
}
```

`\printauthor` HTML version.

```
\newcommand*{\printauthor}{
```

The entire author block is contained in a `<div>` named `author`:

```
\begin{BlockClass}{author}
```

`\and` finishes one author and starts the next:

```
\renewcommand{\and}{%
\end{BlockClass}
\begin{BlockClass}{oneauthor}
}
```

Individual authors are contained in a `<div>` named `oneauthor`:

```
\begin{BlockClass}{oneauthor}
\@author
\end{BlockClass}
\end{BlockClass}
}
```

\printdate

```
\newcommand*{\printdate}{%
\begin{BlockClass}{titledate}
\@date
\end{BlockClass}
}
```

```
\end{warpHTML}
```

65.6 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:

```
\begin{warpprint}
```

\printtitle

```
\newcommand*{\printtitle}{{\Huge\@title}}
```

\printauthor Print mode.

```
\newcommand*{\printauthor}
    {{\large\begin{tabular}[t]{c}\@author\end{tabular}}}
```

\printdate

```
\newcommand*{\printdate}{{\small\textit{\@date}}}
```

```
\end{warpprint}
```

65.7 \maketitle for HTML output

An HTML of class titlepage is used.

\thanks are a form of footnotes used in the title page. See section 58 for other kinds of footnotes.

See \thanksmarkseries{series}, below, to set the style of the footnote marks.

for HTML output:

```
\begin{warpHTML}
```

```
\@ifclassloaded{memoir}
{
\newcommand{\LWR@setfootnoteseries}{%
    \renewcommand\thefootnote{\@arabic\c@footnote}%
```

```
}
}{% not memoir
\if@titlepage
\newcommand{\LWR@setfootnoteseries}{%
    \renewcommand\thefootnote{\@arabic\c@footnote}%
}
\else
\newcommand{\LWR@setfootnoteseries}{%
    \renewcommand\thefootnote{\@fnsymbol\c@footnote}%
}
\fi
}% not memoir
```

`\LWR@maketitlesetup` Patches `\thanks` macros.

```
\newcommand*{\LWR@maketitlesetup}{%
```

Redefine the footnote mark:

```
\LWR@setfootnoteseries%
\def\@makefnmark{\textsuperscript{\thefootnote}}
```

`\thefootnote` ⇒ `\nameuse{arabic}{footnote}`, or
`\thefootnote` ⇒ `\nameuse{fnsymbol}{footnote}`

Redefine the footnote text:

```
\long\def\@makefntext##1{%
```

Make the footnote mark and some extra horizontal space for the tags:

```
\textsuperscript{\@thefnmark}~%
```

`\makethanksmark` ⇒ `\thanksfootmark` ⇒ `\tamark` ⇒ `\@thefnmark` ⇒ `\itshape` a (or similar)

Print the text:

```
##1%
}%
}
```

`\@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*.

```
\def\@fnsymbol#1{\ifcase#1\or *\or \HTMLentity{dagger}\or \HTMLentity{Dagger}\or
   \HTMLentity{sect}\or \HTMLentity{para}\or \text{\HTMLunicode{2016}}\or
   **\or \HTMLentity{dagger}\HTMLentity{dagger} \or
   \HTMLentity{Dagger}\HTMLentity{Dagger} \else\@ctrerr\fi}
```

\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.

```
\renewcommand*{\maketitle}{%
```

An HTML titlepage is used for all classes.

```
\begin{titlepage}
```

Set up special patches:

```
\LWR@maketitlesetup
```

Typeset the title, etc:

```
\@maketitle
```

Immediately generate any \thanks footnotes:

```
\@thanks
```

Close the HTML titlepage div and cleanup:

```
\end{titlepage}
\setcounter{footnote}{0}%
\global\let\thanks\relax
\global\let\maketitle\relax
\global\let\@maketitle\relax
\global\let\@thanks\@empty
\global\let\@author\@empty
\global\let\@date\@empty
\global\let\@title\@empty
\global\let\title\relax
\global\let\author\relax
\global\let\date\relax
\global\let\and\relax
}
```

\@maketitle HTML mode. Typesets the title, etc.:

```
\DeclareDocumentCommand{\@maketitle}{}{%
    \LWR@stoppars\LWR@htmltag{\LWR@tagtitle}%
    \@title%
    \LWR@htmltag{\LWR@tagtitleend}\LWR@startpars%
    \begin{BlockClass}{author}%
```

For IEEEtran class:

```
    \renewcommand*{\cr}{}%
    \renewcommand*{\crcr}{}%
    \renewcommand*{\noalign}{}%
```

```
        \renewcommand{\and}{%
            \end{BlockClass}%
            \begin{BlockClass}{oneauthor}%
        }%
        \begin{BlockClass}{oneauthor}%
            \@author%
        \end{BlockClass}%
    \end{BlockClass}%
    \begin{BlockClass}{titledate}%
    \@date%
    \end{BlockClass}%
}
```

\LWR@titlingmaketitle `\maketitle` for use inside an HTML `titlingpage` environment.

```
\newcommand*{\LWR@titlingmaketitle}{%
```

Keep pending footnotes out of the title block:

```
\@thanks
```

Set up special patches:

```
\LWR@maketitlesetup
```

Typeset the title, etc:

```
\@maketitle
```

Immediately generate any `\thanks` footnotes:

```
\@thanks
}
```

```
\end{warpHTML}
```

65.8 \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:

```
\begin{warpall}
```

`\AddSubtitlePublished` Adds `\published` and `\subtitle`, and related.

```
\newcommand*{\AddSubtitlePublished}{%
\@ifpackageloaded{titling}{% yes titling package
    \newcommand{\@published}{}%
    \newcommand{\published}[1]{\gdef\@published{##1}}%
    \renewcommand*{\maketitlehooka}{\printpublished}%
    \newcommand*{\printpublished}{%
        \warpprintonly{\begin{center}\unskip}%
        \begin{BlockClass}{published}%
        \warpprintonly{\large\itshape}%
        \@published%
        \end{BlockClass}%
        \warpprintonly{\end{center}}%
    }%
    \newcommand{\@subtitle}{}%
    \newcommand{\subtitle}[1]{\gdef\@subtitle{##1}}%
    \renewcommand*{\maketitlehookb}{\printsubtitle}%
    \newcommand*{\printsubtitle}{%
        \warpprintonly{\begin{center}\unskip}%
        \begin{BlockClass}{subtitle}%
        \warpprintonly{\Large\itshape}%
        \@subtitle%
        \end{BlockClass}%
        \warpprintonly{\end{center}}%
    }%
}% yes titling package
{% no titling package
    \newcommand{\published}[1]{}%
    \newcommand*{\printpublished}{}%
    \newcommand{\subtitle}{}%
    \newcommand*{\printsubtitle}%
}% no titling package
}% \AddSubtitlePublished
```

```
\end{warpall}
```

66 Abstract

The following code replaces the LaTeX default, and will itself be replaced later if the abstract package is loaded.

for HTML output:

```
\begin{warpHTML}
```

`\abstractname` User-redefinable title for the abstract.

Also over-written by the babel package.

```
\providecommand*{\abstractname}{Abstract}
```

Some classes allow an optional name, so it is allowed here.

Env abstract

```
\DeclareDocumentEnvironment{abstract}{O{\abstractname}}
{
\LWR@forcenewpage
\BlockClass{abstract}
\BlockClassSingle{abstracttitle}{#1}
}
{
\endBlockClass
}

\end{warpHTML}
```

67 Quote and verse

67.1 Attributions

\attribution For use with quote, quotation, verse:

Ex: `"A quotation." \attribution{\textsc{Author Name}\\\textsl{Book Title}}`

for HTML output:

```
\begin{warpHTML}
\newcommand{\attribution}[1]{%
    \begin{BlockClass}{attribution}
    #1
    \end{BlockClass}
}
\end{warpHTML}
```

for PRINT output:

```
\begin{warpprint}
\newcommand{\attribution}[1]{
    \begin{flushright}
    \unskip
    #1
    \end{flushright}%
}
\end{warpprint}
```

67.2 Quotes, quotations

for HTML output:

```
\begin{warpHTML}
```

Env quote

```
\renewenvironment*{quote}
{
\LWR@forcenewpage
\LWR@htmlblocktag{blockquote}
}
{\LWR@htmlblocktag{/blockquote}}

\renewenvironment*{quotation}
{
\LWR@forcenewpage
\LWR@htmlblocktag{blockquotation}
}
{\LWR@htmlblocktag{/blockquotation}}

\end{warpHTML}
```

67.3 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}
```

Len \vleftskip
Len \vleftmargini
Len \HTMLvleftskip
Len \HTMLleftmargini

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 and \HTMLleftmargini are provided to control the margins in HTML output. These 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 case a label may be shifted by one space.

67.3.1 LaTeX core verse environment

for HTML output:

```
\begin{warpHTML}
```

Env verse

```
\renewenvironment{verse}
               {\let\\\newline% lwarp
                \list{}{\itemsep      \z@
                        \itemindent   -1.5em%
                        \listparindent\itemindent
                        \rightmargin  \leftmargin
                        \advance\leftmargin 1.5em}%
                \item\relax}
               {\endlist}
```

```
\end{warpHTML}
```

for HTML & PRINT:

```
\begin{warpall}
```

67.3.2 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.

Len \HTMLvleftskip Sets \vleftskip inside a verse environment in HTML.

```
\newlength{\HTMLvleftskip}
\setlength{\HTMLvleftskip}{1em}
```

Len \HTMLleftmargini Sets \leftmargini inside a verse environment in HTML.

```
\newlength{\HTMLleftmargini}
\setlength{\HTMLleftmargini}{4.5em}
```

```
\end{warpall}
```

68 Verbatim and tabbing

for HTML & PRINT:

```
\begin{warpall}
```

Len \VerbatimHTMLWidth Width to use in HTML Verbatim environment.

This width is used when placing line numbers to the right. Ignored during print output.

```
\newlength{\VerbatimHTMLWidth}
\setlength{\VerbatimHTMLWidth}{4in}
\end{warpall}
```

for HTML output:

```
\begin{warpHTML}
```

Bool LWR@verbtags Used to temporarily turn off verbatim tags while doing \verbatiminput in the HTML head.

```
\newbool{LWR@verbtags}
\booltrue{LWR@verbtags}
```

`\LWR@atbeginverbatim` `[`⟨*1: style*⟩`]` `{`⟨*2: negative* `\baselineskip` `\vspace`⟩`}` `{`⟨*3: class*⟩`}`

Encloses a verbatim environment with the given css class.

The use of `\textquotedbl` instead of " improves compatibility with xeCJK.

```
\newcommand*{\LWR@atbeginverbatim}[3][]
{%
```

Avoid excessive space between lines:

```
\setlength{\parskip}{0ex}%
```

Stop generating HTML paragraph tags:

```
\LWR@stoppars%
```

Create a new `pre` of the given class. The tags may temporarily be turned off for internal use, such as loading the MathJax script.

```
\ifbool{LWR@verbtags}{%
    \LWR@htmltag{pre class=\textquotedbl#3\textquotedbl%
        \ifthenelse{\equal{#1}{}}{}{ style=\textquotedbl#1\textquotedbl}%
    }%
    \LWR@orignewline% pre
    \leavevmode\unskip\LWR@print@vspace*{-#2\baselineskip}%
}{}%
```

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.

```
\begingroup%
```

```
% \LWR@print@normalsize%
\LWR@origttfamily%
\LWR@print@small%
```

Since inside a <pre>, restore the original list processing:

```
\LWR@restoreoriglists%
```

Turn off babel-french extra space before punctuation:

```
\LWR@FBcancel%
```

Do not produce HTML tags for `\hspace` inside a verse `par`. Restore plain LaTeX `\hspace` functionality:

```
\LWR@select@print@hspace%
}
```

`\LWR@afterendverbatim` {⟨*negative* `\baselineskip` `\vspace`⟩}

Finishes enclosing a verbatim environment.

```
\newcommand*{\LWR@afterendverbatim}[1]{%
\endgroup%
\par%
```

At the end of the environment, close the `pre`:

```
\ifbool{LWR@verbtags}{%
    \LWR@print@vspace*{-#1\baselineskip}%
    \noindent\LWR@htmltag{/pre}\LWR@orignewline% pre
}{}%
```

Resume regular paragraph handling:

```
\LWR@startpars%
}
```

`\verbatiminput` {⟨*filename*⟩}

Patch `\verbatiminput` to add HTML tags:

```
\let\LWRV@origverbatim@input\verbatim@input

\renewcommand{\verbatim@input}[2]{%
\ifbool{LWR@verbtags}{\LWR@forcenewpage}{}%
\LWR@atbeginverbatim{2.5}{Verbatim}%
\LWRV@origverbatim@input{#1}{#2}%
\LWR@afterendverbatim{1.5}%
}
```

Env `verbatim`

```
\AfterEndPreamble{
\LWR@traceinfo{Patching verbatim.}
\AtBeginEnvironment{verbatim}{%
\LWR@forcenewpage%
\LWR@atbeginverbatim{2.5}{verbatim}%
}
\AfterEndEnvironment{verbatim}{%
    \LWR@afterendverbatim{1}%
}
}
```

Env `tabbing` The `tabbing` environment works, except that SVG math and `latexmages` do not yet work inside the environment.

⚠ math in `tabbing` If math is used inside `tabbing`, place `tabbing` inside a `latexmage` environment, which will render the entire environment as a single SVG image.

```
\newcommand*{\LWR@HTML@tabbing}{%
\LWR@forcenewpage%
\LWR@atbeginverbatim{3}{tabbing}%
\LWR@print@tabbing%
}

\newcommand*{\LWR@HTML@endtabbing}{%
    \LWR@print@endtabbing%
    \LWR@afterendverbatim{1}%
}

\LWR@formatted{tabbing}
\LWR@formatted{endtabbing}

\end{warpHTML}
```

69 Theorems

\newtheorem {⟨*text*⟩} [⟨*counter*⟩] -or- [⟨*oldname*⟩] {⟨*text*⟩}

A few minor changes are made to supply HTML tags.

- The entire theorem is placed into a of class theoremcontents.
- The label for each theorem is placed inside a of class theoremlabel.
- The contents are placed inside a of class theoremcontents.

for HTML output:

```
\begin{warpHTML}
```

\@begintheorem {⟨*name*⟩} {⟨*number*⟩}

```
\renewcommand{\@begintheorem}[2]{%
\LWR@forcenewpage
\BlockClass{theoremcontents}
\trivlist
\item[\InlineClass{theoremlabel}{#1\ #2\ }]\itshape
}
```

\@opargbegintheorem {⟨*name*⟩} {⟨*number*⟩} {⟨*oparg*⟩}

```
\renewcommand{\@opargbegintheorem}[3]{%
\LWR@forcenewpage
\BlockClass{theoremcontents}
\trivlist
\item[\InlineClass{theoremlabel}{#1\ #2\ (#3)\ }]\itshape
}
```

\@endtheorem

```
\renewcommand*{\@endtheorem}{%
\endtrivlist
\endBlockClass% theoremcontents
}
```

```
\end{warpHTML}
```

70 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.

70.1 List environment

for HTML output:

```
\begin{warpHTML}
```

\LWR@printcloselist May be locally redefined by enumerate or description.

```
\newcommand*{\LWR@printcloselist}{\LWR@printcloseitemize}
```

\LWR@printopenlist May be locally redefined by enumerate or description.

```
\newcommand*{\LWR@printopenlist}{ul style="\LWR@print@mbox{list-style-type:none}"}
```

\@mklab Removes PDF spacing.

```
\AtBeginDocument{
\def\@mklab#1{%
%     \hfil %
    #1}
\let\makelabel\@mklab
}
```

\@donoparitem Modified for HTML output by replacing TEX boxes with plain text. Also removes PDF spacing.

```
\def\@donoparitem{%
  \@noparitemfalse
%   \global\setbox\@labels\hbox{\hskip -\leftmargin
%                                \unhbox\@labels
%                                 \hskip \leftmargin}%
%   \if@minipage\else
%     \@tempskipa\lastskip
%     \vskip -\lastskip
%     \advance\@tempskipa\@outerparskip
%     \advance\@tempskipa -\parskip
%     \vskip\@tempskipa
%   \fi
}
```

\@item Modified for HTML output by replacing TEX boxes with plain text. Also removes PDF spacing.

```
\def\LWR@HTML@item[#1]{%
\LWR@traceinfo{@item}
  \if@noparitem
    \@donoparitem
  \else
%     \if@inlabel
%       \indent
        \par
%     \fi
    \ifhmode
%       \unskip\unskip
        \par
    \fi
    \if@newlist
      \if@nobreak
        \@nbitem
      \else
%         \addpenalty\@beginparpenalty
%         \addvspace\@topsep
%         \addvspace{-\parskip}%
```

```
      \fi
    \else
%       \addpenalty\@itempenalty
%       \addvspace\itemsep
    \fi
    \global\@inlabeltrue
  \fi
%   \everypar{%
    \@minipagefalse
    \global\@newlistfalse

%     \if@inlabel
%       \global\@inlabelfalse

%       {\setbox\z@\lastbox
%        \ifvoid\z@
%          \kern-\itemindent
%        \fi}%

%       \box\@labels
%       \penalty\z@
%     \fi

%     \if@nobreak
%       \@nobreakfalse
%       \clubpenalty \@M
%     \else
%       \clubpenalty \@clubpenalty
%       \everypar{}%
%     \fi}%

  \if@noitemarg
    \@noitemargfalse
    \if@nmbrlist

      \refstepcounter\@listctr
    \fi
  \fi


    \makelabel{#1} % extra space
%   \sbox\@tempboxa{\makelabel{#1}%
%   \global\setbox\@labels\hbox{%
%     \unhbox\@labels
%     \hskip \itemindent
%     \hskip -\labelwidth
%     \hskip -\labelsep
%     \ifdim \wd\@tempboxa >\labelwidth
%       \box\@tempboxa

%     \else
%       \hbox to\labelwidth {\unhbox\@tempboxa}%
%     \fi
%     \hskip \labelsep}%
```

```
  \ignorespaces%
}
```

\@nbitem

```
\def\@nbitem{%
%   \@tempskipa\@outerparskip
%   \advance\@tempskipa -\parskip
%   \addvspace\@tempskipa
}
```

\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.

```
\newcommand*{\LWR@listitem}{%
\LWR@stoppars%
\LWR@startnewdepth{\LWR@depthlistitem}{\LWR@printcloselistitem}%
\LWR@htmltag{li}%
\LWR@startpars%
\LWR@origitem%
}
```

\LWR@nulllistfills Nullifies various TEX fill commands, in case they are used inside \makelabel. Problems are caused when these are nullified all the time.

```
\newcommand*{\LWR@nulllistfills}{%
\renewcommand*{\hss}{}%
\renewcommand*{\llap}[1]{##1}%
\renewcommand*{\rlap}[1]{##1}%
\renewcommand*{\hfil}{}%
\renewcommand*{\hfilneg}{}%
\renewcommand*{\hfill}{}%
}
```

Env list {⟨*label*⟩} {⟨*commands*⟩}

```
\newcommand*{\LWR@liststart}{%
\LWR@traceinfo{LWR@liststart}%
\LWR@stoppars%
\LWR@pushoneclose{\LWR@depthlist}{\LWR@printcloselist}%
\LWR@htmltag{\LWR@printopenlist}\LWR@orignewline%
\LWR@startpars%
\setlength{\topsep}{0pt}%
\setlength{\partopsep}{0pt}%
\setlength{\itemsep}{0pt}%
\setlength{\parsep}{0pt}%
\setlength{\leftmargin}{0pt}%
\setlength{\rightmargin}{0pt}%
```

```
\setlength{\listparindent}{0pt}%
\setlength{\itemindent}{0pt}%
\setlength{\labelsep}{1em}%
\LWR@nulllistfills%
}

\newcommand*{\LWR@listend}{%
\LWR@traceinfo{LWR@listend}%
\LWR@stoppars%
\LWR@closeprevious{\LWR@depthlist}%
\LWR@startpars%
}
```

70.2 Itemize

`\LWR@itemizeitem` [⟨*label*⟩]

Handles `\item` inside an itemize or enumerate.

See `\LWR@openparagraph` where extra `\hspace` is used to leave room for the label while inside a list during paragraph construction.

```
\newcommand*{\LWR@itemizeitem}{%
\LWR@stoppars%
\LWR@startnewdepth{\LWR@depthlistitem}{\LWR@printcloselistitem}%
\LWR@htmltag{li}%
\LWR@startpars%
\LWR@origitem%
}
```

Env `itemize` [⟨*options*⟩]

```
\newcommand*{\LWR@itemizestart}{%
\renewcommand*{\LWR@printcloselist}{\LWR@printcloseitemize}
\renewcommand*{\LWR@printopenlist}{ul style="\LWR@print@mbox{list-style-type:none}"}
\let\item\LWR@itemizeitem%
\LWR@nulllistfills%
}
```

70.3 Enumerate

An HTML unordered list is used with customized LaTeX-generated labels.

Env `enumerate` [⟨*options*⟩]

```
\newcommand*{\LWR@enumeratestart}{%
\renewcommand*{\LWR@printcloselist}{\LWR@printcloseitemize}
\renewcommand*{\LWR@printopenlist}{ul style="\LWR@print@mbox{list-style-type:none}"}
```

```
\let\item\LWR@itemizeitem%
\LWR@nulllistfills%
}
```

70.4 Description

`\LWR@descitem` `[`⟨*label*⟩`]` Handles an `\item` inside a description.

```
\newcommand*{\LWR@descitem}[1][]%
{%
\LWR@stoppars%
\LWR@setlatestname{#1}%
\LWR@startnewdepth{\LWR@depthlistitem}{\LWR@printclosedescitem}%
```

Temporarily disable `\hspace`, which `article.cls`, etc. use per `\item` for `descriptions` only. This causes lwarp to mistakenly place an empty span between HTML list tags.

```
\LWR@select@html@nohspace%
```

Process the original `\item` code:

```
\LWR@origitem[]%
```

Restore `\hspace` for use in the item text:

```
\LWR@select@html@hspace%

\LWR@htmltag{dt}#1\LWR@htmltag{/dt}%
\LWR@orignewline%
\LWR@htmltag{dd}%
\LWR@startpars%
}
```

Env `description` `[`⟨*options*⟩`]`

```
\newcommand*{\LWR@descriptionstart}{%
\renewcommand*{\LWR@printcloselist}{\LWR@printclosedescription}
\renewcommand*{\LWR@printopenlist}{dl}
\let\item\LWR@descitem%
\LWR@nulllistfills%
}
```

70.5 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.

```
\newcommand*{\LWR@patchlists}{%
    \LetLtxMacro\item\LWR@listitem%
    \LetLtxMacro\@item\LWR@HTML@item%
    \renewcommand*{\@trivlist}{%
        \LWR@traceinfo{@trivlist start}%
        \LWR@liststart%
        \LWR@orig@trivlist%
        \LWR@traceinfo{@trivlist done}%
    }%
    \renewcommand*{\trivlist}{%
        \LWR@traceinfo{trivlist}%
        \LWR@origtrivlist%
    }%
    \renewcommand*{\endtrivlist}{%
        \LWR@traceinfo{endtrivlist start}%
        \LWR@origendtrivlist\LWR@listend%
        \LWR@traceinfo{endtrivlist done}%
    }%
    \renewcommand*{\itemize}{%
        \LWR@itemizestart\LWR@origitemize%
    }%
    \renewcommand*{\enumerate}{%
        \LWR@enumeratestart\LWR@origenumerate%
    }%
    \renewcommand*{\description}{%
        \LWR@descriptionstart\LWR@origdescription%
    }%
}
```

\LWR@restoreoriglists Restores the original trivlist environment.

```
\newcommand*{\LWR@restoreoriglists}{%
\LWR@traceinfo{LWR@restoreoriglists}%
\LetLtxMacro\item\LWR@origitem%
\LetLtxMacro\@item\LWR@orig@item%
\let\@trivlist\LWR@orig@trivlist%
\let\trivlist\LWR@origtrivlist%
\let\endtrivlist\LWR@origendtrivlist%
\LetLtxMacro\itemize\LWR@origitemize%
\LetLtxMacro\enditemize\LWR@endorigitemize%
\LetLtxMacro\enumerate\LWR@origenumerate%
\LetLtxMacro\endenumerate\LWR@endorigenumerate%
\LetLtxMacro\description\LWR@origdescription%
\LetLtxMacro\enddescription\LWR@endorigdescription%
\let\@mklab\LWR@orig@mklab%
\let\makelabel\LWR@origmakelabel%
\let\@donoparitem\LWR@orig@donoparitem%
\let\@nbitem\LWR@orig@nbitem%
}

\end{warpHTML}
```

71 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.

71.1 Limitations

Tabular mostly works as expected, but pay special attention to the following, especially if working with environments, macros inside tabulars, multirows, * column specifiers, 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  % because & is used in a definition
\newenvironment{outerenvironment}
{
\tabular{cc}
left & right \\
}
{
\TabularMacro\ResumeTabular
left & right \\
\endtabular
}
\StopDefiningTabulars
```

Cell contents:

⚠ \multirow

- 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.

```
. . . &  \multirow{2}{.5in}{text}  & . . .
. . . &  \mrowcell                 & . . .
```

vposn

Note that recent versions of multirow include a new optional vposn argument.

- 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.
- See section 314.2 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:

⚠ empty cells

```
. . . &  \multicolumnrow{2}{c}[c]{3}[0]{1in}[0pt]{Text}  & . . .
. . . &  \mcolrowcell          &          \mcolrowcell   & . . .
. . . &  \mcolrowcell          &          \mcolrowcell   & . . .
```

vposn

Note that recent versions of multirow include a new optional vposn argument.

⚠ 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:

⚠ * column specification

- * in a column specification is not used (so far). Repeat the column type the correct number of times.

@ 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

- \newcolumntype is ignored; unknown column types are set to l.

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.

full-width rules

- \toprule, \midrule, \bottomrule, and \hline 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.

colortbl:

⚠ row/cell color

Only use `\rowcolor` and `\cellcolor` at the start of a row, in that order.

colortbl ignores the overhang arguments.

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 \\}
  ```

for HTML output:

```
\begin{warpHTML}
```

71.2 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.

71.2.1 arydshln

Emualated by the original LaTeX non-dashed versions.

```
\LetLtxMacro\hdashline\hline
\LetLtxMacro\cdashline\cline
\LetLtxMacro\firsthdashline\hline
\LetLtxMacro\lasthdashline\hline
```

71.3 Token lookahead

Used by \LWR@futurenonspacelet to look at the next token.

\LWR@mynexttoken

```
\newcommand\LWR@mynexttoken\relax
```

\futurelet copies the next token then executes a function to analyze

\LWR@futurenonspacelet does the same, but ignores intervening white space

Based on the booktabs style:

\LWR@futurenonspacelet

```
\def\LWR@futurenonspacelet#1{\def\LWR@cs{#1}%
\afterassignment\LWR@fnslone\let\nexttoken= }
\def\LWR@fnslone{\expandafter\futurelet\LWR@cs\LWR@fnsltwo}
\def\LWR@fnsltwo{%
\expandafter\ifx\LWR@cs\@sptoken\let\next=\LWR@fnslthree%
\else\let\next=\nexttoken\fi\next}
\def\LWR@fnslthree{\afterassignment\LWR@fnslone\let\next= }
```

\LWR@getmynexttoken Looks ahead and copies the next token into \LWR@mynexttoken.

```
\newcommand*{\LWR@getmynexttoken}{%
\LWR@traceinfo{LWR@getmynexttoken}%
% nothing must follow this next line
\LWR@futurenonspacelet\LWR@mynexttoken\LWR@tabledatacolumntag
}
```

71.4 Tabular variables

Bool LWR@startedrow True if should print a row tag before this column.

```
\newbool{LWR@startedrow}
\boolfalse{LWR@startedrow}
```

Bool LWR@tabularcelladded True if have added a data cell for this position.

```
\newbool{LWR@tabularcelladded}
\boolfalse{LWR@tabularcelladded}
```

Ctr LWR@hlines Number of \hlines or \midrules above the next row.

```
\newcounter{LWR@hlines}
```

Ctr LWR@hdashedlines Number of arydshln dashed lines above the next row.

```
\newcounter{LWR@hdashedlines}
```

Bool LWR@doingtbrule True if the next row will have a top/bottom rule above it.

```
\newbool{LWR@doingtbrule}
\boolfalse{LWR@doingtbrule}
```

Bool LWR@doingcmidrule 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`.

```
\newbool{LWR@doingcmidrule}
\boolfalse{LWR@doingcmidrule}
```

Bool LWR@tableparcell True if are handling a paragraph inside a table cell, so must close the paragraph tag before moving on.

```
\newbool{LWR@tableparcell}
```

Bool LWR@skippingmrowcell True if are doing an empty `\multirow` cell, and thus there is no data tag to close.

```
\newbool{LWR@skippingmrowcell}
```

Bool LWR@skippingmcolrowcell True if are doing an empty `\multicolumnrow` cell, and thus there is no data tag to close, and do not print `@` and `!` columns.

```
\newbool{LWR@skippingmcolrowcell}
```

Bool LWR@skipatbang True if just finished a `\multicolumn` so should not create the trailing `@` or `!` columns table data cells.

```
\newbool{LWR@skipatbang}
```

Bool LWR@emptyatbang True if finishing a row and should print empty `@` or `!` column table data cells.

```
\newbool{LWR@emptyatbang}
```

Bool LWR@intabularmetadata True if are in a tabular but not in a data cell. Used to prevent extra HTML breaks if not inside table data.

```
\newbool{LWR@intabularmetadata}
\boolfalse{LWR@intabularmetadata}
```

Ctr LWR@tabularDepth Tracks whether `&` is being used inside a `tabular`.

```
\newcounter{LWR@tabulardepth}
\setcounter{LWR@tabulardepth}{0}
```

Ctr LWR@tabularpardepth 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.

```
\newcounter{LWR@tabularpardepth}
\setcounter{LWR@tabularpardepth}{0}
```

```
\newcommand*{\LWR@colsresult}{}%temp storage for column format results
\newcommand*{\LWR@pposition}{}
\newcommand*{\LWR@pleft}{}
\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.

```
\providecommand*{\LWR@strresult}{}
\providecommand*{\LWR@strresulttwo}{}
```

\LWR@origcolspec — Holds the original column specs given to tabular.

```
\newcommand*{\LWR@origcolspec}{}
```

Ctr LWR@tablecolspecwidth — 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.)

```
\newcounter{LWR@tablecolspecwidth}
```

Ctr LWR@tablecolspecindex — While parsing the LaTeX table column specification, starts at 1 and is incremented per token of the specification.

```
\newcounter{LWR@tablecolspecindex}
```

Ctr LWR@tableLaTeXcolindex — 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.

```
\newcounter{LWR@tableLaTeXcolindex}
```

Ctr LWR@tabletotalLaTeXcols — 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.)

```
\newcounter{LWR@tabletotalLaTeXcols}
```

Ctr LWR@tabletotalLaTeXcolsnext — Holds the next LaTeX table column index while parsing, equal to one more than LWR@tabletotalLaTeXcols.

```
\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.

71.5 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

\LWR@instertatbangcols

```
\newcommand*{\LWR@insertatbangcols}{%
\ifbool{LWR@skipatbang}%
{}%
{%
    \LWR@printatbang{at}{\arabic{LWR@tableLaTeXcolindex}}%
    \LWR@printatbang{bang}{\arabic{LWR@tableLaTeXcolindex}}%
}%
}
```

\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.

```
\newcommand*{\LWR@closetabledatacell}{%
\LWR@stoppars%
\global\booltrue{LWR@intabularmetadata}%
\ifbool{LWR@exitingtabular}{}%
{% not exiting tabular
    \ifboolexpr{bool{LWR@skippingmrowcell} or bool{LWR@skippingmcolrowcell}}%
    {%
```

If not skipping a \multicolumnrow cell, insert the @ and ! columns after this non-existant column.

```
        \ifbool{LWR@skippingmcolrowcell}%
            {}%
            {\LWR@insertatbangcols}%
    }%
    {% not skippingmrowcell
```

Insert any < then any @ and ! column contents, unless muted for the \bottomrule or a \multicolumn:

```
        \unskip%
        \ifboolexpr{%
            bool{LWR@tabularmutemods} or
            bool{LWR@skipatbang} or
            bool{LWR@emptyatbang}
        }%
        {}%
        {\LWR@getexparray{LWR@colafterspec}{\arabic{LWR@tableLaTeXcolindex}}}%
```

Close paragraphs:

```
        \ifbool{LWR@tableparcell}{\LWR@stoppars}{}%
        \global\boolfalse{LWR@tableparcell}%
```

Close the table data cell.

Close any color s.

```
        \whileboolexpr{test {\ifnumcomp{\value{LWR@cellcolordepth}}{>}{0}}}{%
            \LWR@htmltag{/div}\LWR@orignewline%
            \addtocounter{LWR@cellcolordepth}{-1}%
        }%
```

Skip the @ and ! cells if are closing a multicolumn cell.

```
        \leavevmode\unskip\LWR@htmltag{/td}\LWR@orignewline%
        \global\booltrue{LWR@tabularcelladded}%
        \LWR@insertatbangcols%
    }% not skipping mrowcell
}% not exiting tabular
\global\boolfalse{LWR@skippingmrowcell}%
\global\boolfalse{LWR@skippingmcolrowcell}%
\global\boolfalse{LWR@skipatbang}%
```

Color control. Column is set by >{} for each cell, so it must be cleared here.

```
\gdef\LWR@cellHTMLcolor{}
\gdef\LWR@columnHTMLcolor{}
\setcounter{LWR@cellcolordepth}{0}
}
```

When not used inside a tabular, & performs its original function as recorded here (with catcode 4).

```
\let\LWR@origampmacro&
\end{warpHTML}
```

71.5.1 Handling &

for HTML output:

```
\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.

```
\newcommand*{\LWR@tabularampersand}{%
\LWR@traceinfo{LWR@tabularampersand}%
\ifnumcomp{\value{LWR@tabulardepth}}{>}{0}%
{%
```

If not skipping a multirow cell, close the current data cell.

```
    \unskip%
    \LWR@closetabledatacell%
```

Move to the next column.

```
    \addtocounter{LWR@tableLaTeXcolindex}{1}%
```

Have not yet added data in this column:

```
    \global\boolfalse{LWR@tabularcelladded}%
```

Look at the next token to decide multi or single column data tag.

```
    \LWR@getmynexttoken%
}%
```

If not inside a tabular, performs the original action:

```
{\LWR@origampmacro}%
}
```

& 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.

71.5.2 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.

```
\newcommand*{\LWR@tabularfinishrow}{%
```

If not exiting the tabular, or doing a rule, or have already started a row, finish this row:

```
\ifboolexpr{%
    not bool {LWR@exitingtabular} or%
    bool{LWR@doingtbrule} or%
    bool{LWR@doingcmidrule} or%
    test{\ifnumcomp{\value{LWR@hlines}}{>}{0}} or%
    test{\ifnumcomp{\value{LWR@hdashedlines}}{>}{0}} or%
    bool{LWR@startedrow}%
}{%
```

To 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:

```
\ifbool{LWR@exitingtabular}{%
    \global\booltrue{LWR@tabularmutemods}%
}{%
    \global\boolfalse{LWR@tabularmutemods}%
}%
```

Locally reenable the table data tags until finished with the final row:

```
\global\boolfalse{LWR@exitingtabular}%
```

Generate table data tags and ampersands until the right edge:

```
\whileboolexpr{%
    test {
      \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{<}{\value{LWR@tabletotalLaTeXcols}}
    } or %
    (%
        bool{LWR@intabularmetadata} and%
        not bool{LWR@tabularcelladded} and%
        test {
        \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{=}{\value{LWR@tabletotalLaTeXcols}}
        }%
    )%
}%
{%
    \LWR@tabledatasinglecolumntag%
```

The following is essentially \LWR@tabularampersand with LWR@emptyatbang added to empty the following cells:

```
    \LWR@closetabledatacell%
    \addtocounter{LWR@tableLaTeXcolindex}{1}%
    \global\boolfalse{LWR@tabularcelladded}%
    \global\booltrue{LWR@emptyatbang}%
```

Starts the next cell:

```
  \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{<}{\value{LWR@tabletotalLaTeXcols}}%
       {\LWR@getmynexttoken}%
       {}%
}%
```

Reenable the original LWR@exitingtabular to close the entire table:

```
\ifbool{LWR@tabularmutemods}{%
    \global\booltrue{LWR@exitingtabular}%
}{%
    \global\boolfalse{LWR@exitingtabular}%
}%
\global\boolfalse{LWR@tabularmutemods}%

\global\boolfalse{LWR@emptyatbang}%
}{}% ifboolexpr
}
```

71.6 Handling \\

Inside tabular, \\ is redefined to \LWR@tabularendofline

Throws away options \\[dim] or *

\LWR@tabularendofline

```
\NewDocumentCommand{\LWR@tabularendofline}{s o}{%
```

Finish the row:

```
\ifnumcomp{\value{LWR@tableLaTeXcolindex}}{<}{\value{LWR@tabletotalLaTeXcols}}%
    {\LWR@tabularfinishrow}%
    {\LWR@closetabledatacell}%
\LWR@htmltag{/tr}\LWR@orignewline%
```

xcolor row color support:

```
\@rowc@lors%
```

No longer inside a data cell:

```
\global\booltrue{LWR@intabularmetadata}%
```

Not yet started a table row:

```
\global\boolfalse{LWR@startedrow}%
```

Additional setup:

```
\setcounter{LWR@hlines}{0}%
\setcounter{LWR@hdashedlines}{0}%
\global\boolfalse{LWR@doingtbrule}%
\global\boolfalse{LWR@doingcmidrule}%
\LWR@clearmidrules%
\gdef\LWR@rowHTMLcolor{}%
```

Start at first column:

```
\setcounter{LWR@tableLaTeXcolindex}{1}%
```

Have not yet added data in this column:

```
\global\boolfalse{LWR@tabularcelladded}%
```

Allow TEX to flush the pending paragraph. Not doing so causes a slowdown for very large tables.

```
\LWR@stoppars
\LWR@origpar
```

Look at the next token to decide between single column data tag or a special case:

```
\LWR@getmynexttoken%
}
```

71.7 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}`.

```
\newcommand*{\LWR@columnspeclookahead}[1]{%
\setcounter{LWR@tempcountone}{\value{LWR@tablecolspecindex}}%
\addtocounter{LWR@tempcountone}{#1}%
\fullexpandarg%
\StrChar{\LWR@origcolspec}{\arabic{LWR@tempcountone}}[\LWR@strresulttwo]%
```

Get the contents of the first group in `\LWR@strresulttwo`:

```
\exploregroups%
```

```
\StrChar{\LWR@strresulttwo}{1}[\LWR@strresulttwo]%
\noexploregroups%
}
```

71.8 Parsing @, >, <, !, bar columns

Holds the parsed argument for `@`, `>`, `<`, or `!` columns:

```
\newcommand*{\LWR@colparameter}{}
```

`\LWR@parseatcolumn` Handles `@{text}` columns.

```
\newcommand*{\LWR@parseatcolumn}{%
```

Move to the next token after the '@':

```
\LWR@traceinfo{at column}%
\addtocounter{LWR@tablecolspecindex}{1}%
```

Read the next token into `\LWR@colparameter`, expanding once:

```
\LWR@traceinfo{about to read the next token:}%
\expandarg%
\StrChar{\LWR@origcolspec}%
    {\arabic{LWR@tablecolspecindex}}[\LWR@colparameter]
\fullexpandarg%
```

Store the result into a data array, expanding once out of `\LWR@colparameter`:

```
\LWR@traceinfo{have now read the next token}%
\ifnumcomp{\value{LWR@tabletotalLaTeXcols}}{=}{0}%
{% left edge of the table:
    \LWR@traceinfo{at the left edge}%
    \LWR@setexparray{LWR@colatspec}{leftedge}{\LWR@colparameter}%
    \LWR@traceinfo{at the left edge: %
    \LWR@getexparray{LWR@colatspec}{leftedge}}%
}%
{% not at the left edge:
    \LWR@traceinfo{not at the left edge}%
    \LWR@setexparray{LWR@colatspec}%
        {\arabic{LWR@tabletotalLaTeXcols}}{\LWR@colparameter}%
    \LWR@traceinfo{at \arabic{LWR@tabletotalLaTeXcols}: %
    \LWR@getexparray{LWR@colatspec}{\arabic{LWR@tabletotalLaTeXcols}}}%
}%
\let\LWR@colparameter\relax%
\global\booltrue{LWR@validtablecol}%
}
```

`\LWR@parsebangcolumn` Handles `!{text}` columns.

```
\newcommand*{\LWR@parsebangcolumn}{%
```

Move to the next token after the '!':

```
\LWR@traceinfo{bang column}%
\addtocounter{LWR@tablecolspecindex}{1}%
```

Read the next token into \LWR@colparameter, expanding once:

```
\LWR@traceinfo{about to read the next token:}%
\expandarg%
\StrChar{\LWR@origcolspec}%
    {\arabic{LWR@tablecolspecindex}}[\LWR@colparameter]
\fullexpandarg%
```

Store the result into a data array, expanding once out of \LWR@colparameter:

```
\LWR@traceinfo{have now read the next token}%
\ifnumcomp{\value{LWR@tabletotalLaTeXcols}}{=}{0}%
{% left edge of the table:
    \LWR@traceinfo{at the left edge}%
    \LWR@setexparray{LWR@colbangspec}{leftedge}{\LWR@colparameter}%
}%
{% not at the left edge:
    \LWR@traceinfo{not at the left edge}%
    \LWR@setexparray{LWR@colbangspec}%
        {\arabic{LWR@tabletotalLaTeXcols}}{\LWR@colparameter}%
    \LWR@traceinfo{bang \arabic{LWR@tabletotalLaTeXcols}: \LWR@colparameter!}%
}%
\let\LWR@colparameter\relax%
\global\booltrue{LWR@validtablecol}%
}
```

\LWR@parsebeforecolumn Handles >{text} columns.

```
\newcommand*{\LWR@parsebeforecolumn}{%
```

Move to the next token after the '>':

```
\addtocounter{LWR@tablecolspecindex}{1}%
```

Read the next token, expanding once into \LWR@colparameter:

```
\expandarg%
\StrChar{\LWR@origcolspec}%
    {\arabic{LWR@tablecolspecindex}}[\LWR@colparameter]%
\fullexpandarg%
```

Store the result into a data array, expanding once out of \LWR@colparameter:

```
\LWR@setexparray{LWR@colbeforespec}%
    {\arabic{LWR@tabletotalLaTeXcolsnext}}{\LWR@colparameter}%
\let\LWR@colparameter\relax%
\global\booltrue{LWR@validtablecol}%
}
```

\LWR@parseaftercolumn Handles <{text} columns.

```
\newcommand*{\LWR@parseaftercolumn}{%
```

Move to the next token after the '<':

```
\addtocounter{LWR@tablecolspecindex}{1}%
```

Read the next token, expanding once into \LWR@colparameter:

```
\expandarg%
\StrChar{\LWR@origcolspec}%
    {\arabic{LWR@tablecolspecindex}}[\LWR@colparameter]%
\fullexpandarg%
```

Store the result into a data array, expanding once out of \LWR@colparameter:

```
\LWR@setexparray{LWR@colafterspec}%
    {\arabic{LWR@tabletotalLaTeXcols}}{\LWR@colparameter}%
\let\LWR@colparameter\relax%
\global\booltrue{LWR@validtablecol}%
}
```

\LWR@parsebarcolumn Handles vertical rules.

```
\newcommand*{\LWR@parsebarcolumn}{%
\LWR@traceinfo{LWR@parsebarcolumn}%
```

Remember the bar at this position:

```
\ifnumcomp{\value{LWR@tabletotalLaTeXcols}}{=}{0}%
{% left edge of the table:
    \edef\LWR@tempone{\LWR@getexparray{LWR@colbarspec}{leftedge}}%
    \ifdefstring{\LWR@tempone}{tvertbarl}%
    {\LWR@setexparray{LWR@colbarspec}{leftedge}{tvertbarldouble}}%
    {\LWR@setexparray{LWR@colbarspec}{leftedge}{tvertbarl}}%
}%
{% not at the left edge:
    \edef\LWR@tempone{%
        \LWR@getexparray{LWR@colbarspec}{\arabic{LWR@tabletotalLaTeXcols}}%
    }%
    \ifdefstring{\LWR@tempone}{tvertbarr}%
    {%
        \LWR@setexparray{LWR@colbarspec}%
            {\arabic{LWR@tabletotalLaTeXcols}}{tvertbarrdouble}%
    }%
    {%
        \LWR@setexparray{LWR@colbarspec}%
            {\arabic{LWR@tabletotalLaTeXcols}}{tvertbarr}%
    }%
}%
\global\booltrue{LWR@validtablecol}%
}
```

\LWR@parsecoloncolumn Handles vertical rules.

```
\newcommand*{\LWR@parsecoloncolumn}{%
\LWR@traceinfo{LWR@parsecoloncolumn}%
```

Remember the bar at this position:

```
\ifnumcomp{\value{LWR@tabletotalLaTeXcols}}{=}{0}%
{% left edge of the table:
    \edef\LWR@tempone{\LWR@getexparray{LWR@colbarspec}{leftedge}}%
    \ifdefstring{\LWR@tempone}{tvertbarldash}%
    {\LWR@setexparray{LWR@colbarspec}{leftedge}{tvertbarldoubledash}}%
    {\LWR@setexparray{LWR@colbarspec}{leftedge}{tvertbarldash}}%
}%
{% not at the left edge:
    \edef\LWR@tempone{%
        \LWR@getexparray{LWR@colbarspec}{\arabic{LWR@tabletotalLaTeXcols}}%
    }%
    \ifdefstring{\LWR@tempone}{tvertbarrdash}%
    {\LWR@setexparray{LWR@colbarspec}%
        {\arabic{LWR@tabletotalLaTeXcols}}{tvertbarrdoubledash}}%
    {\LWR@setexparray{LWR@colbarspec}%
        {\arabic{LWR@tabletotalLaTeXcols}}{tvertbarrdash}}%
}%
\global\booltrue{LWR@validtablecol}%
}
```

\LWR@parsesemicoloncolumn Handles vertical rules.

```
\newcommand*{\LWR@parsesemicoloncolumn}{%
```

Treat ; as a : column:

```
\LWR@parsecoloncolumn%
```

Skip the following width token:

```
\addtocounter{LWR@tablecolspecindex}{1}%
}
```

71.9 Parsing 'l', 'c', or 'r' columns

\LWR@parsenormalcolumn {⟨*thiscolumn*⟩}

Add to the accumulated column specs, advance counters, and pre-clear another column of at, before, and after specs.

```
\newcommand*{\LWR@parsenormalcolumn}[1]{%
\addtocounter{LWR@tabletotalLaTeXcols}{1}%
\addtocounter{LWR@tabletotalLaTeXcolsnext}{1}%
```

```
\LWR@setexparray{LWR@tablecolspec}{\arabic{LWR@tabletotalLaTeXcols}}{#1}%

\LWR@traceinfo{normal column \arabic{LWR@tabletotalLaTeXcols}: #1}%
\LWR@setexparray{LWR@colatspec}{\arabic{LWR@tabletotalLaTeXcolsnext}}{}%
\LWR@setexparray{LWR@colbangspec}{\arabic{LWR@tabletotalLaTeXcolsnext}}{}%
\LWR@setexparray{LWR@colbeforespec}{\arabic{LWR@tabletotalLaTeXcolsnext}}{}%
\LWR@setexparray{LWR@colafterspec}{\arabic{LWR@tabletotalLaTeXcolsnext}}{}%
\LWR@setexparray{LWR@colbarspec}{\arabic{LWR@tabletotalLaTeXcolsnext}}{}%
\global\booltrue{LWR@validtablecol}%
}
```

71.10 Parsing 'p', 'm', or 'b' columns

`\LWR@parsepcolumn` {⟨*thiscolumn*⟩} The width will be ignored.

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

Converts to the given column type:

```
\LWR@parsenormalcolumn{#1}%
```

Skips the following width token:

```
\addtocounter{LWR@tablecolspecindex}{1}%
}
```

71.11 Parsing 'w' columns

`\LWR@parsewcolumn` The width will be ignored.

```
\newcommand*{\LWR@parsewcolumn}{%

\LWR@columnspeclookahead{1}%
\expandafter\LWR@parsenormalcolumn\expandafter{\LWR@strresulttwo}%
```

Skips the following width and alignment tokens:

```
\addtocounter{LWR@tablecolspecindex}{2}%
}
```

71.12 Parsing 'D' columns

From the dcolumn package.

Table 10: Tabular baseline

l	p	m	b	r
			bot	
		mid	bot	
l	par	mid	bot	r
	par	mid		
	par			

\LWR@parseDcolumn {⟨*thiscolumn*⟩} The three parameters will be ignored.

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

Converts to the given column type.

```
\LWR@parsenormalcolumn{#1}%
```

Skips the following three parameters.

```
\addtocounter{LWR@tablecolspecindex}{3}%
}
```

71.13 Parsing the column specifications

⚠ tabular baselines HTML CSS cannot exactly match the LaTeX concept of a baseline for a table row. Table 10 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 11 for details.

Table 11 describes how each kind of column is converted to HTML.

Bool LWR@validtablecol True if found a valid table column type.

```
\newbool{LWR@validtablecol}
```

Bool LWR@opttablecol True if found a table column optional argument.

```
\newbool{LWR@opttablecol}
```

\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,

Table 11: Tabular HTML column conversions

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 10, 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.

P, M, B: Horizontally-centered versions.

S: Converted to 'r'. Ignores optional argument. From the siunitx package.

D: Converted to 'c'. From the dcolumn package.

@, !, >, <: One each, in that order.

|: Vertical rule.

Unknown: Converted to 'l'.

`\newcolumn`: Currently treated as unknown.

which is the number of & and \\ in each line, but which does not include @, !, <, > specifications in the count.

```
\newcommand*{\LWR@parsetablecols}[1]{%
\LWR@traceinfo{LWR@parsetablecols}%
```

Remember the original supplied column spec:

```
\renewcommand*{\LWR@origcolspec}{#1}%
```

Remove spaces:

```
\expandarg%
\StrSubstitute{\LWR@origcolspec}{ }{}[\LWR@origcolspec]%
```

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:

```
\setcounter{LWR@tabletotalLaTeXcols}{0}%
\setcounter{LWR@tabletotalLaTeXcolsnext}{1}%
\LWR@setexparray{LWR@colatspec}{leftedge}{}%
\LWR@setexparray{LWR@colatspec}{1}{}%
\LWR@setexparray{LWR@colatspec}{2}{}%
\LWR@setexparray{LWR@colatspec}{3}{}%
\LWR@setexparray{LWR@colbangspec}{leftedge}{}%
\LWR@setexparray{LWR@colbangspec}{1}{}%
\LWR@setexparray{LWR@colbangspec}{2}{}%
\LWR@setexparray{LWR@colbangspec}{3}{}%
\LWR@setexparray{LWR@colbeforespec}{1}{}%
\LWR@setexparray{LWR@colbeforespec}{2}{}%
\LWR@setexparray{LWR@colbeforespec}{3}{}%
\LWR@setexparray{LWR@colafterspec}{1}{}%
\LWR@setexparray{LWR@colafterspec}{2}{}%
\LWR@setexparray{LWR@colafterspec}{3}{}%
\LWR@setexparray{LWR@colbarspec}{leftedge}{}%
\LWR@setexparray{LWR@colbarspec}{1}{}%
\LWR@setexparray{LWR@colbarspec}{2}{}%
\LWR@setexparray{LWR@colbarspec}{3}{}%
```

Starting at the first column specification:

```
\setcounter{LWR@tablecolspecindex}{1}%
```

Place the colspecs string length into \LWR@strresult, and remember the number of characters in the column specification:

```
\expandarg%
\StrLen{\LWR@origcolspec}[\LWR@strresult]%
\fullexpandarg%
\LWR@traceinfo{original column spec length: \LWR@strresult}%
\setcounter{LWR@tablecolspecwidth}{\LWR@strresult}%
```

Haven't seen any optional arguments so far

```
\global\boolfalse{LWR@opttablecol}%
```

Scan through the column specifications:

```
\whileboolexpr{%
    not test{%
        \ifnumcomp{\value{LWR@tablecolspecindex}}{>}%
            {\value{LWR@tablecolspecwidth}}%
    }%
}%
{%
```

Place the next single-character column type into `\LWR@strresult`:

```
\expandarg%
\StrChar{\LWR@origcolspec}{\arabic{LWR@tablecolspecindex}}[\LWR@strresult]%
\LWR@traceinfo{position \arabic{LWR@tablecolspecindex}: \LWR@strresult}%
\fullexpandarg%
```

Not yet found a valid column type:

```
\global\boolfalse{LWR@validtablecol}%
```

Skip over any optional arguments, such as siunitx `S` column:

```
\IfStrEq{\LWR@strresult}{[}{\global\booltrue{LWR@opttablecol}}{}%
```

Throw away anything found inside the optional argument:

```
\ifbool{LWR@opttablecol}%
{}% inside an optional argument
{% not an optional tabular argument
```

Not inside an optional argument, so consider the column type:

```
\IfStrEq{\LWR@strresult}{l}{\LWR@parsenormalcolumn{l}}{}%
\IfStrEq{\LWR@strresult}{c}{\LWR@parsenormalcolumn{c}}{}%
\IfStrEq{\LWR@strresult}{r}{\LWR@parsenormalcolumn{r}}{}%
\IfStrEq{\LWR@strresult}{L}{\LWR@parsenormalcolumn{l}}{}%
\IfStrEq{\LWR@strresult}{C}{\LWR@parsenormalcolumn{c}}{}%
\IfStrEq{\LWR@strresult}{R}{\LWR@parsenormalcolumn{r}}{}%
\IfStrEq{\LWR@strresult}{J}{\LWR@parsenormalcolumn{l}}{}%

\IfStrEq{\LWR@strresult}{S}{\LWR@parsenormalcolumn{c}}{}%
\IfStrEq{\LWR@strresult}{s}{\LWR@parsenormalcolumn{c}}{}%

\IfStrEq{\LWR@strresult}{\detokenize{@}}{\LWR@parseatcolumn}{}%
\IfStrEq{\LWR@strresult}{!}{\LWR@parsebangcolumn}{}%
\IfStrEq{\LWR@strresult}{>}{\LWR@parsebeforecolumn}{}%
\IfStrEq{\LWR@strresult}{<}{\LWR@parseaftercolumn}{}%
\IfStrEq{\LWR@strresult}{|}{\LWR@parsebarcolumn}{}%
```

```
\IfStrEq{\LWR@strresult}{:}{\LWR@parsecoloncolumn}{}%
\IfStrEq{\LWR@strresult}{;}{\LWR@parsesemicoloncolumn}{}%

\IfStrEq{\LWR@strresult}{p}{\LWR@parsepcolumn{p}}{}%
\IfStrEq{\LWR@strresult}{m}{\LWR@parsepcolumn{m}}{}%
\IfStrEq{\LWR@strresult}{b}{\LWR@parsepcolumn{b}}{}%

\IfStrEq{\LWR@strresult}{w}{\LWR@parsewcolumn}{}%
\IfStrEq{\LWR@strresult}{W}{\LWR@parsewcolumn}{}%
```

From the dcolumn package:

```
\IfStrEq{\LWR@strresult}{D}{\LWR@parseDcolumn{c}}{}%
```

From the tabularx package. X column has no parameter, but will be given paragraph tags.

```
\IfStrEq{\LWR@strresult}{X}{\LWR@parsenormalcolumn{X}}{}%
```

Many people define centered versions "P", "M", and "B":

```
\newcolumntype{P}[1]{>{\centering\arraybackslash}p{#1}}
```

```
\IfStrEq{\LWR@strresult}{P}{\LWR@parsepcolumn{P}}{}%
\IfStrEq{\LWR@strresult}{M}{\LWR@parsepcolumn{M}}{}%
\IfStrEq{\LWR@strresult}{B}{\LWR@parsepcolumn{B}}{}%
```

If this column was an invalid column type, convert it to an l column:

```
\ifbool{LWR@validtablecol}{}{%
    \LWR@traceinfo{invalid column type: \LWR@strresult}%
    \LWR@parsenormalcolumn{l}%
}%
}% not an optional column argument
```

If read the closing bracket, no longer inside the optional argument:

```
\IfStrEq{\LWR@strresult}{]}{\global\boolfalse{LWR@opttablecol}}{}%
```

Move to the next character:

```
\addtocounter{LWR@tablecolspecindex}{1}%
}% whiledo
}%
```

71.14 colortbl and xparse 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.

```
\@ifundefined{rownum}{\newcount\rownum}{}
```

`\@rowcolors` Emulated in case xcolor is not used.

```
\newcommand*{\@rowcolors}{}
```

`\@rowc@lors` Emulated in case xcolor is not used.

```
\newcommand*{\@rowc@lors}{}
```

`\LWR@xcolorrowHTMLcolor` Emulated xcolor row color.

```
\newcommand*{\LWR@xcolorrowHTMLcolor}{}
```

`\LWR@columnHTMLcolor` HTMLstyle code for the column color.

```
\def\LWR@columnHTMLcolor{}
```

`\LWR@rowHTMLcolor` HTMLstyle code for the row color.

```
\def\LWR@rowHTMLcolor{}
```

`\LWR@cellHTMLcolor` HTMLstyle code for the cell color.

```
\def\LWR@cellHTMLcolor{}
```

`\LWR@ruleHTMLcolor` HTMLstyle code for the cell color.

```
\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`.

```
\newcommand*{\rowcolor}{\LWR@getmynexttoken}%
```

`\arrayrulecolor` `[`⟨*model*⟩`]` `{`⟨*color*⟩`}`

\arrayrulecolornexttoken [⟨*model*⟩] {⟨*color*⟩}

Print versions for use outside and inside a tabular:

```
\newcommand{\arrayrulecolor}[2][named]{}
\newcommand{\arrayrulecolornexttoken}[2][named]{\LWR@getmynexttoken}
```

\doublerulesepcolor [⟨*model*⟩] {⟨*color*⟩}

\doublerulesepcolornexttoken [⟨*model*⟩] {⟨*color*⟩}

Print versions for use inside and outside a tabular:

```
\newcommand{\doublerulesepcolor}[2][named]{}
\newcommand{\doublerulesepcolornexttoken}[2][named]{\LWR@getmynexttoken}
```

71.15 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.

```
\newcommand*{\LWR@maybenewtablerow}
{%
\ifbool{LWR@startedrow}%
{}% started the row
{% not started the row
```

Remember that now have started the row:

```
    \global\booltrue{LWR@startedrow}%
```

Create the row tag, with a class if necessary.

```
    \global\booltrue{LWR@intabularmetadata}%
    \ifboolexpr{%
        test{\ifnumcomp{\value{LWR@hlines}}{>}{0}} or%
        test{\ifnumcomp{\value{LWR@hdashedlines}}{>}{0}}%
    }%
    {%
        \LWR@htmltag{tr class="hline" }%
        \LWR@orignewline%
    }%
    {% not doing hline
        \ifbool{LWR@doingtbrule}%
        {%
            \ifdefvoid{\LWR@ruleHTMLcolor}{%
                \LWR@htmltag{tr class="tbrule"}%
            }{%
                \LWR@htmltag{%
                    tr class="tbrule" % space
                    style="border-top: 1px solid % space
```

```
                        \LWR@origpound\LWR@ruleHTMLcolor "%
                }%
            }%
            \LWR@orignewline%
        }%
        {\LWR@htmltag{tr}\LWR@orignewline}%
    }% end of not doing hline
}% end of not started the row
}
```

71.16 Printing vertical bar tags

\LWR@printbartag {⟨*index*⟩}

Adds to a tabular data cell an HTML class name for a left/right vertical bar.

```
\newcommand*{\LWR@printbartag}[1]{%
\LWR@traceinfo{LWR@printbartag !#1!}%
\ifboolexpr{bool{LWR@tabularmutemods} or bool{LWR@emptyatbang}}%
{}% muting or empty
{% not muting
    \edef\LWR@tempone{\LWR@getexparray{LWR@colbarspec}{#1}}%
    \ifdefempty{\LWR@tempone}{}{ \LWR@tempone}%
}% not muting
\LWR@traceinfo{LWR@printbartag done}%
}
```

71.17 Printing at or bang tags

\LWR@printatbang {⟨*at -or- bang*⟩} {⟨*index*⟩}

```
\newcommand*{\LWR@printatbang}[2]{%
```

Fetch the column at or bang spec:

```
\xdef\LWR@atbangspec{\LWR@getexparray{LWR@col#1spec}{#2}}%
\LWR@traceinfo{atbang: #2 !\LWR@atbangspec!}%
```

Only generate if is not empty;

```
\ifdefempty{\LWR@atbangspec}%
{}%
{% not empty
    \LWR@htmltag{%
        td class="td#1%
        \LWR@subaddcmidruletrim{}{}%
        \LWR@printbartag{#2}%
        "%
        \LWR@tdstartstyles%
```

```
        \LWR@addcmidrulewidth%
        \LWR@addcdashline%
        \LWR@addtabularrulecolors%
        \LWR@tdendstyles%
    }%
```

Create an empty cell if muting for the \bottomrule:

```
    \ifboolexpr{bool{LWR@tabularmutemods} or bool{LWR@emptyatbang}}%
    {}%
    {\LWR@atbangspec}%
%
    \LWR@htmltag{/td}\LWR@orignewline%
    \global\booltrue{LWR@tabularcelladded}%
}% not empty
}%
```

\LWR@addleftmostbartag

```
\newcommand*{\LWR@addleftmostbartag}{%
\ifnumcomp{\value{LWR@tableLaTeXcolindex}}{=}{1}{%
    \LWR@printbartag{leftedge}%
}{}%
}
```

\LWR@tabularleftedge

```
\newcommand*{\LWR@tabularleftedge}{%
\ifnumcomp{\value{LWR@tableLaTeXcolindex}}{=}{1}%
{%
    \LWR@printatbang{at}{leftedge}%
    \LWR@printatbang{bang}{leftedge}%
}% left edge
{}% not left edge
}
```

71.18 Data opening tag

\LWR@thiscolspec Temporary storage.

```
\newcommand*{\LWR@thiscolspec}{}
```

\LWR@tabledatasinglecolumntag Print a table data opening tag with style for alignment and color.

```
\newcommand*{\LWR@tabledatasinglecolumntag}%
{%
\LWR@traceinfo{LWR@tabledatasinglecolumntag}%
\LWR@maybenewtablerow%
```

Don't start a new paragraph tag if have already started one:

```
\ifbool{LWR@intabularmetadata}%
{%
```

If have found the end of tabular command, do not create the next data cell:

```
    \ifbool{LWR@exitingtabular}{}%
    {% not exiting tabular
```

Print the `@` and `!` contents before first column:

```
        \LWR@tabularleftedge%
```

Fetch the current column's alignment character into `\LWR@strresult`:

```
        \xdef\LWR@strresult{%
          \LWR@getexparray{LWR@tablecolspec}{\arabic{LWR@tableLaTeXcolindex}}%
        }%
```

print the start of a new table data cell:

```
\LWR@traceinfo{LWR@tabledatasinglecolumntag: about to print td tag}%
        \LWR@htmltag{td class="td%
```

append this column's spec:

```
        \LWR@strresult%
```

If this column has a cmidrule, add "rule" to the end of the HTML class tag. Also add vertical bar tags.

```
        \LWR@addcmidruletrim%
        \LWR@addleftmostbartag%
        \LWR@printbartag{\arabic{LWR@tableLaTeXcolindex}}%
        "%
```

Add styles for rules, alignment:

```
        \LWR@tdstartstyles%
        \LWR@addcmidrulewidth%
        \LWR@addcdashline%


        \xdef\LWR@thiscolspec{%
          \LWR@getexparray{LWR@tablecolspec}{\arabic{LWR@tableLaTeXcolindex}}%
        }%
        \LWR@addformatwpalignment{\LWR@thiscolspec}%
```

Add styles for cell and rule colors:

```
        \LWR@addtabularrowcolor%
        \LWR@addtabularrulecolors%
```

```
        \LWR@tdendstyles%
        }% HTML td
        \LWR@traceinfo{LWR@tabledatasinglecolumntag: done printing td tag}%
```

If this is a p, m, b, or X column, allow paragraphs:

```
        \ifboolexpr{%
            test{ \ifdefstring{\LWR@strresult}{p} } or
            test{ \ifdefstring{\LWR@strresult}{m} } or
            test{ \ifdefstring{\LWR@strresult}{b} } or
            test{ \ifdefstring{\LWR@strresult}{P} } or
            test{ \ifdefstring{\LWR@strresult}{M} } or
            test{ \ifdefstring{\LWR@strresult}{B} } or
            test{ \ifdefstring{\LWR@strresult}{X} }
        }%
        {% allow pars
          \LWR@traceinfo{LWR@tabledatasinglecolumntag: about to LWR@startpars}%
            \global\booltrue{LWR@tableparcell}%
            \LWR@startpars%
         \LWR@traceinfo{LWR@tabledatasinglecolumntag: done with LWR@startpars}%
        }% allow pars
        {}% no pars
```

Print the > contents unless muted for the \bottomrule:

```
        \ifboolexpr{bool{LWR@tabularmutemods} or bool{LWR@emptyatbang}}%
        {}%
        {%
         \LWR@getexparray{LWR@colbeforespec}{\arabic{LWR@tableLaTeXcolindex}}%
        }%
        \global\boolfalse{LWR@intabularmetadata}%
    }% not exiting tabular
}{}% in tabular metadata
\LWR@traceinfo{LWR@tabledatasinglecolumntag: done}%
}%
```

71.19 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.

Ctr LWR@midrulecounter — Indexes across the LWR@midrules and LWR@trim<l/r>rules data arrays.

```
7483 \newcounter{LWR@midrulecounter}
```

Len \LWR@heavyrulewidth The default width of the rule.

```
7484 \newlength{\LWR@heavyrulewidth}
7485 \setlength{\LWR@heavyrulewidth}{.08em}
```

Len \LWR@lightrulewidth The default width of the rule.

```
7486 \newlength{\LWR@lightrulewidth}
7487 \setlength{\LWR@lightrulewidth}{.05em}
```

Len \LWR@cmidrulewidth The default width of the rule.

```
7488 \newlength{\LWR@cmidrulewidth}
7489 \setlength{\LWR@cmidrulewidth}{.03em}
```

Len \LWR@thiscmidrulewidth 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.

```
7490 \newlength{\LWR@thiscmidrulewidth}
7491 \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.

```
7492 \newcommand*{\LWR@clearmidrules}
7493 {%
7494 \setcounter{LWR@midrulecounter}{1}%
7495 \whileboolexpr{%
7496     not test{%
7497         \ifnumcomp{\value{LWR@midrulecounter}}{>}%
7498             {\value{LWR@tabletotalLaTeXcols}}%
7499     }%
7500 }%
7501 {%
7502     \LWR@setexparray{LWR@midrules}{\arabic{LWR@midrulecounter}}{0pt}%
7503     \setlength{\LWR@thiscmidrulewidth}{\LWR@cmidrulewidth}%
7504     \LWR@setexparray{LWR@trimlrules}{\arabic{LWR@midrulecounter}}{}%
7505     \LWR@setexparray{LWR@trimrrules}{\arabic{LWR@midrulecounter}}{}%
7506     \LWR@setexparray{LWR@cdashlines}{\arabic{LWR@midrulecounter}}{N}%
7507     \addtocounter{LWR@midrulecounter}{1}%
7508 }%
7509 }
```

\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.

```
\newcommand*{\LWR@subcmidrule}[4]{%
\setcounter{LWR@midrulecounter}{#3}%
\whileboolexpr{%
    not test {%
        \ifnumcomp{\value{LWR@midrulecounter}}{>}{#4}%
    }%
}%
{%
    \LWR@setexparray{LWR@midrules}{\arabic{LWR@midrulecounter}}{#1}%
    \addtocounter{LWR@midrulecounter}{1}%
}% whiledo
\IfSubStr{#2}{l}{\LWR@setexparray{LWR@trimlrules}{#3}{l}}{}%
\IfSubStr{#2}{r}{\LWR@setexparray{LWR@trimrrules}{#4}{r}}{}%
\global\booltrue{LWR@doingcmidrule}%
}
```

\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.

```
\NewDocumentCommand{\LWR@docmidrule}
    {O{\LWR@cmidrulewidth} D(){} >{\SplitArgument{1}{-}}m}
    {\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.

```
\newcommand*{\LWR@subcdashline}[2]{%
\setcounter{LWR@midrulecounter}{#1}%
\whileboolexpr{%
    not test {%
        \ifnumcomp{\value{LWR@midrulecounter}}{>}{#2}%
    }%
}%
{%
    \LWR@setexparray{LWR@cdashlines}{\arabic{LWR@midrulecounter}}{Y}%
    \addtocounter{LWR@midrulecounter}{1}%
}% whiledo
\global\booltrue{LWR@doingcmidrule}%
}
```

\LWR@docdashline {⟨*leftcolumn-rightcolumn*⟩}

Marks LWR@cdashlines data array elements to be Y from left to right columns.

```
\NewDocumentCommand{\LWR@docdashline}
    {>{\SplitArgument{1}{-}}m}%
    {%
        \LWR@subcdashline#1%
    }
```

Used to compute margins, tabular trims, column offsets:

```
\newlength{\LWR@templengthone}
\newlength{\LWR@templengthtwo}
\newlength{\LWR@templengththree}
\newcounter{LWR@tempcountone}
```

Used to add a style to a table data cell:

```
\newbool{LWR@tdhavecellstyle}
```

`\LWR@tdstartstyles` Begins possibly adding a table data cell style.

```
\newcommand*{\LWR@tdstartstyles}{\global\boolfalse{LWR@tdhavecellstyle}}
```

`\LWR@tdaddstyle` Starts adding a table data cell style.

```
\newcommand*{\LWR@tdaddstyle}{%
\ifbool{LWR@tdhavecellstyle}%
{; }%
{ style="}%
\booltrue{LWR@tdhavecellstyle}%
}
```

`\LWR@tdendstyles` Finishes possibly adding a table data cell style. Prints the closing quote.

```
\newcommand*{\LWR@tdendstyles}{%
\ifbool{LWR@tdhavecellstyle}{%
    "%
    \global\boolfalse{LWR@tdhavecellstyle}%
}{}%
}
```

`\LWR@subaddcmidruletrim` {⟨*lefttrim*⟩} {⟨*righttrim*⟩} Adds a `\cmidrule` with optional trim.

```
\newcommand*{\LWR@subaddcmidruletrim}[2]{%
\setlength{\LWR@templengthone}{%
        \LWR@getexparray{LWR@midrules}{\arabic{LWR@tableLaTeXcolindex}}%
}%
\ifdimcomp{\LWR@templengthone}{>}{0pt}%
{%
```

Print the class with left and right trim letters appended:

```
    \LWR@origtilde tdrule#1#2%
```

Remember the width of the rule:

```
    \setlength{\LWR@thiscmidrulewidth}{\LWR@templengthone}%
}%
{%
    \setlength{\LWR@thiscmidrulewidth}{0pt}%
}%
}
```

`\LWR@addcmidruletrim` Adds left or right trim to a `\cmidrule`.

```
\newcommand*{\LWR@addcmidruletrim}{%
\LWR@subaddcmidruletrim%
{\LWR@getexparray{LWR@trimlrules}{\arabic{LWR@tableLaTeXcolindex}}}%
{\LWR@getexparray{LWR@trimrrules}{\arabic{LWR@tableLaTeXcolindex}}}%
}
```

`\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`.

```
\newcommand{\LWR@addrulewidth}[2]{%
```

Only add a custom width if `thiswidth` is different than the `defaultwidth`, or if a color is being used:

```
\ifboolexpr{%
    test{\ifdimcomp{#1}{=}{0pt}} or
    (
        ( test{\ifdimcomp{#1}{=}{#2}} and not bool{FormatWP} )
        and ( test {\ifdefvoid{\LWR@ruleHTMLcolor}} )
    )
}%
{}% default width and color
{% custom width and/or color
```

Ensure that the width is wide enough to display in the browser:

```
    \LWR@forceminwidth{#1}%
```

Begin adding another style:

```
    \LWR@tdaddstyle%
```

The style itself:

```
    border-top:\LWR@printlength{\LWR@atleastonept} solid %
```

If default gray, the darkness of the color depends on the thickness of the rule:

```
    \ifdefvoid{\LWR@ruleHTMLcolor}{%
        \ifdimcomp{#1}{<}{\LWR@lightrulewidth}%
        {\LWR@origpound{}A0A0A0}%
        {% lightrule or heaver
            \ifdimcomp{#1}{<}{\LWR@heavyrulewidth}%
            {\LWR@origpound{}808080}%
            {black}%
        }% lightrule or heavier
    }{%
        \LWR@origpound\LWR@ruleHTMLcolor%
    }
}% custom width and/or color
}
```

`\LWR@addcmidrulewidth` Adds a `style` for the rule width.

Must be placed between `\LWR@tdstartstyles` and `\LWR@tdendstyles`.

```
\newcommand{\LWR@addcmidrulewidth}{%
\LWR@addrulewidth{\LWR@thiscmidrulewidth}{\LWR@cmidrulewidth}%
}
```

`\LWR@addcdashline` Must be placed between `\LWR@tdstartstyles` and `\LWR@tdendstyles`.

```
\newcommand{\LWR@addcdashline}{%
\edef\LWR@tempone{%
    \LWR@getexparray{LWR@cdashlines}{\arabic{LWR@tableLaTeXcolindex}}%
}%
\ifdefstring{\LWR@tempone}{Y}{%
    \LWR@tdaddstyle%
    border-top: 1pt dashed %
    \ifdefvoid{\LWR@ruleHTMLcolor}%
        {black}%
        {\LWR@origpound\LWR@ruleHTMLcolor}%
}{}%
}
```

`\LWR@WPcell` {⟨*text-align*⟩} {⟨*vertical-align*⟩}

```
\newcommand*{\LWR@WPcell}[2]{%
\LWR@tdaddstyle%
\LWR@print@mbox{text-align:#1; \LWR@print@mbox{vertical-align:#2}%
}
```

`\LWR@addformatwpalignment` If `FormatWP`, adds a `style` for the alignment.

Must be placed between `\LWR@tdstartstyles` and `\LWR@tdendstyles`.

```
\newcommand*{\LWR@addformatwpalignment}[1]{%
\ifbool{FormatWP}{%
\IfSubStr{#1}{l}{\LWR@WPcell{left}{middle}}{}%
\IfSubStr{#1}{c}{\LWR@WPcell{center}{middle}}{}%
```

```
\IfSubStr{#1}{r}{\LWR@WPcell{right}{middle}}{}%
\IfSubStr{#1}{p}{\LWR@WPcell{left}{bottom}}{}%
\IfSubStr{#1}{m}{\LWR@WPcell{left}{middle}}{}%
\IfSubStr{#1}{b}{\LWR@WPcell{left}{top}}{}%
\IfSubStr{#1}{P}{\LWR@WPcell{center}{bottom}}{}%
\IfSubStr{#1}{M}{\LWR@WPcell{center}{middle}}{}%
\IfSubStr{#1}{B}{\LWR@WPcell{center}{top}}{}%
}{}%
}
```

71.20 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`.

```
\newcommand*{\LWR@addtabularrowcolor}{%
\ifbool{LWR@tabularmutemods}{}{%
    \ifdefvoid{\LWR@rowHTMLcolor}{%
        \ifdefvoid{\LWR@xcolorrowHTMLcolor}{}%
        {% xcolor row color
            \LWR@tdaddstyle%
            background:\LWR@origpound\LWR@xcolorrowHTMLcolor%
        }%
    }%
    {% explicit row color
        \LWR@tdaddstyle%
        background:\LWR@origpound\LWR@rowHTMLcolor%
    }%
}%
}
```

\LWR@addtabularhrulecolor Adds a cell's horizontal rule color style, if needed.

```
\newcommand*{\LWR@addtabularhrulecolor}{%
```

If either form of horizontal rule is requested:

```
\ifboolexpr{%
    test{\ifnumcomp{\value{LWR@hlines}}{>}{0}} or%
    test{\ifnumcomp{\value{LWR@hdashedlines}}{>}{0}} or%
    bool{LWR@doingtbrule}%
}{%
```

If there is a no custom color:

```
    \ifdefvoid{\LWR@ruleHTMLcolor}%
    {%
        \ifnumcomp{\value{LWR@hlines}}{>}{1}%
        {%
            \LWR@tdaddstyle%
```

```
            border-top: 4px double%
        }{% else
        \ifnumcomp{\value{LWR@hdashedlines}}{>}{1}%
        {%
            \LWR@tdaddstyle%
            border-top: 2px dashed%
        }{% else
        \ifnumcomp{\value{LWR@hdashedlines}}{=}{1}%
        {%
            \LWR@tdaddstyle%
            border-top: 1px dashed%
        }{}}}%
```

If no color and not doubled or dashed, then add nothing, since a simpler rule is the default.

```
    }%
```

If there is a custom color:

```
    {%
        \ifnumcomp{\value{LWR@hlines}}{>}{1}%
        {%
            \LWR@tdaddstyle%
            border-top: 4px double \LWR@origpound\LWR@ruleHTMLcolor%
        }{% else
        \ifnumcomp{\value{LWR@hdashedlines}}{>}{1}%
        {%
            \LWR@tdaddstyle%
            border-top: 2px dashed \LWR@origpound\LWR@ruleHTMLcolor%
        }{% else
        \ifnumcomp{\value{LWR@hdashedlines}}{=}{1}%
        {%
            \LWR@tdaddstyle%
            border-top: 1px dashed \LWR@origpound\LWR@ruleHTMLcolor%
        }{% else
            \LWR@tdaddstyle%
            border-top: 1px solid \LWR@origpound\LWR@ruleHTMLcolor%
        }}}%
    }%
}{}%
}
```

`\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`.

```
\newcommand*{\LWR@addtabularrulecolors}{%
```

Custom horizonal rule color:

```
\LWR@addtabularhrulecolor%
```

No vertical rules if finishing the tabular with a row of empty cells:

```
\ifbool{LWR@tabularmutemods}{}{%
```

If at the leftmost cell, possibly add a leftmost vertical rule:

```
    \ifnumequal{\value{LWR@tableLaTeXcolindex}}{1}{%
```

Fetch the left edge's vertical bar specification:

```
        \edef\LWR@tempone{\LWR@getexparray{LWR@colbarspec}{leftedge}}%
```

Add a custom style if a vertical bar was requested:

```
        \ifdefstring{\LWR@tempone}{tvertbarl}{%
                \LWR@tdaddstyle%
                border-left: 1px solid % space
                    \LWR@vertruleHTMLcolor%
        }{}%
        \ifdefstring{\LWR@tempone}{tvertbarldouble}{%
                \LWR@tdaddstyle%
                border-left: 4px double % space
                    \LWR@vertruleHTMLcolor%
        }{}%
        \ifdefstring{\LWR@tempone}{tvertbarldash}{%
                \LWR@tdaddstyle%
                border-left: 1px dashed % space
                    \LWR@vertruleHTMLcolor%
        }{}%
        \ifdefstring{\LWR@tempone}{tvertbarldoubledash}{%
                \LWR@tdaddstyle%
                border-left: 2px dashed % space
                    \LWR@vertruleHTMLcolor%
        }{}%
    }{}%
```

Possibly add a right vertical rule for this cell:

```
    \edef\LWR@tempone{%
        \LWR@getexparray{LWR@colbarspec}{\arabic{LWR@tableLaTeXcolindex}}%
    }%
    \ifdefstring{\LWR@tempone}{tvertbarr}{%
```

Add a custom style if a vertical bar was requested:

```
            \LWR@tdaddstyle%
            border-right: 1px solid \LWR@vertruleHTMLcolor%
    }{}%
    \ifdefstring{\LWR@tempone}{tvertbarrdouble}{%
            \LWR@tdaddstyle%
            border-right: 4px double \LWR@vertruleHTMLcolor%
    }{}%
    \ifdefstring{\LWR@tempone}{tvertbarrdash}{%
            \LWR@tdaddstyle%
```

```
            border-right: 1px dashed \LWR@vertruleHTMLcolor%
    }{}%
    \ifdefstring{\LWR@tempone}{tvertbarrdoubledash}{%
            \LWR@tdaddstyle%
            border-right: 2px dashed \LWR@vertruleHTMLcolor%
    }{}%
}%
}
```

Ctr `LWR@cellcolordepth` Counts how many cell color `<div>`s were added to the current tabular data cell.

```
\newcounter{LWR@cellcolordepth}
```

`\LWR@subaddtabularcellcolor` {⟨*HTML color*⟩}

```
\newcommand*{\LWR@subaddtabularcellcolor}[1]{%
\LWR@htmltag{div class="cellcolor" style="%
    background:\LWR@origpound{}{}#1 %
" }%
\addtocounter{LWR@cellcolordepth}{1}%
}
```

`\LWR@addtabularcellcolor` Adds a cell color style, if needed.

```
\newcommand*{\LWR@addtabularcellcolor}{%
\ifdefvoid{\LWR@cellHTMLcolor}%
{%
    \ifdefvoid{\LWR@rowHTMLcolor}%
    {%
        \ifdefvoid{\LWR@xcolorrowHTMLcolor}%
        {%
            \ifdefvoid{\LWR@columnHTMLcolor}%
            {}%
            {\LWR@subaddtabularcellcolor{\LWR@columnHTMLcolor}}%
        }%
        {\LWR@subaddtabularcellcolor{\LWR@xcolorrowHTMLcolor}}%
    }%
    {\LWR@subaddtabularcellcolor{\LWR@rowHTMLcolor}}%
}%
{\LWR@subaddtabularcellcolor{\LWR@cellHTMLcolor}}%
}
```

71.21 Multicolumns

71.21.1 Parsing multicolumns

```
\newcounter{LWR@tablemulticolswidth}
```

Indexes into the multicolumn specification:

```
\newcounter{LWR@tablemulticolspos}
```

Remembers multicolumn vertical rules if found in the column spec.

```
\newcounter{LWR@mcolvertbarsl}
\newcounter{LWR@mcolvertbarsr}
\newcounter{LWR@mcolvertbarsldash}
\newcounter{LWR@mcolvertbarsrdash}
\newbool{LWR@mcolvertbaronleft}%
```

`\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`.

```
\newcommand*{\LWR@printmccoltype}[1]{%
\LWR@traceinfo{lwr@printmccoltype -#1-}%
```

Get one token of the column spec:

```
\StrChar{#1}{\arabic{LWR@tablemulticolspos}}[\LWR@strresult]%
```

Add to the HTML tag depending on which column type is found:

```
\IfStrEq{\LWR@strresult}{l}{l}{}%
\IfStrEq{\LWR@strresult}{c}{c}{}%
\IfStrEq{\LWR@strresult}{r}{r}{}%
\IfStrEq{\LWR@strresult}{p}{p}{}%
\IfStrEq{\LWR@strresult}{m}{m}{}%
\IfStrEq{\LWR@strresult}{b}{b}{}%
\IfStrEq{\LWR@strresult}{P}{P}{}%
\IfStrEq{\LWR@strresult}{M}{M}{}%
\IfStrEq{\LWR@strresult}{B}{B}{}%

\IfStrEq{\LWR@strresult}{w}{w}{}%
\IfStrEq{\LWR@strresult}{W}{W}{}%

\IfStrEq{\LWR@strresult}{S}{c}{}%
\IfStrEq{\LWR@strresult}{s}{c}{}%

\IfStrEq{\LWR@strresult}{X}{p}{}%

\IfStrEq{\LWR@strresult}{|}%
{%
    \ifbool{LWR@mcolvertbaronleft}%
        {\addtocounter{LWR@mcolvertbarsl}{1}}% left edge
        {\addtocounter{LWR@mcolvertbarsr}{1}}% not left edge
}%
{%
    \IfStrEq{\LWR@strresult}{:}%
    {%
        \ifbool{LWR@mcolvertbaronleft}%
            {\addtocounter{LWR@mcolvertbarsldash}{1}}% left edge
```

```
            {\addtocounter{LWR@mcolvertbarsrdash}{1}}% not left edge
    }%
    {%
        \IfStrEq{\LWR@strresult}{;}%
        {%
            \ifbool{LWR@mcolvertbaronleft}%
                {\addtocounter{LWR@mcolvertbarsldash}{1}}% left edge
                {\addtocounter{LWR@mcolvertbarsrdash}{1}}% not left edge
        }%
        {\boolfalse{LWR@mcolvertbaronleft}}%
    }%
}%
\LWR@traceinfo{lwr@printmccoltype done}%
}
```

`\LWR@multicolpartext` {⟨*num parameters*⟩} Print the data with paragraph tags, advance to bypass the given number of parameters.

```
\newcommand*{\LWR@multicolpartext}[1]{%
\LWR@startpars%
\LWR@multicoltext%
\addtocounter{LWR@tablemulticolspos}{#1}%
\LWR@stoppars%
}
```

`\LWR@multicolother` {⟨*colspec*⟩} For `@`, `!`, `>`, `<`, print the next token without paragraph tags:

```
\newcommand*{\LWR@multicolother}[1]{%
\addtocounter{LWR@tablemulticolspos}{1}%
\StrChar{#1}{\arabic{LWR@tablemulticolspos}}[\LWR@strresult]%
\LWR@strresult%
```

A valid column data type was found:

```
\global\booltrue{LWR@validtablecol}%
}
```

`\LWR@multicolskip` Nothing to print for this column type.

```
\newcommand*{\LWR@multicolskip}{%
```

A valid column data type was found:

```
\global\booltrue{LWR@validtablecol}%
}
```

`\LWR@printmccoldata` {⟨*colspec*⟩} Print the data for any valid column type found.

```
\newcommand*{\LWR@printmccoldata}[1]{%
\LWR@traceinfo{lwr@printmccoldata -#1}%
```

Not yet found a valid column type:

```
\global\boolfalse{LWR@validtablecol}%
```

Get one token of the column spec:

```
\StrChar{#1}{\arabic{LWR@tablemulticolspos}}[\LWR@strresult]%
```

Print the text depending on which column type is found. Also handles `@`, `>`, `<` as it comes to them.

```
\IfStrEq{\LWR@strresult}{l}{\LWR@multicoltext}{}%
\IfStrEq{\LWR@strresult}{c}{\LWR@multicoltext}{}%
\IfStrEq{\LWR@strresult}{r}{\LWR@multicoltext}{}%
\IfStrEq{\LWR@strresult}{D}{%
\addtocounter{LWR@tablemulticolspos}{3}% skip parameters
\LWR@multicoltext%
}{}%


\IfStrEq{\LWR@strresult}{p}{\LWR@multicolpartext{0}}{}%
\IfStrEq{\LWR@strresult}{m}{\LWR@multicolpartext{0}}{}%
\IfStrEq{\LWR@strresult}{b}{\LWR@multicolpartext{0}}{}%
\IfStrEq{\LWR@strresult}{P}{\LWR@multicolpartext{0}}{}%
\IfStrEq{\LWR@strresult}{M}{\LWR@multicolpartext{0}}{}%
\IfStrEq{\LWR@strresult}{B}{\LWR@multicolpartext{0}}{}%


\IfStrEq{\LWR@strresult}{w}{\LWR@multicolpartext{3}}{}%
\IfStrEq{\LWR@strresult}{W}{\LWR@multicolpartext{3}}{}%

\IfStrEq{\LWR@strresult}{S}{\LWR@multicoltext}{}%
\IfStrEq{\LWR@strresult}{s}{\LWR@multicoltext}{}%

\IfStrEq{\LWR@strresult}{X}{\LWR@multicolpartext{0}}{}%
\IfStrEq{\LWR@strresult}{|}{\LWR@multicolskip}{}%
\IfStrEq{\LWR@strresult}{:}{\LWR@multicolskip}{}%
\IfStrEq{\LWR@strresult}{;}{%
    \LWR@multicolskip%
    \addtocounter{LWR@tablemulticolspos}{1}% skip parameter
}{}%
\IfStrEq{\LWR@strresult}{\detokenize{@}}{\LWR@multicolother{#1}}{}%
\IfStrEq{\LWR@strresult}{\detokenize{!}}{\LWR@multicolother{#1}}{}%
\IfStrEq{\LWR@strresult}{\detokenize{>}}{\LWR@multicolother{#1}}{}%
\IfStrEq{\LWR@strresult}{\detokenize{<}}{\LWR@multicolother{#1}}{}%
```

If an invalid column type:

```
\ifbool{LWR@validtablecol}{}{\LWR@multicoltext}%
```

Tracing:

```
\LWR@traceinfo{lwr@printmccoldata done}%
}
```

\parsemulticolumnalignment {⟨*1: colspec*⟩} {⟨*2: printresults*⟩}

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.

```
\newcommand*{\LWR@parsemulticolumnalignment}[2]{%
\setcounter{LWR@tablemulticolspos}{1}%
\StrLen{#1}[\LWR@strresult]%
\setcounter{LWR@tablemulticolswidth}{\LWR@strresult}%
```

Scan across the tokens in the column spec:

```
\whileboolexpr{%
    not test {%
        \ifnumcomp{\value{LWR@tablemulticolspos}}{>}%
            {\value{LWR@tablemulticolswidth}}%
    }%
}%
{%
```

Execute the assigned print function for each token in the column spec:

```
#2{#1}%
```

Move to the next token in the column spec:

```
\addtocounter{LWR@tablemulticolspos}{1}%
}%
}
```

71.21.2 Multicolumn factored code

\LWR@addmulticolvertrulecolor

```
\newcommand*{\LWR@addmulticolvertrulecolor}{%
```

No vertical rules if finishing the tabular with a row of empty cells:

```
\ifbool{LWR@tabularmutemods}{}{%
```

Left side:

```
    \ifnumcomp{\value{LWR@mcolvertbarsl}}{=}{1}{%
        \LWR@tdaddstyle%
        border-left: 1px solid \LWR@vertruleHTMLcolor%
    }{}%
    \ifnumcomp{\value{LWR@mcolvertbarsl}}{>}{1}{%
        \LWR@tdaddstyle%
        border-left: 4px double \LWR@vertruleHTMLcolor%
    }{}%
```

```
    \ifnumcomp{\value{LWR@mcolvertbarsldash}}{=}{1}{%
        \LWR@tdaddstyle%
        border-left: 1px dashed \LWR@vertruleHTMLcolor%
    }{}%
    \ifnumcomp{\value{LWR@mcolvertbarsldash}}{>}{1}{%
        \LWR@tdaddstyle%
        border-left: 2px dashed \LWR@vertruleHTMLcolor%
    }{}%
```

Right side:

```
    \ifnumcomp{\value{LWR@mcolvertbarsr}}{=}{1}{%
        \LWR@tdaddstyle%
        border-right: 1px solid \LWR@vertruleHTMLcolor%
    }{}%
    \ifnumcomp{\value{LWR@mcolvertbarsr}}{>}{1}{%
        \LWR@tdaddstyle%
        border-right: 4px double \LWR@vertruleHTMLcolor%
    }{}%
    \ifnumcomp{\value{LWR@mcolvertbarsrdash}}{=}{1}{%
        \LWR@tdaddstyle%
        border-right: 1px dashed \LWR@vertruleHTMLcolor%
    }{}%
    \ifnumcomp{\value{LWR@mcolvertbarsrdash}}{>}{1}{%
        \LWR@tdaddstyle%
        border-right: 2px dashed \LWR@vertruleHTMLcolor%
    }{}%
}%
}
```

```
\newcommand{\LWR@multicoltext}{}
```

To find multicolumn right trim:

```
\newcounter{LWR@lastmulticolumn}
```

`\LWR@domulticolumn` `[`⟨*1: vpos*⟩`]` `[`⟨*2: #rows*⟩`]` {⟨*3: numLaTeXcols*⟩} {⟨*4: numHTMLcols*⟩} {⟨*5: colspec*⟩} {⟨*6: text*⟩}

```
\NewDocumentCommand{\LWR@domulticolumn}{o o m m m +m}{%
\LWR@traceinfo{LWR@domulticolumn -#1- -#2- -#4- -#5-}%
```

Remember the text to be inserted, and remember that a valid column type was found:

```
\renewcommand{\LWR@multicoltext}{%
#6%
\global\booltrue{LWR@validtablecol}%
}%
```

Compute the rightmost column to be included. This is used to create the right trim.

```
\setcounter{LWR@lastmulticolumn}{\value{LWR@tableLaTeXcolindex}}%
\addtocounter{LWR@lastmulticolumn}{#3}%
\addtocounter{LWR@lastmulticolumn}{-1}%
```

Row processing:

```
\LWR@maybenewtablerow%
```

Begin the opening table data tag:

```
\LWR@htmltag{td colspan="#4" %

\IfValueT{#2}{ % rows?
rowspan="#2" %


\IfValueT{#1}{% vpos?
\ifstrequal{#1}{b}{style="\LWR@print@mbox{vertical-align:bottom}" }{}%
\ifstrequal{#1}{t}{style="\LWR@print@mbox{vertical-align:top}" }{}%
}% vpos?
}% rows?

 class="td%
```

Print the column type and vertical bars:

```
\setcounter{LWR@mcolvertbarsl}{0}%
\setcounter{LWR@mcolvertbarsr}{0}%
\setcounter{LWR@mcolvertbarsldash}{0}%
\setcounter{LWR@mcolvertbarsrdash}{0}%
\booltrue{LWR@mcolvertbaronleft}%
\LWR@parsemulticolumnalignment{#5}{\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:

```
\LWR@subaddcmidruletrim%
{\LWR@getexparray{LWR@trimlrules}{\arabic{LWR@tableLaTeXcolindex}}}%
{\LWR@getexparray{LWR@trimrrules}{\arabic{LWR@lastmulticolumn}}}%
```

Also add vertical bar class.

```
\ifnumcomp{\value{LWR@mcolvertbarsl}}{=}{1}{ tvertbarl}{}%
\ifnumcomp{\value{LWR@mcolvertbarsl}}{>}{1}{ tvertbarldouble}{}%
\ifnumcomp{\value{LWR@mcolvertbarsr}}{=}{1}{ tvertbarr}{}%
\ifnumcomp{\value{LWR@mcolvertbarsr}}{>}{1}{ tvertbarrdouble}{}%
\ifnumcomp{\value{LWR@mcolvertbarsldash}}{=}{1}{ tvertbarldash}{}%
\ifnumcomp{\value{LWR@mcolvertbarsldash}}{>}{1}{ tvertbarldoubledash}{}%
\ifnumcomp{\value{LWR@mcolvertbarsrdash}}{=}{1}{ tvertbarrdash}{}%
\ifnumcomp{\value{LWR@mcolvertbarsrdash}}{>}{1}{ tvertbarrdoubledash}{}%
```

Close the class tag's opening quote: " NOT A TYPO

```
%

\LWR@tdstartstyles%
```

```
\LWR@addtabularrowcolor%

\LWR@addcmidrulewidth%
\LWR@addcdashline%
\LWR@addtabularhrulecolor%
\LWR@addmulticolvertrulecolor%
\LWR@addformatwpalignment{#5}%
\LWR@tdendstyles%
}% end of the opening table data tag
\global\boolfalse{LWR@intabularmetadata}%
\LWR@parsemulticolumnalignment{#5}{\LWR@printmccoldata}%
}
```

71.21.3 Multicolumn

`\LWR@htmlmulticolumn` {⟨*numcols*⟩} {⟨*alignment*⟩} {⟨*text*⟩}

```
\NewDocumentCommand{\LWR@htmlmulticolumn}{m m +m}%
{%
```

Figure out how many extra HTML columns to add for `@` and `!` columns:

```
\LWR@tabularhtmlcolumns{\arabic{LWR@tableLaTeXcolindex}}{#1}
```

Create the multicolumn tag:

```
\LWR@domulticolumn{#1}{\arabic{LWR@tabhtmlcoltotal}}{#2}{#3}%
```

Move to the next LaTeX column:

```
\addtocounter{LWR@tableLaTeXcolindex}{#1}%
\addtocounter{LWR@tableLaTeXcolindex}{-1}%
```

Skip any trailing `@` or `!` columns for this cell:

```
\global\booltrue{LWR@skipatbang}%
}
```

71.21.4 Longtable captions

longtable captions use `\multicolumn`.

Bool `LWR@starredlongtable` Per the caption pacakge, step the counter if `longtable*`.

```
\newbool{LWR@starredlongtable}
\boolfalse{LWR@starredlongtable}
```

Per the caption package. User-redefinable float type.

```
\providecommand*{\LTcaptype}{table}
```

\LWR@longtabledatacaptiontag * [⟨*toc entry*⟩] {⟨*caption*⟩}

```
\NewDocumentCommand{\LWR@longtabledatacaptiontag}{s o +m}
{%
```

Remember the latest name for \nameref:

```
\IfValueTF{#2}{% optional given?
    \ifblank{#2}% optional empty?
    {\LWR@setlatestname{#3}}% empty
    {\LWR@setlatestname{#2}}% given and non-empty
}% optional given
{\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:

```
\LWR@tabularhtmlcolumns{1}{\arabic{LWR@tabletotalLaTeXcols}}
```

Create the multicolumn tag:

```
\LWR@domulticolumn{\arabic{LWR@tabletotalLaTeXcols}}%
    {\arabic{LWR@tabhtmlcoltotal}}%
    {P}%
{% \LWR@domulticolumn
\IfBooleanTF{#1}% star?
```

Star version, show a caption but do not make a LOT entry:

```
{% yes star
    \LWR@figcaption%
    \LWR@isolate{#3}%
    \endLWR@figcaption%
}%
{% No star:
```

Not the star version:

Don't step the counter if \caption[]{A caption.}

```
    \ifbool{LWR@starredlongtable}%
    {%
        \ifblank{#2}% TOC entry
        {}%
        {%
            \refstepcounter{\LTcaptype}%
            \protected@edef\@currentlabel{%
                \@nameuse{p@\LTcaptype}\@nameuse{the\LTcaptype}%
            }%
        }%
    }{}%
```

Create an HTML caption. Afterwards, maybe make a LOT entry.

```
    \LWR@figcaption%
    \LWR@isolate{\@nameuse{fnum@\LTcaptype}}%
    \CaptionSeparator%
    \LWR@isolate{#3}%
    \endLWR@figcaption%
```

See if an optional caption was given:

```
    \ifblank{#2}% TOC entry empty
```

if the optional caption was given, but empty, do not form a TOC entry

```
    {}%
```

If the optional caption was given, but might only be `[]`:

```
    {% TOC entry not empty
        \IfNoValueTF{#2}% No TOC entry?
```

The optional caption is `[]`:

```
        {% No TOC entry
            \addcontentsline%
            {\@nameuse{ext@\LTcaptype}}%
            {\LTcaptype}%
            {%
                \protect\numberline%
              {\LWR@isolate{\@nameuse{p@\LTcaptype}}\@nameuse{the\LTcaptype}}%
                {\ignorespaces \LWR@isolate{#3}\protect\relax}%
            }%
        }% end of No TOC entry
```

The optional caption has text enclosed:

```
        {% yes TOC entry
            \addcontentsline%
            {\@nameuse{ext@\LTcaptype}}%
            {\LTcaptype}%
            {%
                \protect\numberline%
              {\LWR@isolate{\@nameuse{p@\LTcaptype}}\@nameuse{the\LTcaptype}}%
                {\ignorespaces \LWR@isolate{#2}\protect\relax}%
            }%
        }% end of yes TOC entry
    }% end of TOC entry not empty
}% end of no star
```

Skip any trailing `@` or `!` columns for this cell:

```
\global\booltrue{LWR@skipatbang}%
}% end of \LWR@domulticolumn

```

```
\addtocounter{LWR@tableLaTeXcolindex}{\arabic{LWR@tabletotalLaTeXcols}}
\addtocounter{LWR@tableLaTeXcolindex}{-1}

}
```

71.21.5 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.

```
\newcounter{LWR@tabhtmlcolindex}
\newcounter{LWR@tabhtmlcolend}
\newcounter{LWR@tabhtmlcoltotal}
```

`\LWR@subtabularhtmlcolumns` {⟨*index*⟩}

Factored from \LWr@tabularhtmlcolumns, which follows.

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

Temporarily define a macro equal to the @ specification for this column:

```
    \edef\LWR@atbangspec{\LWR@getexparray{LWR@colatspec}{#1}}%
```

If the @ specification is not empty, add to the count:

```
    \ifdefempty{\LWR@atbangspec}%
        {}%
        {\addtocounter{LWR@tabhtmlcoltotal}{1}}%
```

Likewise for the ! columns:

```
    \edef\LWR@atbangspec{\LWR@getexparray{LWR@colbangspec}{#1}}%
    \ifdefempty{\LWR@atbangspec}%
        {}%
        {\addtocounter{LWR@tabhtmlcoltotal}{1}}%
}
```

`\LWR@tabularhtmlcolumns` {⟨*starting LaTeX column*⟩} {⟨*number LaTeX columns*⟩}

Compute the total number of HTML columns being spanned, considering the starting 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.

```
\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:

```
\setcounter{LWR@tabhtmlcolindex}{#1}%
\setcounter{LWR@tabhtmlcoltotal}{#2}%
\setcounter{LWR@tabhtmlcolend}{#1}%
\addtocounter{LWR@tabhtmlcolend}{#2}%
```

If at the left edge, add the at/bang columns for the left edge:

```
\ifnumcomp{\value{LWR@tabhtmlcolindex}}{=}{1}{%
    \LWR@subtabularhtmlcolumns{leftedge}%
}{}%
```

Walk across the LaTeX columns looking for `@` and `!` columns:

```
\whileboolexpr{%
    test {%
        \ifnumcomp{\value{LWR@tabhtmlcolindex}}{<}{\value{LWR@tabhtmlcolend}}%
    }%
}%
{%
    \LWR@subtabularhtmlcolumns{\arabic{LWR@tabhtmlcolindex}}%
    \addtocounter{LWR@tabhtmlcolindex}{1}%
}% whiledo
}
```

```
\end{warpHTML}
```

71.22 Multirow if not loaded

A default defintion in case multirow is not loaded. This is used during table parsing.

```
\begin{warpHTML}
\newcommand{\multirow}[2][c]{}
\end{warpHTML}
```

71.23 Multicolumnrow

A print-mode version is defined here, and is also used during HTML output while inside a `lateximage`.

See section 314 for the HTML versions.

for HTML & PRINT:

```
\begin{warpall}
```

`\multicolumnrow` {⟨*1:cols*⟩} {⟨*2:halign*⟩} [⟨*3:vpos*⟩] {⟨*4:numrows*⟩} [⟨*5:bigstruts*⟩] {⟨*6:width*⟩} [⟨*7:fixup*⟩] {⟨*8:text*⟩}

For discussion of the use of \DeclareExpandableDocumentCommand, see:
https://tex.stackexchange.com/questions/168434/
problem-with-abbreviation-of-multirow-and-multicolumn-latex

\AtBeginDocument to adjust after the user may have loaded multirow, which requires several tests to determine which version is loaded and thus which options are available.

```
\AtBeginDocument{
```

\@ifundefined{@xmultirow} determines if multirow was never loaded.

Null action if not loaded:

```
\@ifundefined{@xmultirow}
{
\DeclareExpandableDocumentCommand{\LWR@print@multicolumnrow}%
    {+m +m +O{c} +m +O{0} +m +O{0pt} +m}%
    {}%
}% no version of multirow was loaded
{% \@xmultirow defined, so some version of multirow was loaded
```

\@ifpackageloaded{multirow} determines if v2.0 or later of multirow was used, which included the \ProvidesPackage macro.

The print version:

```
\@ifpackageloaded{multirow}{% v2.0 or newer
\@ifpackagelater{multirow}{2016/09/01}% 2016/09/27 for v2.0
{% v2.0+:
\DeclareExpandableDocumentCommand{\LWR@print@multicolumnrow}%
    {+m +m +O{c} +m +O{0} +m +O{0pt} +m}%
    {\multicolumn{#1}{#2}{\@xmultirow[#3]{#4}[#5]{#6}[#7]{#8}}}%
}
{% loaded but older, probably not executed:
\DeclareExpandableDocumentCommand{\LWR@print@multicolumnrow}%
    {+m +m +O{c} +m +O{0} +m +O{0pt} +m}%
    {\multicolumn{#1}{#2}{\@xmultirow{#4}[#5]{#6}[#7]{#8}}}%
}
}% packageloaded{multirow}
```

If not \@ifpackageloaded{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.

```
{% v1.6 or older did not \ProvidePackage
\DeclareExpandableDocumentCommand{\LWR@print@multicolumnrow}%
    {+m +m +O{c} +m +O{0} +m +O{0pt} +m}%
    {\multicolumn{#1}{#2}{\@xmultirow{#4}[#5]{#6}[#7]{#8}}}%
}

}% \@ifundefined{@xmultirow}

\providecommand*{\multicolumnrow}{\LWR@print@multicolumnrow}

```

```
}% AtBeginDocument

\end{warpall}
```

71.24 Utility macros inside a table

for HTML output:

```
\begin{warpHTML}
```

Used to prevent opening a tabular data cell if the following token is one which does not create tabular data:

```
\newcommand*{\LWR@donothing}{}
```

In case array is not loaded:

```
\let\firsthline\relax
\let\lasthline\relax
\newcommand*{\firsthline}{}
\newcommand*{\lasthline}{}
```

In case bigdelim is not loaded:

```
\newcommand*{\ldelim}{}
\newcommand*{\rdelim}{}

\end{warpHTML}
```

71.25 Special-case tabular markers

for HTML & PRINT:

```
\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 9.10.

```
\newcommand*{\TabularMacro}{}

\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 % because & is used in a definition
\newenvironment{outerenvironment}
{
\tabular{cc}
left & right \\
}
{
\TabularMacro\ResumeTabular
left & right \\
\endtabular
}
\StopDefiningTabulars
```

for HTML output:

```
\begin{warpHTML}

\newcommand*{\ResumeTabular}{%
\global\boolfalse{LWR@exitingtabular}%
\global\boolfalse{LWR@tabularmutemods}%
\LWR@getmynexttoken%
}

\end{warpHTML}
```

for PRINT output:

```
\begin{warpprint}

\newcommand*{\ResumeTabular}{}

\end{warpprint}
```

71.26 Checking for a new table cell

for HTML output:

```
\begin{warpHTML}
```

Bool `LWR@exitingtabular` When `\end` is found, turns off the next opening data tag.

```
\newbool{LWR@exitingtabular}
```

Bool `LWR@tabularmutemods` Mutes HTML output for `@`, `!`, < and >.

This is used while printing the final row to generate `\bottomrules`.

```
\newbool{LWR@tabularmutemods}
```

`\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:

```
\newcommand*{\LWR@tabledatacolumntag}%
{%
\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:

```
\global\let\LWR@mynextaction\LWR@tabledatasinglecolumntag%
```

If exiting the tabular:

```
\ifdefequal{\LWR@mynexttoken}{\end}%
    {\global\booltrue{LWR@exitingtabular}}{}%
```

longtable can have a caption in a cell

```
\ifdefequal{\LWR@mynexttoken}{\caption}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%
```

Look for other things which would not start a table cell:

```
\ifdefequal{\LWR@mynexttoken}{\multicolumn}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%
\ifdefequal{\LWR@mynexttoken}{\multirow}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%
\ifdefequal{\LWR@mynexttoken}{\multicolumnrow}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%
\ifdefequal{\LWR@mynexttoken}{\noalign}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%
```

If an \mrowcell, this is a cell to be skipped over:

```
\ifdefequal{\LWR@mynexttoken}{\mrowcell}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%
```

If an \mcolrowcell, this is a cell to be skipped over:

```
\ifdefequal{\LWR@mynexttoken}{\mcolrowcell}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%

\ifdefequal{\LWR@mynexttoken}{\TabularMacro}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%

\ifdefequal{\LWR@mynexttoken}{\hline}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%

\ifdefequal{\LWR@mynexttoken}{\firsthline}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%

\ifdefequal{\LWR@mynexttoken}{\lasthline}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%

\ifdefequal{\LWR@mynexttoken}{\toprule}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%
```

```
\ifdefequal{\LWR@mynexttoken}{\midrule}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%

\ifdefequal{\LWR@mynexttoken}{\cmidrule}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%

\ifdefequal{\LWR@mynexttoken}{\morecmidrules}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%

\ifdefequal{\LWR@mynexttoken}{\specialrule}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%

\ifdefequal{\LWR@mynexttoken}{\cline}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%

\ifdefequal{\LWR@mynexttoken}{\bottomrule}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%

\ifdefequal{\LWR@mynexttoken}{\rowcolor}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%

\ifdefequal{\LWR@mynexttoken}{\arrayrulecolor}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%

\ifdefequal{\LWR@mynexttoken}{\doublerulesepcolor}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%

\ifdefequal{\LWR@mynexttoken}{\warpprintonly}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%

\ifdefequal{\LWR@mynexttoken}{\warpHTMLonly}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%

\ifdefequal{\LWR@mynexttoken}{\ldelim}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%

\ifdefequal{\LWR@mynexttoken}{\rdelim}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%
```

For arydshln:

```
\ifdefequal{\LWR@mynexttoken}{\hdashline}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%

\ifdefequal{\LWR@mynexttoken}{\cdashline}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%

\ifdefequal{\LWR@mynexttoken}{\firsthdashline}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%

\ifdefequal{\LWR@mynexttoken}{\lasthdashline}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%
```

Ignore an empty line between rows:

```
\ifdefequal{\LWR@mynexttoken}{\par}%
    {\global\let\LWR@mynextaction\LWR@donothing}{}%
```

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:

```
\LWR@traceinfo{LWR@tabledatacolumntag: about to do mynext}%
\LWR@mynextaction%
\LWR@traceinfo{LWR@tabledatacolumntag: done}%
}
```

```
\end{warpHTML}
```

71.27 \mrowcell

for HTML & PRINT:

```
\begin{warpall}
```

\mrowcell
⚠ multirow cells

The user must insert \mrowcell into any \multirow cells which must be skipped. This command has no action during print output.

```
\newcommand*{\mrowcell}{}
```

```
\end{warpall}
```

71.28 \mcolrowcell

for HTML & PRINT:

```
\begin{warpall}
```

\mcolrowcell
⚠ multirow cells

The user must insert \mcolrowcell into any \multicolumnrow cells which must be skipped. This command has no action during print output.

```
\newcommand*{\mcolrowcell}{}
```

```
\end{warpall}
```

71.29 HTML tabular environment

for HTML output:

```
\begin{warpHTML}
```

These are default defininitions in case booktabs is not loaded, and are not expected to used, but must exist as placeholders. They are pre-deleted in case memoir has already loaded booktabs.

```
\LetLtxMacro\toprule\relax
\LetLtxMacro\midrule\relax
\LetLtxMacro\cmidrule\cline
\LetLtxMacro\bottomrule\relax
\LetLtxMacro\addlinespace\relax
\LetLtxMacro\morecmidrules\relax
\LetLtxMacro\specialrule\relax

\newcommand*{\toprule}[1][]{\hline}
\newcommand*{\midrule}[1][]{\hline}
\LetLtxMacro\cmidrule\cline
\newcommand*{\bottomrule}[1][]{\hline}
\newcommand*{\addlinespace}[1][]{}
\newcommand*{\morecmidrules}{}
\newcommand*{\specialrule}[3]{\hline}
```

`\noalign` {⟨*text*⟩} Redefined for use inside `tabular`.

```
\LetLtxMacro\LWR@orignoalign\noalign

\newcommand{\LWR@tabularnoalign}[1]{%
\begingroup%
\global\advance\rownum\m@ne%
\renewcommand*{\LWR@xcolorrowHTMLcolor}{}%
\multicolumn{\value{LWR@tabletotalLaTeXcols}}{l}{#1} \\
\endgroup%
% \@rowc@lors%
\LWR@getmynexttoken%
}
```

`\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.

```
\AtBeginDocument{
\@ifpackageloaded{lwarp-tabls}
{
\newcommand*{\LWR@HTMLhline}[1][]{%
    \ifbool{FormatWP}%
    {\LWR@docmidrule{1-\arabic{LWR@tabletotalLaTeXcols}}}%
    {\addtocounter{LWR@hlines}{1}}%
    \LWR@getmynexttoken}%
}
{
\newcommand*{\LWR@HTMLhline}{%
    \ifbool{FormatWP}%
    {\LWR@docmidrule{1-\arabic{LWR@tabletotalLaTeXcols}}}%
    {\addtocounter{LWR@hlines}{1}}%
    \LWR@getmynexttoken}%
}
}% AtBeginDocument
```

\LWR@HTMLcline {⟨*columns*⟩}

```
\NewDocumentCommand{\LWR@HTMLcline}{m}%
{\LWR@docmidrule{#1}\LWR@getmynexttoken}%
```

\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 column parsing code, which uses xstring.

```
\AtBeginDocument{
\@ifundefined{frenchbsetup}%
{% no babel-french
    \newcommand*{\LWR@nullifyNoAutoSpacing}{}
}% no babel-french
{% yes babel-french
    \newcommand*{\LWR@nullifyNoAutoSpacing}{%
        \NoAutoSpacing%
        \renewcommand*{\NoAutoSpacing}{}%
        \renewcommand*{\LWR@FBcancel}{}%
    }
}% yes babel-french
}% AtBeginDocument
```

Env tabular <direction> [⟨*vertposition*⟩] {⟨*colspecs*⟩}

The <direction> is from plext for Japanese documents, and is ignored.

```
\StartDefiningTabulars

\NewDocumentCommand{\LWR@HTML@@tabular}{d<> o m}
{%
\LWR@traceinfo{LWR@HTML@@tabular started}%
\addtocounter{LWR@tabulardepth}{1}%
```

Not yet started a table row:

```
\global\boolfalse{LWR@startedrow}%
```

Not yet doing any rules:

```
\setcounter{LWR@hlines}{0}%
\setcounter{LWR@hdashedlines}{0}%
\global\boolfalse{LWR@doingtbrule}%
\global\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.

```
\LWR@nullifyNoAutoSpacing%
```

Have not yet found the end of tabular command. Unmute the @ and ! columns.

```
\global\boolfalse{LWR@exitingtabular}%
\global\boolfalse{LWR@tabularmutemods}%
```

Create the `table` tag:

```
\global\booltrue{LWR@intabularmetadata}%
\LWR@traceinfo{LWR@@tabular: About to LWR@forecenewpage.}%
\LWR@forcenewpage
\LWR@htmlblocktag{table}%
```

Parse the table columns:

```
\LWR@parsetablecols{#3}%
```

Table col spec is: `\LWR@tablecolspec` which is a string of `llccrr`, etc.

Do not place the table inside a paragraph:

```
\LWR@stoppars%
```

Track column #:

```
\setcounter{LWR@tableLaTeXcolindex}{1}%
```

Have not yet added data in this column:

```
\global\boolfalse{LWR@tabularcelladded}%
```

Start looking for midrules:

```
\LWR@clearmidrules%
```

`\\` becomes a macro to end the table row:

```
\LetLtxMacro{\\}{\LWR@tabularendofline}%
```

The following adjust for colortbl.

```
\LetLtxMacro\arrayrulecolor\arrayrulecolornexttoken%
\LetLtxMacro\doublerulesepcolor\doublerulesepcolornexttoken%
\gdef\LWR@columnHTMLcolor{}%
\gdef\LWR@rowHTMLcolor{}%
\gdef\LWR@cellHTMLcolor{}%
\@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.

```
\ifdefvoid{\LWR@ruleHTMLcolor}%
    {\edef\LWR@vertruleHTMLcolor{black}}%
    {\edef\LWR@vertruleHTMLcolor{\LWR@origpound\LWR@ruleHTMLcolor}}%
```

Tracking the depth of cell color `<div>`s:

```
\setcounter{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.

```
\LWR@traceinfo{LWR@@HTML@tabular: redefining macros}%
\LetLtxMacro\noalign\LWR@tabularnoalign%
\LetLtxMacro\hline\LWR@HTMLhline%
\LetLtxMacro\cline\LWR@HTMLcline%

\DeclareDocumentCommand{\hdashline}{o}{%
    \ifbool{FormatWP}%
        {\LWR@docdashline{1-\arabic{LWR@tabletotalLaTeXcols}}}%
        {\addtocounter{LWR@hdashedlines}{1}}%
    \LWR@getmynexttoken%
}%

\DeclareDocumentCommand{\cdashline}{m}{%
    \LWR@docdashline{##1}\LWR@getmynexttoken%
}%

\DeclareDocumentCommand{\firsthdashline}{o}{%
    \ifbool{FormatWP}%
        {\LWR@docdashline{1-\arabic{LWR@tabletotalLaTeXcols}}}%
        {\addtocounter{LWR@hdashedlines}{1}}%
    \LWR@getmynexttoken%
}%

\DeclareDocumentCommand{\lasthdashline}{o}{%
    \ifbool{FormatWP}%
        {\LWR@docdashline{1-\arabic{LWR@tabletotalLaTeXcols}}}%
        {\addtocounter{LWR@hdashedlines}{1}}%
    \LWR@getmynexttoken%
}%
```

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`.

```
\renewcommand{\multicolumn}{\LWR@htmlmulticolumn}%
\renewcommand*{\mrowcell}{%
    \LWR@maybenewtablerow%
    \LWR@tabularleftedge%
    \global\booltrue{LWR@skippingmrowcell}%
}%
\renewcommand*{\mcolrowcell}{%
    \LWR@maybenewtablerow%
    \global\booltrue{LWR@skippingmcolrowcell}%
}%
\LetLtxMacro\caption\LWR@longtabledatacaptiontag%
```

Reset for new processing:

```
\global\boolfalse{LWR@tableparcell}%
\global\boolfalse{LWR@skippingmrowcell}%
\global\boolfalse{LWR@skippingmcolrowcell}%
\global\boolfalse{LWR@skipatbang}%
\global\boolfalse{LWR@emptyatbang}%
```

Set & for its special meaning inside the tabular:

```
\StartDefiningTabulars%
\protected\gdef&{\LWR@tabularampersand}%
```

Locally force any minipages to be fullwidth, until the end of the tabular:

```
\booltrue{LWR@forceminipagefullwidth}
```

Nest one level deeper of tabular paragraph handling:

```
\addtocounter{LWR@tabularpardepth}{1}%
```

Look ahead for a possible table data cell:

```
\LWR@traceinfo{LWR@@HTML@tabular: about to LWR@getmynexttoken}%
\LWR@getmynexttoken%
}%
```

Ending the environment:

```
\newcommand*{\LWR@HTML@endtabular}
{%
\LWR@traceinfo{LWR@HTML@endtabular}%
```

Unnest one level of tabular paragraph handling:

```
\addtocounter{LWR@tabularpardepth}{-1}%
\ifboolexpr{%
    test {%
      \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{<}{\value{LWR@tabletotalLaTeXcols}}
    } or %
    (%
        bool{LWR@intabularmetadata} and%
        not bool{LWR@tabularcelladded} and%
        test {%
        \ifnumcomp{\value{LWR@tableLaTeXcolindex}}{=}{\value{LWR@tabletotalLaTeXcols}}%
        }%
    )%
}%
{%
    \LWR@tabularfinishrow%
}%
{%
    \LWR@closetabledatacell%
}%
```

```
\LWR@htmlblocktag{/tr}%
```

xcolor row color support:

```
\@rowc@lors%

\LWR@htmlblocktag{/table}%
\global\boolfalse{LWR@intabularmetadata}%
```

Unnest one level of tabular:

```
\addtocounter{LWR@tabulardepth}{-1}%
```

Restore & to its usual meaning:

```
\protected\gdef&{\LWR@origampmacro}%
\StopDefiningTabulars%
\LWR@traceinfo{LWR@HTML@endtabular finished}%
}

\csletcs{LWR@HTML@endtabular*}{LWR@HTML@endtabular}

\StopDefiningTabulars
```

siunitx may redefine `tabular`, so set the following later:

```
\AtBeginDocument{
    \LetLtxMacro\LWR@origendtabular\endtabular
    \csletcs{LWR@origendtabular*}{endtabular*}
    \LWR@formatted{@tabular}
    \LWR@formatted{endtabular}
    \LWR@formatted{endtabular*}
}

\end{warpHTML}
```

72 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.

File `*_html.aux` A new entry in `*_html.aux` is used to remember section name, file, and latexmage depth and number for each label:

```
\newlabel{<labelname>@lwarp}{{<section name>}{<filename>}
                                    {<limagedepth>}{<limagenumber>}}
```

Table 12 shows the data structures related to cross-referencing.

for HTML output:

```
\begin{warpHTML}
```

Table 12: 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 72.3), and another .aux entry (section 72.2).

\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 name 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 72.4.

\ref and \nameref: Adds HTML tags. See section 72.4.

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 58.4 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 86.

\label inside math: See section 78.7.1.

Footnotes: See \noteentry in section 58.4.

72.1 Setup

\@currentlabelname To remember the most recently defined section name, description, or caption, for \nameref.

```
\providecommand*{\@currentlabelname}{}
```

\LWR@stripperiod {⟨*text*⟩} [⟨.⟩]

Removes a trailing period.

```
\def\LWR@stripperiod#1.\ltx@empty#2\@nil{#1}%
```

\LWR@setlatestname {⟨*object name*⟩}

Removes \label, strips any final period, and remembers the result.

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

Remove \label and other commands from the name, the strip any final period. See gettitlestring.

```
\GetTitleStringExpand{#1}%
\edef\@currentlabelname{\detokenize\expandafter{\GetTitleStringResult}}%
\edef\@currentlabelname{%
    \expandafter\LWR@stripperiod\@currentlabelname%
    \ltx@empty.\ltx@empty\@nil%
}%
}
```

72.2 New lwarp labels.

File *_html.aux A new entry in *_html.aux is used to remember section name, file, and latexiimage 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

\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.

```
\def\LWR@setref#1#2#3{%
```

```
  \ifx#1\relax%
    ??%
  \else%
   \expandafter#2#1%
  \fi}
```

`\LWR@nameref` {⟨*label*⟩} Returns the section name for this label:

```
\newcommand*{\LWR@nameref}[1]{%
    \expandafter\LWR@setref\csname r@#1@lwarp\endcsname\LWR@firstoffour{#1}%
}
```

`\LWR@htmlfileref` {⟨*label*⟩} Returns the file number or name for this label:

```
\newcommand*{\LWR@htmlfileref}[1]{%
    \expandafter\LWR@setref\csname r@#1@lwarp\endcsname\LWR@secondoffour{#1}%
}
```

`\LWR@lateximagedepthref` {⟨*label*⟩} Returns the `lateximagedepth` for this label:

```
\newcommand*{\LWR@lateximagedepthref}[1]{%
    \expandafter\LWR@setref\csname r@#1@lwarp\endcsname\LWR@thirdoffour{#1}%
}
```

`\LWR@lateximagenumberref` {⟨*label*⟩} Returns the `lateximagenumber` for this label:

```
\newcommand*{\LWR@lateximagenumberref}[1]{%
    \expandafter\LWR@setref\csname r@#1@lwarp\endcsname\LWR@fourthoffour{#1}%
}
```

`\LWR@lwarplabel` {⟨*label*⟩} Sanitize the name and then creates the label:

```
\newcommand*{\LWR@lwarplabel}[1]{%
\LWR@traceinfo{LWR@lwarplabel !#1!}%
\LWR@setlatestname{\@currentlabelname}%
    \@bsphack%
    \protected@write\@auxout{}%
        {\string\newlabel{#1@lwarp}{%
            {\@currentlabelname}%
        {\ifbool{FileSectionNames}{\LWR@thisfilename}{\arabic{LWR@htmlfilenumber}}}%
            {\arabic{LWR@lateximagedepth}}%
            {\arabic{LWR@lateximagenumber}}%
        }}%
    \@esphack%
}
```

72.3 Labels

`\LWR@sublabel` {⟨*label*⟩} Creates an HTML id tag.

\detokenize is used to allow underscores in the labels.

```
\newcommand*{\LWR@sublabel}[1]{%
\LWR@traceinfo{LWR@sublabel !#1!}%
```

Create an HTML id tag unless are inside a lateximage, since it would appear in the image:

```
\ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
{}%
{% not lateximage
```

If not doing a lateximage, create an HTML ID tag: (To be factored...)

```
    \LWR@sanitize{#1}%
    \ifbool{LWR@doingstartpars}%
    {% pars allowed
        \ifbool{LWR@doingapar}%
        {% par started
         \LWR@htmltag{a \LWR@print@mbox{id="\LWR@sanitized"}}\LWR@htmltag{/a}%
        }% par started
        {% par not started
            \LWR@stoppars%
         \LWR@htmltag{a \LWR@print@mbox{id="\LWR@sanitized"}}\LWR@htmltag{/a}%
            \LWR@startpars%
        }% par not started
    }% pars allowed
    {% pars not allowed
        \LWR@htmltag{a \LWR@print@mbox{id="\LWR@sanitized"}}\LWR@htmltag{/a}%
    }% pars not allowed
}% not lateximage
}
```

\LWR@new@label (⟨*bookmark*⟩) {⟨*label*⟩} [⟨*type*⟩]

\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@sublabel does not generate HTML tags inside a lateximage.

clevereref later encases this to add its own cross-referencing.

The optional ⟨*bookmark*⟩ is per the memoir class, and is ignored.

The optional ⟨*type*⟩ is per the ntheorem package, and is ignored.

```
\NewDocumentCommand{\LWR@new@label}{d() m o}{%
\LWR@traceinfo{LWR@new@label: starting}%
\LWR@traceinfo{LWR@new@label: !#2!}%
% \@bsphack%
```

Create a traditional LaTeX label, as modified by cleveref:

```
\LWR@orig@label{#2}%
```

Create a special label which holds the section number, LWR@htmlfilenumber, LWR@lateximagedepth, and LWR@lateximagenumber:

```
\LWR@traceinfo{LWR@new@label: filesectionnames is \ifbool{FileSectionNames}{true}{false}}%
\LWR@traceinfo{LWR@new@label: LWR@thisfilename is !\LWR@thisfilename!}%
\LWR@traceinfo{LWR@new@label: LWR@htmlfilenumber is \arabic{LWR@htmlfilenumber}}%
\LWR@lwarplabel{#2}%
\LWR@sublabel{#2}%
% \@esphack%
\LWR@traceinfo{LWR@new@label: done}%
}
```

72.4 References

`\LWR@startref` {⟨*label*⟩} (Common code for \ref and \nameref.)

Open an HTML tag reference to a filename, # character, and a label.

```
\newcommand*{\LWR@startref}[1]
{%
\LWR@sanitize{#1}%
\LWR@traceinfo{LWR@startref A: !#1!}%
```

Create the filename part of the link:

```
\LWR@htmltag{a href="%
\LWR@traceinfo{LWR@startref B}%
\LWR@print@mbox{\LWR@htmlrefsectionfilename{#1}}%
\LWR@traceinfo{LWR@startref C}%
\LWR@origpound%
```

Create the destination id:

See if LWR@lateximagedepth is unknown:

```
\LWR@traceinfo{LWR@startref D: !#1!}%
\ifcsundef{r@#1@lwarp}%
```

"??" if LWR@lateximagedepth is unknown, so create a link with an unknown destination:

```
{%
    \LWR@traceinfo{LWR@startref D0: ??}%
    ??%
}%
```

If LWR@lateximagedepth is known. Use a lateximage if the depth is greater than zero, or a regular link otherwise:

```
{%
    \ifthenelse{\cnttest{\LWR@lateximagedepthref{#1}}{>}{0}}%
        {%
            \LWR@ImagesName\LWR@lateximagenumberref{#1}%
        }%
        {%
            \LWR@traceinfo{LWR@startref D3}%
```

\detokenize is used to allow underscores in the labels:

```
            \LWR@print@mbox{\LWR@sanitized}%
        }%
}%
\LWR@traceinfo{LWR@startref E}%
```

Closing quote:

```
"}%
\LWR@traceinfo{LWR@startref F}%
}
```

\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)".

```
\NewDocumentCommand{\LWR@subnewref}{m m}{%
\LWR@traceinfo{LWR@subnewref #1 #2}%
\LWR@startref{#1}%
\LWR@print@ref{#2}%
\LWR@htmltag{/a}%
}
```

\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.

```
\NewDocumentCommand{\LWR@HTML@ref}{s m}{%
\LWR@traceinfo{LWR@HTML@ref !#2!}%
\IfBooleanTF{#1}%
{\LWR@print@ref{#2}}%
{\LWR@subnewref{#2}{#2}}%
}

\LWR@formatted{ref}
```

\LWR@ref@ignorestar * {⟨*label*⟩} For use inside \hyperref. Ignores the star, then uses the original \ref.

```
\NewDocumentCommand{\LWR@ref@ignorestar}{s m}{%
    \LWR@print@ref{#2}%
}
```

\pagerefPageFor Text for page references.

```
\newcommand*{\pagerefPageFor}{see }
```

\pageref * {⟨*label*⟩} Create an internal document reference, or just the unlinked number if starred, per hyperref.

```
\NewDocumentCommand{\LWR@new@pageref}{s m}{%
\IfBooleanTF{#1}%
{(\pagerefPageFor\LWR@print@ref{#2})}%
{(\cpageref{#2})}%
}
```

\nameref {⟨*label*⟩}

```
\newrobustcmd*{\nameref}[1]{%
\LWR@traceinfo{nameref}%
\LWR@startref{#1}%
\LWR@traceinfo{nameref B}%
\LWR@nameref{#1}%
\LWR@traceinfo{nameref C}%
\LWR@htmltag{/a}%
\LWR@traceinfo{nameref: done}%
}
```

\Nameref {⟨*label*⟩} In print, adds the page number. In HTML, does not.

```
\LetLtxMacro\Nameref\nameref
```

72.5 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.

Pkg hyperref

⚠ 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.

```
% DO NOT TELL OTHER PACKAGES TO ASSUME HYPERREF, lest they attempt to patch it:
% \EmulatesPackage{hyperref}[2015/08/01]% Disabled.  Do not do this.
```

Emulates hyperref:

\@currentHref Added to support backref.

```
\AtBeginDocument{
\def\@currentHref{%
autopage-\theLWR@currentautosec%
}
}
```

`\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.

```
\NewDocumentCommand{\LWR@subhyperref}{m}{%
\LWR@traceinfo{LWR@subhyperref !#1!}%
\LWR@sanitize{#1}%
\LWR@htmltag{%
    a href="\LWR@sanitized" % space
    target="\_{}blank" % space
}%
}
```

`\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.

```
\newcommand{\LWR@subhyperreftext}[1]{%
#1%
\LWR@htmltag{/a}%
\LWR@ensuredoingapar%
}
```

`\LWR@subhyperrefclass` {⟨*URL*⟩} {⟨*text*⟩} {⟨*htmlclass*⟩}

```
\NewDocumentCommand{\LWR@subhyperrefclass}{m +m m}{%
\LWR@htmltag{%
    a % space
    href="\begingroup\@sanitize#1\endgroup" % space
    class="#3" % space
}\LWR@orignewline%
#2\LWR@orignewline%
\LWR@htmltag{/a}%
\LWR@ensuredoingapar%
}
```

`\href` [⟨*options*⟩] {⟨*URL*⟩}

Create a link with accompanying text:

```
\DeclareDocumentCommand{\LWR@hrefb}{O{} m}{%
\LWR@ensuredoingapar%
\LWR@subhyperref{#2}%
```

```
\endgroup% restore catcodes
\LWR@subhyperreftext%
}

\newrobustcmd*{\href}{%
\begingroup%
\catcode`\#=12%
\catcode`\%=12%
\catcode`\&=12%
\catcode`\~=12%
\catcode`\_=12%
\LWR@hrefb%
}
```

\nolinkurl {⟨*URL*⟩}

Print the name of the link without creating the link:

```
\newcommand*{\LWR@nolinkurlb}[1]{%
\LWR@ensuredoingapar%
\def\LWR@templink{#1}%
\@onelevel@sanitize\LWR@templink%
\LWR@templink%
\endgroup%
}

\newrobustcmd*{\nolinkurl}{%
\begingroup%
\catcode`\#=12%
\catcode`\%=12%
\catcode`\&=12%
\catcode`\~=12%
\catcode`\_=12%
\LWR@nolinkurlb%
}
```

\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@urla`here and also redefined by lwarp-url.

```
\DeclareDocumentCommand{\LWR@urlb}{m}{%
\LWR@ensuredoingapar%
\def\LWR@templink{#1}%
\@onelevel@sanitize\LWR@templink%
\href{\LWR@templink}{\LWR@templink}%
\endgroup%
}

\newrobustcmd*{\url}{%
\begingroup%
\catcode`\#=12%
```

```
\catcode`\%=12%
\catcode`\&=12%
\catcode`\~=12%
\catcode`\_=12%
\LWR@urlb%
}
```

`\LWR@subinlineimage` [⟨*alttag*⟩] {⟨*class*⟩} {⟨*filename*⟩} {⟨*extension*⟩} {⟨*style*⟩}

```
\newcommand*{\LWR@subinlineimage}[5][]{%
\ifblank{#1}%
{%
    \LWR@htmltag{img \LWR@indentHTML
        src="#3.#4" \LWR@indentHTML
        alt="#3" \LWR@indentHTML
        style="#5" \LWR@indentHTML
        class="#2" \LWR@orignewline
    }%
}%
{%
    \LWR@htmltag{img \LWR@indentHTML
        src="#3.#4" \LWR@indentHTML
        alt="#1" \LWR@indentHTML
        style="#5" \LWR@indentHTML
        class="#2" \LWR@orignewline
    }%
}%
}
```

```
\end{warpHTML}
```

Table 13: 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.

73 Floats

Floats are supported, although partially through emulation.

Table 13 shows the data structure associated with each <type> of float.

`\@makecaption` is redefined to print the float number and caption text, separated by `\CaptionSeparator`, which works with the babel package to adjust the caption separator according to the language. French, for example, uses an en-dash instead of a colon: "Figure 123 – Caption text".

73.1 Float environment

for HTML output: 8648 `\begin{warpHTML}`

`\LWR@floatbegin` {⟨*type*⟩} [⟨*placement*⟩] Begins a `\newfloat` environment.

```
\NewDocumentCommand{\LWR@floatbegin}{m o}{%
\ifbool{FormatWP}{\newline}{}%
\LWR@stoppars
```

There is a new float, so increment the unique float counter:

```
\addtocounter{LWR@thisautoid}{1}%
\booltrue{LWR@freezethisautoid}%

\begingroup%
```

Settings while inside the environment:

```
\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.

```
\LWR@htmltag{%
    figure id="\LWR@print@mbox{autoid-\arabic{LWR@thisautoid}}" % space
    class="#1 \@nameuse{LWR@floatstyle@#1}"%
}%
\ifbool{FormatWP}{%
    \LWR@orignewline%
    \LWR@BlockClassWP{}{}{wp#1}%
}{}%
```

Update the caption type:

```
\renewcommand*{\@captype}{#1}%
\caption@settype{#1}%
```

Mark the float for a word processor conversion:

```
\LWR@startpars%
\ifboolexpr{bool{FormatWP} and bool{WPMarkFloats}}{%

=== begin #1 ===

}{}%
```

Look for `\centering`, etc:

```
\LWR@futurenonspacelet\LWR@mynexttoken\LWR@floatalignment%
}
```

For koma-script. The following does not work for tables.

```
\AtBeginDocument{
\@ifpackageloaded{tocbasic}{
\appto\figure@atbegin{%
```

```
\LWR@futurenonspacelet\LWR@mynexttoken\LWR@floatalignment%
}
}{}
}
```

\@float
\@dlbfloat

Support packages which create floats directly.

```
\let\@float\LWR@floatbegin
\let\@dblfloat\LWR@floatbegin
```

\LWR@floatend

Ends a \newfloat environment.

```
\newcommand*{\LWR@floatend}{%
```

If saw a \centering, finish the center environment:

```
\LWR@endfloatalignment%
```

Mark the float end for a word processor conversion:

```
\ifboolexpr{bool{FormatWP} and bool{WPMarkFloats}}{%

=== end ===

}{}%
\LWR@stoppars%
```

Close an HTML figure tag:

```
\ifbool{FormatWP}{\endLWR@BlockClassWP}{}%
\LWR@htmlelementend{figure}%
\endgroup%
\boolfalse{LWR@freezethisautoid}%
\LWR@startpars%
\ifbool{FormatWP}{\newline}{}%
}
```

\end@float
\end@dlbfloat

Support packages which create floats directly.

```
\let\end@float\LWR@floatend
\let\end@dblfloat\LWR@floatend
```

73.2 Float tracking

Ctr LWR@thisautoid

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.

```
\newcounter{LWR@thisautoid}
```

Ctr `LWR@thisautoidWP` A sequential counter for all word processor conversion s. This is used to convince LibreOffice to form a frame around this element.

```
\newcounter{LWR@thisautoidWP}
```

Bool `LWR@freezethisautoid` Prevents multiple increments of `\LWR@thisautoid` inside a float.

```
\newbool{LWR@freezethisautoid}
\boolfalse{LWR@freezethisautoid}
```

`\LWR@newautoidanchor` Adds a new <autoid> anchor.

```
\newcommand*{\LWR@newautoidanchor}{%
\ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
{}%
{
    \ifbool{LWR@freezethisautoid}{}{%
        \addtocounter{LWR@thisautoid}{1}%
        \LWR@htmltag{a id="\LWR@print@mbox{autoid-\arabic{LWR@thisautoid}}"}%
            \LWR@htmltag{/a}%
    }%
}
}
```

`\@captype` Remembers which float type is in use.

```
\newcommand*{\@captype}{}
```

`\LWR@floatalignmentname` Set to `center`, `flushleft`, or `flushright` if saw `\centering`, `\raggedright`, or `\raggedleft`.

```
\newcommand*{\LWR@floatalignmentname}{}
```

`\LWR@floatalignment` If sees a `\centering`, `\raggedleft`, or `\raggedright`, creates a `center`, `flushright`, or `flushleft` environment.

```
\newcommand*{\LWR@floatalignment}{%
\ifdefstrequal{\LWR@mynexttoken}{\centering}{%
    \center%
    \renewcommand*{\LWR@floatalignmentname}{center}%
}{}%
\ifdefstrequal{\LWR@mynexttoken}{\raggedright}{%
    \flushleft%
    \renewcommand*{\LWR@floatalignmentname}{flushleft}%
}{}%
\ifdefstrequal{\LWR@mynexttoken}{\raggedleft}{%
    \flushright%
    \renewcommand*{\LWR@floatalignmentname}{flushright}%
}{}%
}
```

`\LWR@endfloatalignment` Closes an environment from `\LWR@floatalignment`.

```
\newcommand*{\LWR@endfloatalignment}{%
\ifdefvoid{\LWR@floatalignmentname}{}{\@nameuse{end\LWR@floatalignmentname}}%
\renewcommand*{\LWR@floatalignmentname}{}%
}
```

73.3 Caption inside a float environment

\CaptionSeparator How to separate the float number and the caption text.

```
\AtBeginDocument{\providecommand*{\CaptionSeparator}{:~}}
```

\@makecaption {⟨*name and num*⟩} {⟨*text*⟩}

Prints the float type and number, the caption separator, and the caption text.

```
\AtBeginDocument{\renewcommand{\@makecaption}[2]{%
    \LWR@traceinfo{@makecaption}%
    \LWR@isolate{#1}\CaptionSeparator\LWR@isolate{#2}%
    \LWR@traceinfo{@makecaption: done}%
}%
}
```

73.4 Caption and LOF linking and tracking

When a new HTML file is marked in the LaTeX PDF file, the LaTeX page number at that point is stored in LWR@latestautopage, (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. 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.

Ctr LWR@nextautoid Tracks autoid for floats. Tracks autopage for floats.

Ctr LWR@nextautopage These are updated per float as the .lof, .lot file is read.

```
\newcounter{LWR@nextautoid}
\newcounter{LWR@nextautopage}
```

\LWRsetnextfloat {⟨*autopage*⟩} {⟨*float autoid*⟩}

File *_html.lof This is written to the *_html.lof or *_html.lot file just before each float's usual
File *_html.lot entry. The autopage and the float's autoid are remembered for \l@figure to use when creating the HTML links.

```
\newcommand*{\LWRsetnextfloat}[2]{%
```

```
    \setcounter{LWR@nextautopage}{#1}%
    \setcounter{LWR@nextautoid}{#2}%
}
```

Ctr LWR@latestautopage

Updated each time a new HTML file is begun. \LWRsetnextfloat is written with this and the autoid by the modified \addcontentsline just before each float's entry.

```
\newcounter{LWR@latestautopage}
\setcounter{LWR@latestautopage}{1}
```

Env LWR@figcaption

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 of class figurecaption is used instead.

```
\newenvironment*{LWR@figcaption}
    {
        \ifbool{FormatWP}{%
            \BlockClass[font-style:italic]{figurecaption}
%             \LWR@print@vspace*{\baselineskip}
        }{
            \BlockClass{figurecaption}
        }%
    }
    {\endBlockClass}
```

\LWR@HTML@caption@begin {⟨*type*⟩}

Low-level code to create HTML tags for captions.

The print versions are from the caption package.

```
\newcommand*{\LWR@HTML@caption@begin}[1]
{
\LWR@traceinfo{LWR@HTML@caption@begin}%
```

Keep par and minipage changes local:

```
\begingroup%
```

The caption code was not allowing the closing par tag:

```
\@setpar{\LWR@closeparagraph\@@par}%
```

No need for a minipage or \parbox inside the caption:

```
\RenewDocumentEnvironment{minipage}{O{t} o O{t} m}{}{}%
\RenewDocumentCommand{\parbox}{O{t} O{} O{t} m +m}{##5}%
```

Enclose the original caption code inside an HTML tag:

```
\LWR@figcaption%
\LWR@traceinfo{LWR@HTML@caption@begin: about to LWR@origcaption@begin}%
```

```
\LWR@print@caption@begin{#1}%
\LWR@traceinfo{LWR@HTML@caption@begin: done}%
}
```

\LWR@HTML@caption@end Low-level patches to create HTML tags for captions.

```
\newcommand*{\LWR@HTML@caption@end}
{%
\LWR@traceinfo{LWR@HTML@caption@end}%
\LWR@print@caption@end%
```

Closing tag:

```
\endLWR@figcaption%
\endgroup%
% \leavevmode% avoid bad space factor (0) error
\LWR@traceinfo{LWR@HTML@caption@end: done}%
}
```

\caption@begin
\caption@end Low-level patches to create HTML tags for captions. These are assigned \AtBeginDocument so that other packages which modify captions will have already been loaded before saving the print-mode version.

```
\AtBeginDocument{
\LWR@formatted{caption@begin}
\LWR@formatted{caption@end}
}
```

\captionlistentry Tracks the float number for this caption used outside a float. Patched to create an HTML anchor.

```
\let\LWR@origcaptionlistentry\captionlistentry

\renewcommand*{\captionlistentry}{%
\LWR@ensuredoingapar%
\LWR@origcaptionlistentry%
}

\def\LWR@LTcaptionlistentry{%
\LWR@ensuredoingapar%
\LWR@htmltag{a id="\LWR@print@mbox{autoid-\arabic{LWR@thisautoid}}"}\LWR@htmltag{/a}%
  \bgroup
  \@ifstar{\egroup\LWR@LT@captionlistentry}% gobble *
          {\egroup\LWR@LT@captionlistentry}}%
\def\LWR@LT@captionlistentry#1{%
  \caption@listentry\@firstoftwo[\LTcaptype]{#1}}%
```

\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

```
\let\LWR@origaddcontentsline\addcontentsline

\renewcommand*{\addcontentsline}[3]{%
\ifstrequal{#1}{toc}{}{% not TOC

\ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
    {}%
    {\LWR@newautoidanchor}%

    \ifcsvoid{ext@#2}{\csdef{ext@#2}{#1}}{}%

    \addtocontents{\@nameuse{ext@#2}}{%
        \protect\LWRsetnextfloat%
        {\arabic{LWR@latestautopage}}%
        {\arabic{LWR@thisautoid}}%
    }%
}% not TOC
\LWR@origaddcontentsline{#1}{#2}{#3}%
}
```

Pkg capt-of
Pkg caption

Either package provides `\captionof`, which is later patched at the beginning of the document.

\captionof

Patched to handle paragraph tags.

```
\AtBeginDocument{
\let\LWR@origcaptionof\captionof

\renewcommand*{\captionof}{%
\LWR@stoppars
\LWR@origcaptionof%
}
}

\end{warpHTML}
```

74 Table of Contents, LOF, LOT

This section controls the generation of the TOC, LOF, and LOT.

The `.toc`, `.lof`, and `.lot` files are named by the source code `\jobname`.

In HTML, the printed tables are placed inside a `<div>` of class `toc`, `lof`, or `lot`.

A "sidetoc" is provided which prints a subset of the TOC on the side of each page other than the homepage.

The regular LaTeX infrastructure is used for TOC, along with some patches to generate HTML output.

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

74.1 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.

```
\newcommand*{\LWR@myshorttoc}[1]{%
\LWR@traceinfo{LWR@myshorttoc: #1}%
\LWR@ensuredoingapar%
```

Only if the file exists:

```
\IfFileExists{\jobname.#1}{%
\LWR@traceinfo{LWR@myshorttoc: loading}%
```

⚠ Many of the commands in the file will have `@` characters in them, so `@` must be made a regular letter.

```
\begingroup%
\makeatletter%
```

Read in the TOC file:

```
\@input{\jobname.#1}%
\endgroup%
}%
{}%
\LWR@traceinfo{LWR@myshorttoc: done}%
}
```

`\LWR@subtableofcontents` {⟨*toc/lof/lot*⟩} {⟨*sectionstarname*⟩}

Places a TOC/LOF/LOT at the current position.

```
\NewDocumentCommand{\LWR@subtableofcontents}{m m}{%
```

Closes previous levels:

```
\@ifundefined{chapter}
{\LWR@closeprevious{\LWR@depthsection}}
{\LWR@closeprevious{\LWR@depthchapter}}
```

Prints any pending footnotes so that they appear above the potentially large TOC:

```
\LWR@printpendingfootnotes
```

Place the list into its own chapter (if defined) or section:

```
\@ifundefined{chapter}{\section*{#2}}{\chapter*{#2}}
```

Create a new HTML nav containing the TOC/LOF/LOT:

```
\LWR@htmlelementclass{nav}{#1}
```

Create the actual list:

```
\LWR@myshorttoc{#1}
```

Close the nav:

```
\LWR@htmlelementclassend{nav}{#1}
}
```

`\@starttoc` {⟨*ext*⟩}

Patch `\@starttoc` to encapsulate the TOC inside HTML tags:

```
\let\LWR@orig@starttoc\@starttoc

\renewcommand{\@starttoc}[1]{
\LWR@htmlelementclass{nav}{#1}
\LWR@orig@starttoc{#1}
\LWR@htmlelementclassend{nav}{#1}
}
```

Bool `LWR@copiedsidetoc` 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.)

```
\newbool{LWR@copiedsidetoc}
\boolfalse{LWR@copiedsidetoc}
```

`\tableofcontents` Patch `\tableofcontents`, etc. to print footnotes first. newfloat uses `\listoffigures` for all future float types.

```
\AtBeginDocument{
\let\LWR@origtableofcontents\tableofcontents

\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:

```
\ifboolexpr{bool{FormatWP} and bool{WPMarkTOC}}{

=== table of contents ===

```

```
}
{
```

Copy the .toc file to .sidetoc for printing the sideTOC. The original .toc file is renewed when \tableofcontents is finished.

```
    \ifbool{LWR@copiedsidetoc}{}{%
        \LWR@copyfile{\jobname.toc}{\jobname.sidetoc}%
        \booltrue{LWR@copiedsidetoc}%
    }%
    \LWR@printpendingfootnotes
    \LWR@origtableofcontents
}
}% \tableofcontents
}% AtBeginDocument
```

\listoffigures

```
\let\LWR@origlistoffigures\listoffigures

\renewcommand*{\listoffigures}{
\ifboolexpr{bool{FormatWP} and bool{WPMarkLOFT}}{

=== list of figures ===

}
{
    \LWR@printpendingfootnotes
    \LWR@origlistoffigures
}
}
```

\listoftables

```
\let\LWR@origlistoftables\listoftables

\renewcommand*{\listoftables}{
\ifboolexpr{bool{FormatWP} and bool{WPMarkLOFT}}{

=== list of tables ===

}
{
    \LWR@printpendingfootnotes
    \LWR@origlistoftables
}
}
```

74.2 High-level TOC commands

\listof {⟨*type*⟩} {⟨*title*⟩}

Emulate the \listof command from the float package (section 222). Used to create lists of custom float types. Also used to redefine the standard LaTeX \listoffigures and \listoftables commands.

```
\NewDocumentCommand{\listof}{m +m}{%
\@ifundefined{l@#1}{%
    \csdef{l@#1}##1##2{\hypertocfloat{1}{#1}{\@nameuse{ext@#1}}{##1}{##2}}%
}{}%
\LWR@subtableofcontents{\@nameuse{ext@#1}}{#2}
\expandafter\newwrite\csname tf@\csname ext@#1\endcsname\endcsname
\immediate\openout \csname tf@\csname ext@#1\endcsname\endcsname
    \jobname.\@nameuse{ext@#1}\relax
}
```

74.3 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.

CSS may be used to format the sideTOC:

CSS related to sideTOC:

div.sidetoccontainer: The entire sideTOC.
div.sidetoctitle: The title.
div.sidetoccontents: The table of contents.

```
\end{warpHTML}
```

for HTML & PRINT:

```
\begin{warpall}
```

Ctr SideTOCDepth — Controls how deep the side-TOC gets. Use a standard LaTeX section level similar to tocdepth.

```
\newcounter{SideTOCDepth}
\setcounter{SideTOCDepth}{1}
```

\sidetocname — Holds the default name for the sideTOC.

```
\newcommand{\sidetocname}{Contents}
```

```
\end{warpall}
```

for HTML output:

```
\begin{warpHTML}
```

\LWR@sidetoc Creates the actual side-TOC.

```
\newcommand*{\LWR@sidetoc}{
\LWR@forcenewpage
\LWR@stoppars

```

The entire sideTOC is placed into a nav of class sidetoc.

```
\LWR@htmlelementclass{div}{sidetoccontainer}
\LWR@htmlelementclass{nav}{sidetoc}

\setcounter{tocdepth}{\value{SideTOCDepth}}

```

The title is placed into a of class sidetoctitle, and may contain paragraphs.

```
\begin{BlockClass}{sidetoctitle}
\ifcsvoid{thetitle}{}{\InlineClass{sidetocthetitle}{\thetitle}\par}
\sidetocname
\end{BlockClass}
```

The table of contents is placed into a of class sidetoccontents.

```
\begin{BlockClass}{sidetoccontents}
\LinkHome

\LWR@myshorttoc{sidetoc}
\end{BlockClass}
\LWR@htmlelementclassend{nav}{sidetoc}
\LWR@htmlelementclassend{div}{sidetoccontainer}
}
```

74.4 Low-level TOC line formatting

\numberline {⟨*number*⟩}

(Called from each line in the .aux, .lof files.)

Record this section number for further use:

```
\newcommand*{\LWR@numberline}[1]{%
\LWR@sectionnumber{#1}\quad%
}

\LetLtxMacro\numberline\LWR@numberline
```

\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

```
\NewDocumentCommand{\hypertoc}{m m +m m}{%
\LWR@traceinfo{hypertoc !#1!#2!#3!#4!}%
```

Respond to `tocdepth`:

```
\ifthenelse{\cnttest{#1}{<=}{\value{tocdepth}}}%
    {%
        \LWR@startpars%
```

Create an HTML link to `<filename>#autosec-(page)`, with the name, of the given HTML class.

```
        \LWR@subhyperrefclass{%
            \LWR@htmlrefsectionfilename{autopage-#4}%
                \LWR@origpound\LWR@print@mbox{autosec-#4}%
        }{#3}{toc#2}%
        \LWR@stoppars%
    }%
    {}%
\LWR@traceinfo{hypertoc done}%
}
```

Ctr `lofdepth` TOC depth for figures.

```
\@ifclassloaded{memoir}{}{
\newcounter{lofdepth}
\setcounter{lofdepth}{1}
}
```

Ctr `lotdepth` TOC depth for tables.

```
\@ifclassloaded{memoir}{}{
\newcounter{lotdepth}
\setcounter{lotdepth}{1}
}
```

`\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

```
\newcommand{\hypertocfloat}[5]{%
\LWR@startpars
```

If some float-creation package has not yet defined the float type's `lofdepth` counter, etc, define it here:

```
\@ifundefined{c@#3depth}{%
\newcounter{#3depth}%
\setcounter{#3depth}{1}%
}{}%
```

Respond to `lofdepth`, etc.:

```
\LWR@traceinfo{hypertocfloat depth is #1 #3depth is \arabic{#3depth}}%
\ifthenelse{\cnttest{#1}{<=}{\arabic{#3depth}}}%
    {%
        \LWR@startpars%
```

Create an HTML link to filename#autoid-(float number), with text of the caption, of the given HTML class.

```
        \LWR@subhyperrefclass{%
        \LWR@htmlrefsectionfilename{autopage-\arabic{LWR@nextautopage}}%
        \LWR@origpound\LWR@print@mbox{autoid-\arabic{LWR@nextautoid}}}%
        {#4}{toc#2}%
        \LWR@stoppars%
    }%
    {}%
}
```

Automatically called by `\contentsline`:

`\l@part` {⟨*name*⟩} {⟨*page*⟩}

Uses `\DeclareDocumentCommand` in case the class does not happen to have a `\part`.

```
\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`.

```
\@ifundefined{chapter}
{}
{
\DeclareDocumentCommand{\l@chapter}{m m}
    {\hypertoc{0}{chapter}{#1}{#2}}
}
```

\l@section {⟨*name*⟩} {⟨*page*⟩}

```
\renewcommand{\l@section}[2]{\hypertoc{1}{section}{#1}{#2}}
```

\l@subsection {⟨*name*⟩} {⟨*page*⟩}

```
\renewcommand{\l@subsection}[2]{\hypertoc{2}{subsection}{#1}{#2}}
```

\l@subsubsection {⟨*name*⟩} {⟨*page*⟩}

```
\renewcommand{\l@subsubsection}[2]{\hypertoc{3}{subsubsection}{#1}{#2}}
```

\l@paragraph {⟨*name*⟩} {⟨*page*⟩}

```
\renewcommand{\l@paragraph}[2]{\hypertoc{4}{paragraph}{#1}{#2}}
```

\l@subparagraph {⟨*name*⟩} {⟨*page*⟩}

```
\renewcommand{\l@subparagraph}[2]{\hypertoc{5}{subparagraph}{#1}{#2}}
```

\l@figure {⟨*name*⟩} {⟨*page*⟩}

```
\renewcommand{\l@figure}[2]{\hypertocfloat{1}{figure}{lof}{#1}{#2}}
```

\l@table {⟨*name*⟩} {⟨*page*⟩}

```
\renewcommand{\l@table}[2]{\hypertocfloat{1}{table}{lot}{#1}{#2}}

\end{warpHTML}
```

75 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.

for HTML output:

```
\begin{warpHTML}

\newcounter{LWR@autoindex}
\setcounter{LWR@autoindex}{0}

\newcounter{LWR@autoglossary}
\setcounter{LWR@autoglossary}{0}
```

Env theindex

```
\@ifundefined{chapter}
{\newcommand*{\LWR@indexsection}[1]{\section*{#1}}}
{\newcommand*{\LWR@indexsection}[1]{\chapter*{#1}}}

\AtBeginDocument{
\renewenvironment*{theindex}{%
    \LWR@indexsection{\indexname}%
    \let\item\LWR@indexitem%
    \let\subitem\LWR@indexsubitem%
    \let\subsubitem\LWR@indexsubsubitem%
}{}
}% AtBeginDocument
```

\LWR@indexitem [⟨*index key*⟩] The optional argument is added to support repeatindex.

```
\newcommand{\LWR@indexitem}[1][\@empty]{

\InlineClass{indexitem}{}#1%
}
```

\LWR@indexsubitem

```
\newcommand{\LWR@indexsubitem}{

\InlineClass{indexsubitem}{}
}
```

\LWR@indexsubsubitem

```
\newcommand{\LWR@indexsubsubitem}{

\InlineClass{indexsubsubitem}{}
}
```

\@wrindex {⟨*term*⟩} Redefined to write the LWR@autoindex counter instead of page.

```
\def\LWR@wrindex#1{%
\addtocounter{LWR@autoindex}{1}%
\LWR@new@label{LWRindex-\arabic{LWR@autoindex}}%
\protected@write\@indexfile{}%
{\string\indexentry{#1}{\arabic{LWR@autoindex}}}%
\endgroup
\@esphack}

\AtBeginDocument{
\let\@wrindex\LWR@wrindex
}
```

\@wrglossary {⟨*term*⟩} Redefined to write the LWR@latestautopage counter instead of page.

```
\def\@wrglossary#1{%
\addtocounter{LWR@autoglossary}{1}%
\LWR@new@label{LWRglossary-\theLWR@autoglossary}%
\protected@write\@glossaryfile{}%
{\string\glossaryentry{#1}{\theLWR@autoglossary}}%
\endgroup
\@esphack}
```

`\LWR@indexnameref` {⟨*LWR@autoindex*⟩}

Creates a hyperlink based on the given entry's autoindex.

```
\newcommand*{\LWR@indexnameref}[1]{\nameref{LWRindex-#1}}
```

`\LWR@doindexentry` {⟨*LWR@autoindex, or macros.*⟩}

Creates a hyperlink, or handles `\see`, `\textbf`, etc.

```
\newrobustcmd{\LWR@doindexentry}[1]{%
\IfInteger{#1}%
    {\LWR@indexnameref{#1}}%
    {#1}%
}
```

`\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{. . . }
```

```
\newcommand{\LWR@hyperindexrefnullified}{%
\renewrobustcmd{\emph}[1]{\LWR@HTML@emph{\LWR@doindexentry{##1}}}%
\renewrobustcmd{\textbf}[1]{\LWR@HTML@textbf{\LWR@doindexentry{##1}}}%
\renewrobustcmd{\textrm}[1]{\LWR@HTML@textrm{\LWR@doindexentry{##1}}}%
\renewrobustcmd{\textsf}[1]{\LWR@HTML@textsf{\LWR@doindexentry{##1}}}%
\renewrobustcmd{\texttt}[1]{\LWR@HTML@texttt{\LWR@doindexentry{##1}}}%
\renewrobustcmd{\textup}[1]{\LWR@HTML@textup{\LWR@doindexentry{##1}}}%
\renewrobustcmd{\textsc}[1]{\LWR@HTML@textsc{\LWR@doindexentry{##1}}}%
\renewrobustcmd{\textsi}[1]{\LWR@HTML@textsi{\LWR@doindexentry{##1}}}%
\renewrobustcmd{\textit}[1]{\LWR@HTML@textit{\LWR@doindexentry{##1}}}%
\renewrobustcmd{\textsl}[1]{\LWR@HTML@textsl{\LWR@doindexentry{##1}}}%
}
```

`\hyperindexref` {⟨*LWR@autoindex*⟩}

`\hyperindexref{LWR@autoindex}` is inserted into `*.ind` by the makeindex style file `lwarp.ist` or the xindy style file `lwarp.xdy`.

```
\newcommand{\hyperindexref}[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.

```
\def\LWR@tempone{#1}%
\IfBeginWith{\LWR@tempone}{ }{%
    \StrGobbleLeft{\LWR@tempone}{1}[\LWR@tempone]%
}{}%
```

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.

```
\IfInteger{\LWR@tempone}%
    {\LWR@indexnameref{\LWR@tempone}}%
    {%
        \begingroup%
        \LWR@hyperindexrefnullified
        #1%
        \endgroup%
    }%
}
```

```
\end{warpHTML}
```

for PRINT output: A null command for print mode, in case hyperref was not used:

```
\begin{warpprint}
\newcommand{\hyperindexref}[1]{#1}
\end{warpprint}
```

for HTML & PRINT: For the glossaries package, try to prevent an error where `\glo@name` was not found:

```
\begin{warpall}
\providecommand{\glo@name}{}
\end{warpall}
```

76 Bibliography presentation

for HTML output:

```
\begin{warpHTML}
```

`\bibliography` {⟨*filenames*⟩}

Modified to use the base jobname instead of the `_html` jobname.

```
\def\bibliography#1{%
    \if@filesw
    \immediate\write\@auxout{\string\bibdata{#1}}%
    \fi
%     \@input@{\jobname.bbl}% original
    \begingroup%
```

```
    \@input@{\BaseJobname.bbl}% lwarp
    \endgroup%
}
```

`\@biblabel` {⟨*text-refnumber*⟩}

```
\renewcommand{\@biblabel}[1]{[#1]\quad}
```

Env `thebibliography` 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

```
\AtBeginDocument{
\AtBeginEnvironment{thebibliography}{
\providecommand*{\LWR@newem}[1]{\textit{#1}}

\renewrobustcmd{\em}{%
  \begingroup
    \gdef\LWR@em@after{\LWR@em@finish\LWR@newem}%
    \afterassignment\LWR@em@after
    \toks@\bgroup
}

\def\LWR@em@finish#1{%
    \xdef\LWR@em@after{\noexpand#1{\the\toks@}}%
  \endgroup
  \LWR@em@after\egroup
}
}% \AtBeginEnvironment{thebibliography}
}% \AtBeginDocument

\end{warpHTML}
```

77 Restoring original formatting

`\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`.

for HTML output:

```
\begin{warpHTML}
```

```
\newcommand*{\LWR@restoreorigformatting}{%
\LWR@traceinfo{LWR@restoreorigformatting}%
```

Numerous macros change their print/HTML meaning depending on `\LWR@formatting`:

```
\renewcommand*{\LWR@formatting}{print}%
\linespread{1}%

\let\par\LWR@origpar%

\LWR@select@print@hspace%

\LetLtxMacro\hfil\LWR@orighfil%
\let\hss\LWR@orighss%
\let\llap\LWR@origllap%
\let\rlap\LWR@origrlap%
\let\hfilneg\LWR@orighfilneg%

\let\,\LWR@origcomma% disable HTML short unbreakable space
\let\thinspace\LWR@origthinspace% disable HTML short unbreakable space
\let\negthinspace\LWR@orignegthinspace% disable HTML negative short unbreakable space
\let\textellipsis\LWR@origtextellipsis%
\let\textless\LWR@origtextless%
\let\textgreater\LWR@origtextgreater%
\LetLtxMacro\rmfamily\LWR@origrmfamily%
\LetLtxMacro\sffamily\LWR@origsffamily%
\LetLtxMacro\ttfamily\LWR@origttfamily%
\LetLtxMacro\bfseries\LWR@origbfseries%
\LetLtxMacro\mdseries\LWR@origmdseries%
\LetLtxMacro\upshape\LWR@origupshape%
\LetLtxMacro\slshape\LWR@origslshape%
\LetLtxMacro\scshape\LWR@origscshape%

\LetLtxMacro\sishape\LWR@origsishape%

\LetLtxMacro\itshape\LWR@origitshape%
\LetLtxMacro\em\LWR@origem%
\LetLtxMacro\normalfont\LWR@orignormalfont%
\let\sp\LWR@origsp%
\let\sb\LWR@origsb%
\LetLtxMacro\textsuperscript\LWR@origtextsuperscript%
\LetLtxMacro\@textsuperscript\LWR@orig@textsuperscript%
\LetLtxMacro\textsubscript\LWR@origtextsubscript%
\LetLtxMacro\@textsubscript\LWR@orig@textsubscript%
\LetLtxMacro\underline\LWR@origunderline%
\let~\LWR@origtilde%
\let\enskip\LWR@origenskip%
\let\quad\LWR@origquad%
\let\qquad\LWR@origqquad%
```

`\endtabular` must be restored to its original, instead of relying on lwarp's `\LWR@formatted` mechanism:

```
\LetLtxMacro\endtabular\LWR@origendtabular%
\csletcs{endtabular*}{LWR@origendtabular*}%

\LetLtxMacro\noalign\LWR@orignoalign%
\LetLtxMacro\hline\LWR@orighline%

\let\newline\LWR@orignewline%
\LetLtxMacro\includegraphics\LWR@origincludegraphics%
\let\TeX\LWR@origTeX%
\let\LaTeX\LWR@origLaTeX%
\let\LaTeXe\LWR@origLaTeXe%
\renewcommand*{\Xe}{X\textsubscript{E}}%

\LetLtxMacro\@ensuredmath\LWR@origensuredmath%
%
\LWR@restoreorigaccents%
\LWR@restoreoriglists%
%
\LWR@FBcancel%
}

\end{warpHTML}
```

78 Math

78.1 Limitations

78.1.1 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`:

```
\section{A math \texorpdfstring{$1+2=3$}{text description} name}
```

78.1.2 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 in-line

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 11 regarding EPUB output with MathJax.

78.1.3 SVG option

SVG math option

For SVG math, math is rendered as usual by LaTeX into the initial PDF file using the current font[18], 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:

```
\newcommand*{\LateximageFontScale}{.75}
```

SVG math copy/paste

For SVG math, text copy/paste from the HTML `<alt>` tags lists the equation number or tag for single equations, along with the LaTeX code for the math expression. For $\mathcal{AMS}$ environments with multiple numbers in the same environment, only the first and last is copy/pasted, as a range. No tags are listed inside a starred $\mathcal{AMS}$ environment, although the `\tag` macro will still appear inside the LaTeX math expression.

⚠ SVG math in TeX boxes

SVG math does not work inside TeX boxes, since a `\newpage` is required before and after each image.

78.1.4 MathJax option

MathJax math option

Prog MathJax

The popular MathJax alternative (`mathjax.org`) may be used to display math.

When MathJax is enabled, math is rendered twice:

[18] See section 481 regarding fonts and fractions.

1. As regular LaTeX PDF output placed inside an HTML comment, allowing equation numbering and cross referencing to be almost entirely under the control of LaTeX, and
2. As detokenized printed LaTeX commands placed directly into the HTML output for interpretation by the MathJax display scripts. An additional script is used to pre-set the equation number format and value according to the current LaTeX values, and the MathJax cross-referencing system is ignored in favor of the LaTeX internal system, seamlessly integrating with the rest of the LaTeX code.

78.1.5 Customizing MathJax

MathJax does not have preexisting support every possible math function. Additional MathJax function definitions may be defined. These will be declared at the start of each HTML page, and thus will have a global effect.

Examples:

```
\CustomizeMathJax{
    \newcommand{\expval}[1]{\langle#1\rangle}
    \newcommand{\abs}[1]{\lvert#1\rvert}
}
\CustomizeMathJax{\newcommand{\arsinh}{\text{arsinh}}}
\CustomizeMathJax{\newcommand{\arcosh}{\text{arcosh}}}
\CustomizeMathJax{\newcommand{\NN}{\mathbb{N}}}
```

78.1.6 MathJax limitations

MathJax limitations

Prog MathJax

Limitations when using MathJax include:

subequations

- MathJax itself does not support subequations. This may be improved by parsing the LaTeX math expression to manually insert tags, but this has not yet been done.

footnotes in math

- Footnotes inside equations are not yet supported while using MathJax.

lateximage

- Math appearing inside a lateximage, and therefore also inside a Ti*k*z or `picture` environment, is rendered as SVG math even if MathJax is used in the rest of the document.

siunitx

⚠ siunitx inside an equation

- Usage of siunitx inside a math equation is supported via a third-party MathJax extension. While inside a math expression, do not use `\SI` or `\si` inside `\text`, where it will be rendered as normal text.

  https://github.com/burnpanck/MathJax-siunitx

  Also see section 9.7.11.

tabbing

- A `tabbing` environment is emulated using an HTML <pre>.While MathJax is enabled inside `tabbing`, the browser may not correctly render the horizontal alignment of the math and text following after on the same line.

⚠ other macros and packages

- Other math-related macros and packages are not supported by MathJax, including \ensuremath and bigdelim, along with occasionally-used macros such as \relax. lwarp emulates footnotes, units, and nicefrac for MathJax.

78.1.7 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.

78.1.8 Complicated inline math objects

\inlinemathnormal
\inlinemathother

changing contents

complicated alt tag

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. The macro \inlinemathother may be used before a dynamic math expression, and \inlinemathnormal after. Doing so tells lwarp to use an unhashed SVG math image, even if MathJax is in use. See section 44.

78.1.9 Complicated display math objects

\displaymathnormal

By default, or when selecting \displaymathnormal, MathJax math display environments print their contents as text into HTML, 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
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 Ti*k*z 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 Ti*k*z pictures are more likely to compile successfully.

78.2 HTML alt tag names

Redefinable names for the HTML alt tags, for translation according to the reader's native language.

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

\mathimagename
Default: ``math image''

The HTML alt tag for an SVG math image.

```
\newcommand*{\mathimagename}{math image}
```

\packagediagramname
Default: ``diagram''

Appended to the lateximage HTML alt tag for the images generated by many packages.

```
\newcommand*{\packagediagramname}{diagram}
```

```
\end{warpall}
```

78.3 Inline and display math

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

Ctr LWR@externalfilecnt

Counter for the external files which are generated and then referenced from the HTML:

```
\newcounter{LWR@externalfilecnt}
```

Bool LWR@indisplaymathimage

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.

```
\newbool{LWR@indisplaymathimage}
```

Bool LWR@xfakebold

True if xfakebold \setBold is in use.

```
\newbool{LWR@xfakebold}
\boolfalse{LWR@xfakebold}
```

\LWR@orig@setBold

Redefined by lwarp-xfakebold.

```
\newcommand*{\LWR@orig@setBold}{}
```

\LWR@orig@unsetBold

Redefined by lwarp-xfakebold.

```
\newcommand*{\LWR@orig@unsetBold}{}
```

\LWR@applyxfakebold

Redefined by lwarp-xfakebold.

```
\newcommand*{\LWR@applyxfakebold}{}
```

\$ 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.

```
\let\LWR@origtextdollar\$

\renewcommand*{\$}{%
\ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
{\LWR@origtextdollar}%
{\HTMLentity{dollar}}%
}
```

File lwarp_baseline_marker.png
File lwarp_baseline_marker.eps

A marker to be used to help *pdfcrop* identify the inline math baseline and width. If either graphicx or graphics is loaded, this marker is placed at the lower left and lower right corners of the inline math. *pdfcrop* is then able to identify the width of the image, and also the height of an image such as a horizontal dash which does not otherwise touch the baseline.

A marker with alpha or opacity of 0% is not registered by *pdfcrop*, so the marker is a small square block of 1% alpha, which seems to work while still being effectively invisible in the final SVG image.

If graphicx is loaded, this marker is sized as a tiny 1 sp square. If graphics is loaded, this marker is used at its default size of around .25 pt. If neither graphics package is loaded, the marker is replaced by a 10 sp horizontal space, and there is no assistance for determining baseline or width of the inline math image. The best results are obtained when using graphicx.

\LWR@addbaselinemarker Places a small marker in an SVG inline image. If graphics or graphicx are loaded, the marker is a mostly transparent image. If neither is loaded, no marker is used.

```
\AtBeginDocument{

\ifpdf
    \newcommand*{\LWR@baselinename}{lwarp_baseline_marker.png}
\else
    \ifXeTeX
        \newcommand*{\LWR@baselinename}{lwarp_baseline_marker.png}
    \else
        \newcommand*{\LWR@baselinename}{lwarp_baseline_marker.eps}
    \fi
\fi

\IfFileExists{\LWR@baselinename}%
{
    \@ifpackageloaded{graphicx}{
        \newcommand*{\LWR@addbaselinemarker}{%
            \LWR@origincludegraphics{\LWR@baselinename}%
        }
    }{
        \@ifpackageloaded{graphics}{
            \newcommand*{\LWR@addbaselinemarker}{%
                \LWR@origincludegraphics{\LWR@baselinename}%
```

```
            }
        }{
            \PackageWarning{lwarp}{Load graphicx or graphics
                for improved SVG math baselines,}
            \newcommand*{\LWR@addbaselinemarker}{}
        }
    }
}{% lwarp_baseline_marker.png or .eps is not present
   \PackageWarning{lwarp}{File \LWR@baselinename\space is not installed alongside
        the lwarp-*.sty files, so SVG math baselines may not be accurate,}
    \newcommand*{\LWR@addbaselinemarker}{}
}

}% AtBeginDocument
```

\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`.

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 duplicated in the additional hashing argument.

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.

```
\newlength{\LWR@singledollarwidth}
\newlength{\LWR@singledollarheight}
\newlength{\LWR@singledollardepth}

\newsavebox{\LWR@singledollarbox}

\NewDocumentCommand{\LWR@subsingledollar}{s m m m}{%
\LWR@traceinfo{LWR@subsingledollar}%

\ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
{%
\LWR@traceinfo{LWR@subsingledollar: already in a lateximage}%
    #4% contents
}%
{% not in a lateximage
\begingroup%
```

Support for xfakebold:

```
\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 `\mathimagename`. 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:

```
\ifboolexpr{%
    (
        bool{mathjax} or
        ( bool{FormatWP} and bool{WPMarkMath} )
    ) and
   ( not test { \ifstrequal {#2} {(\mathimagename)} } ) and % from \ensuredmath
    ( not bool{LWR@dynamicmath} )
}%
```

For MathJax, print the math between `\(` and `\)`:

```
{%
    \LWR@traceinfo{LWR@subsingledollar: Mathjax}%
    {\textbackslash(\LWR@HTMLsanitize{#4}\textbackslash)}%
}% 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.

```
{% not mathjax
\LWR@traceinfo{LWR@subsingledollar: NOT mathjax, or is ensuremath, or is dynamic}%
```

Measure the depth, width, and height of the math image:

```
    \begingroup%
```

Temporarily disable formatting while measuring the image parameters:

```
    \LWR@restoreorigformatting%
    \RenewDocumentEnvironment{lateximage}{s o o o}{}{}% inside group
    \LWR@print@normalsize%
```

Temporarily set font for the HTML PDF output:

```
    \LWR@traceinfo{Using font family \LWR@f@family}%
    \@nameuse{LWR@orig\LWR@f@family family}%
    \LWR@traceinfo{Using font series \LWR@f@series}%
    \@nameuse{LWR@orig\LWR@f@series series}%
    \LWR@traceinfo{Using font shape \LWR@f@shape}%
    \@nameuse{LWR@orig\LWR@f@shape shape}%
```

`lateximagedepth` must be nested to avoid generating paragraph tags. $\mathcal{AMS}$ math modifies the `\text` macro such that `\addtocounter` does not always occur as expected. Lower-level code is used instead.

```
\global\advance\c@LWR@lateximagedepth 1\relax%
```

Typeset and save the contents, depending on how they were generated:

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".

```
\ifmmode%
    \global\sbox{\LWR@singledollarbox}{#4}%
\else%
    \ifbool{LWR@dynamicmath}{%
        \ifbool{mathjax}{%
            \global\sbox{\LWR@singledollarbox}{\LWR@origensuredmath{#4}}%
        }{%
            \global\sbox{\LWR@singledollarbox}{#4}%
        }%
    }{%
        \global\sbox{\LWR@singledollarbox}{#4}%
    }%
\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.

```
\global\sbox{\LWR@singledollarbox}{%
    \usebox{\LWR@singledollarbox}%
    \raisebox{-\dp\LWR@singledollarbox}{%
        \LWR@addbaselinemarker%
    }%
}%
```

More low-level code to undo the counter change.

```
\global\advance\c@LWR@lateximagedepth -1\relax% Due to AmS \text macro.
```

Measure the depth:

```
    \setlength{\LWR@singledollardepth}{%
        \LateximageFontScale\dp\LWR@singledollarbox%
    }%
```

Make the length a global change:

```
    \global\LWR@singledollardepth=\LWR@singledollardepth%
```

Likewise for width:

```
    \setlength{\LWR@singledollarwidth}{%
        \LateximageFontScale\wd\LWR@singledollarbox%
    }%
    \global\LWR@singledollarwidth=\LWR@singledollarwidth%
```

Likewise for total height:

```
    \setlength{\LWR@singledollarheight}{%
        \LateximageFontScale\ht\LWR@singledollarbox%
    }%
    \addtolength{\LWR@singledollarheight}{%
        \LateximageFontScale\dp\LWR@singledollarbox%
    }%
    \global\LWR@singledollarheight=\LWR@singledollarheight%

    \endgroup%
```

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:

```
    \ifdimgreater{\LWR@singledollarwidth}{.7\LWR@singledollarheight}{%
        \def\LWR@singledollarstyle{%
            width:\LWR@convertto{em}{\the\LWR@singledollarwidth} em%
        }%
    }{%
        \def\LWR@singledollarstyle{%
            height:\LWR@convertto{em}{\the\LWR@singledollarheight} em%
        }%
    }%
```

If a very narrow width, use the height.

```
    \ifdimless{\LWR@singledollarwidth}{.2em}%
    {%
        \def\LWR@singledollarstyle{%
            height:\LWR@convertto{em}{\the\LWR@singledollarheight} em%
        }%
    }%
    {}%
```

If very wide and short, use the width:

```
    \ifdimless{\LWR@singledollarheight}{.2em}%
    {%
        \def\LWR@singledollarstyle{%
            width:\LWR@convertto{em}{\the\LWR@singledollarwidth} em%
        }%
    }%
    {}%
```

If there is significant text depth, add the depth to the style.

```
    \ifdimgreater{\LWR@singledollardepth}{0.05ex}{%
        \def\LWR@singledollardepthstyle{%
            \ ; % extra space
            \LWR@print@mbox{%
            vertical-align:-\LWR@convertto{em}{\the\LWR@singledollardepth} em%
            } % extra space
        }%
    }{%
        \def\LWR@singledollardepthstyle{}%
    }%
```

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.

```
    \ifbool{LWR@dynamicmath}{%
        \LWR@traceinfo{subsingledollar: dynamic}%
        \begin{lateximage}% no hashing
            [(\mathimagename)]% alt tag
            []% no add'l hashing
            [\LWR@singledollarstyle \LWR@singledollardepthstyle]% CSS
    }{%
        \LWR@traceinfo{subsingledollar: static}%
        \IfValueTF{#1}{%
            \LWR@findcurrenttextcolor% sets \LWR@tempcolor
```

Support for xfakebold:

```
            \ifbool{LWR@xfakebold}%
                {\def\LWR@tempone{Y}}%
                {\def\LWR@tempone{N}}%

            \begin{lateximage}*% use hashing
                [#2]% alt
                [% addl' hashing
                    #3%
                    FM\LWR@f@family%
                    SR\LWR@f@series%
                    SH\LWR@f@shape%
                    CL\LWR@tempcolor%
                    FB\LWR@tempone% xfakebold
                ]%
```

```
                [\LWR@singledollarstyle \LWR@singledollardepthstyle]% CSS
        }{%
            \begin{lateximage}% no hashing
                [#2]% alt
                []% no add'l hashing
                [\LWR@singledollarstyle \LWR@singledollardepthstyle]% CSS
        }%
    }%
```

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.)

```
    \LWR@addbaselinemarker%
```

Support for xfakebold:

```
    \LWR@applyxfakebold%
```

Typeset the contents:

```
    \usebox{\LWR@singledollarbox}%
```

The closing baseline marker:

```
    \LWR@addbaselinemarker%

    \end{lateximage}%
%
}% not mathjax
\endgroup%
}% not in a lateximage
\LWR@traceinfo{LWR@subsingledollar: done}%
}

\LetLtxMacro\LWR@origdollar$
\LetLtxMacro\LWR@secondorigdollar$% balance for editor syntax highlighting

\LetLtxMacro\LWR@origopenparen\(
\LetLtxMacro\LWR@origcloseparen\)
\LetLtxMacro\LWR@origopenbracket\[
\LetLtxMacro\LWR@origclosebracket\]
```

$
$$

Redefine the dollar sign to place math inside a `lateximage`, or use MathJax:

```
\begingroup
\catcode`\$=\active%
\protected\gdef${\@ifnextchar$\LWR@doubledollar\LWR@singledollar}%
```

Used by chemformula to escape single-dollar math:

```
\protected\gdef\LWR@newsingledollar{\@ifnextchar$\LWR@doubledollar\LWR@singledollar}%
```

\LWR@doubledollar Redefine the double dollar sign to place math inside a lateximage, or use MathJax:

```
\protected\gdef\LWR@doubledollar$#1$${%
```

If MathJax or formatting for a word processor, print the LaTeX expression:

```
\ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
```

For MathJax, print the math between \[and \]:

```
{

    \textbackslash[%
    \LWR@HTMLsanitize{#1}%
    \textbackslash]

}% mathjax
```

For SVG, print the math inside a lateximage, with an <alt> tag of the LaTeX code:

```
{% not mathjax
    \begin{BlockClass}{displaymath}%
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}%
    [%
        \textbackslash{[} % extra space
        \LWR@HTMLsanitize{#1} % extra space
        \textbackslash{]}%
    ]%
```

Support for xfakebold:

```
    \LWR@applyxfakebold%

    \LWR@origdollar\LWR@origdollar#1\LWR@origdollar\LWR@origdollar%
    \end{lateximage}%
    \end{BlockClass}%
}% not mathjax
}%
```

\LWR@singledollar {⟨*alt text*⟩} {⟨*math expression*⟩}

```
\protected\gdef\LWR@singledollar#1${%
\ifbool{mathjax}{%
    \LWR@subsingledollar*%
    {% alt tag
        \textbackslash( %
        \LWR@HTMLsanitize{#1} % extra space
```

```
        \textbackslash)%
    }%
    {singledollar}% add'l hashing
    {#1}% contents
}{% not mathjax
    \LWR@subsingledollar*%
    {% alt tag
        \textbackslash( %
        \LWR@HTMLsanitize{#1} % extra space
        \textbackslash)%
    }%
    {singledollar}% add'l hashing
    {\LWR@origensuredmath{#1}}% contents
}% not mathjax
}
```

`\(` `\[` Redefine to the above dollar macros.

```
\AtBeginDocument{
\protected\gdef\(#1\){$#1$}
\protected\gdef\[#1\]{$$#1$$}
}

\endgroup
```

```
\AtBeginDocument{
\LetLtxMacro\LWR@openbracketnormal\[
\LetLtxMacro\LWR@closebracketnormal\]
}
```

`\@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.

```
\LetLtxMacro\LWR@origensuredmath\@ensuredmath

\renewcommand{\@ensuredmath}[1]{%
\ifbool{mathjax}{%
    \LWR@subsingledollar*{(\mathimagename)}{%
            \protect\LWR@HTMLsanitize{\detokenize\expandafter{#1}}%
    }{\relax%
        \LWR@origensuredmath{#1}%
    }%
}{% SVG math
```

If already inside a `lateximage` in math mode, continue as-is.

```
\ifmmode%
    \LWR@origensuredmath{#1}%
\else%
```

Create an inline math `lateximage` with a simple `alt` tag and additional hashing according to the contents.

```
    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
    {\LWR@origensuredmath{#1}}%
    {%
        \LWR@subsingledollar*{(\mathimagename)}{%
           \protect\LWR@HTMLsanitize{\detokenize\expandafter{#1}}%
        }{%
            \LWR@origensuredmath{#1}%
        }%
    }%
\fi%
}%
}
```

Remove the old `math` and `displaymath` environments:

```
\let\math\relax
\let\endmath\relax
\let\displaymath\relax
\let\enddisplaymath\relax
```

Env `math` Set math mode then typeset the body of what was between the begin/end. See the environ package for `\BODY`.

```
\NewEnviron{math}{\expandafter\(\BODY\)}
```

Env `LWR@displaymathnormal` Set math mode then typeset the body of what was between the begin/end. See the environ package for `\BODY`.

```
\NewEnviron{LWR@displaymathnormal}{\expandafter\[\BODY\]\@ignoretrue}
```

Set the default displaymath to the normal version:

```
\LetLtxMacro\[\LWR@openbracketnormal%
\LetLtxMacro\]\LWR@closebracketnormal%
\LetLtxMacro\displaymath\LWR@displaymathnormal%
\LetLtxMacro\enddisplaymath\endLWR@displaymathnormal%
```

Env `LWR@displaymathother` A version of `displaymath` which can handle complicated objects, but does not supply MathJax or HTML `alt` tags.

```
\newenvironment{LWR@displaymathother}
{%
    \begin{BlockClass}{displaymath}%
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}%
    [(display math)]%
    \LWR@origdollar\LWR@origdollar%
```

```
}
{%
    \LWR@origdollar\LWR@origdollar%
    \end{lateximage}%
    \end{BlockClass}%
}
```

Env LWR@equationother

A version of displaymath which can handle complicated objects, but does not supply MathJax or HTML alt tags.

```
\newenvironment{LWR@equationother}
{%
    \begin{BlockClass}{displaymathnumbered}%
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}%
    [(display math)]%
    \LWR@origequation%
}
{%
    \LWR@origendequation%
    \end{lateximage}%
    \end{BlockClass}%
}
```

78.4 MathJax support

Ctr LWR@nextequation

Used to add one to compute the next equation number.

```
\newcounter{LWR@nextequation}
```

\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.

```
\newcommand*{\LWR@syncmathjax}{%
```

If using chapters, place the chapter number in front of the equation. Otherwise, use the simple equation number.

```
\ifcsdef{thechapter}{
    \InlineClass{hidden}{
        \textbackslash(
        \textbackslash{}seteqsection \{\thechapter\}
        \textbackslash)
    }
}
{}% not using chapters
```

MathJax doesn't allow setting the equation number to 1:

```
\ifthenelse{\cnttest{\value{equation}}>0}
{
```

Tell MathJax that the next set of equations begins with the current LaTeX equation number, plus one.

```
    \setcounter{LWR@nextequation}{\value{equation}}
    \addtocounter{LWR@nextequation}{1}
```

Place the MathJax command inside "\(" and "\)" characters, to be printed to HTML, not interpreted by LaTeX.

```
    \InlineClass{hidden}{
        \textbackslash(
        \textbackslash{}seteqnumber \{\arabic{LWR@nextequation}\}
        \textbackslash)
    }
}{}% not eq > 0
}
```

`\LWR@hidelatexequation` {⟨*environment*⟩} {⟨*contents*⟩}

Creates the LaTeX version of the equation inside an HTML comment.

```
\NewDocumentCommand{\LWR@hidelatexequation}{m +m}{%
```

Stop HTML paragraph handling and open an HTML comment:

```
\LWR@stoppars
\LWR@htmlopencomment

```

Start the LaTeX math environment inside the HTML comment:

```
\begingroup
\@nameuse{LWR@orig#1}
```

While in the math environment, restore various commands to their LaTeX meanings.

```
\LWR@restoreorigformatting
```

See `\LWR@htmlmathlabel` in section 78.7.1.

Print the contents of the equation:

```
#2
```

End the LaTeX math environment inside the HTML comment:

```
\@nameuse{LWR@origend#1}
\endgroup

```

Close the HTML comment and resume HTML paragraph handling:

```
\LWR@htmlclosecomment
\LWR@startpars
}
```

`\LWR@addmathjax` {⟨*environment*⟩} {⟨*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.

```
\NewDocumentCommand{\LWR@addmathjax}{m +m}{%
```

Enclose the MathJax environment inside printed "`\(`" and "`\)`" characters.

```
\LWR@origtilde\LWR@orignewline
\textbackslash{}begin\{#1\}
```

Print the contents, sanitizing for HTML special characters.

```
\LWR@HTMLsanitizeexpand{\detokenize\expandafter{#2}}
```

Close the MathJax environment:

```
\textbackslash{}end\{#1\}
\LWR@orignewline
}
```

78.5 Equation environment

Remember existing `equation` environment:

```
\AtBeginDocument{
\let\LWR@origequation\equation
\let\LWR@origendequation\endequation
\csletcs{LWR@origequation*}{equation*}
\csletcs{LWR@origendequation*}{endequation*}
}
```

`\LWR@doequation` {⟨*env contents*⟩} {⟨*env name*⟩}

For SVG math output, the contents are typeset using the original `equation` inside a `latexmage`, 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.

```
\newcommand*{\LWR@doequation}[2]{%
```

If mathjax or FormatWP, print the LaTeX expression:

```
\ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
```

MathJax output:

```
{
```

Print commands to syncronize MathJax's equation number and format to the current LaTeX chapter/section and equation number:

```
    \LWR@syncmathjax
```

Print the LaTeX math inside an html comment:

```
    \LWR@hidelatexequation{#2}{#1}
}
```

SVG output: Create the lateximage along with an html <alt> tag having an equation number, the LaTeX equation environment commmands, and the contents of the environment's \BODY.

```
{% not mathjax
```

Begin the lateximage with an <alt> tag containing the math source:

```
    \ifstrequal{#2}{equation*}{%
        \begin{BlockClass}{displaymath}%
    }{%
        \begin{BlockClass}{displaymathnumbered}%
    }%
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[%
        \ifstrequal{#2}{equation*}{%
            \ifdefequal{\LWR@equationtag}{\theequation}{%
%                                   no tag was given
            }{%
                (\LWR@equationtag) % tag was given
            }%
        }{%
            (\LWR@equationtag) % automatic numbering
        }%
        \textbackslash{begin\{#2\}} % extra space
        \LWR@HTMLsanitizeexpand{\detokenize\expandafter{#1}} % extra space
        \textbackslash{end\{#2\}}%
    ]% alt tag
```

Support for xfakebold:

```
    \LWR@applyxfakebold%
```

Create the actual LaTeX-formatted equation inside the `lateximage` using the contents of the environment.

```
    \@nameuse{LWR@orig#2}%
    #1% contents collected by \collect@body
    \@nameuse{LWR@origend#2}%
    \end{lateximage}%
    \end{BlockClass}%
}% not mathjax
}
```

After the environment, if MathJax, print the math to the HTML output for MathJax processing:

```
\newcommand*{\LWR@doendequation}[1]{%
    \ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
    {%
        \LWR@addmathjax{#1}{\BODY}%
    }{}%

}
```

Remove existing `equation` environment:

```
\AtBeginDocument{
\let\equation\relax
\let\endequation\relax
\csletcs{equation*}{relax}
\csletcs{endequation*}{relax}
}
```

Env `equation` The new `equation` environment is created with `\NewEnviron` (from the environ package), which stores the contents of its environment in a macro called `\BODY`.

```
\AtBeginDocument{
\NewEnviron{equation}%
{\LWR@doequation{\BODY}{equation}}%
[\LWR@doendequation{equation}]

\LetLtxMacro\LWR@equationnormal\equation
\LetLtxMacro\endLWR@equationnormal\endequation
}
```

Env `equation*`

```
\AtBeginDocument{
\NewEnviron{equation*}%
{\LWR@doequation{\BODY}{equation*}}%
[\LWR@doendequation{equation*}]

\csletcs{LWR@equationnormalstar}{equation*}
\csletcs{LWR@endequationnormalstar}{endequation*}
}
```

Remember the “less” version of equation, which uses MathJax and alt tags, but does not support complicated contents such as some Ti*k*z expressions.

```
\AtBeginDocument{
\LetLtxMacro\LWR@equationless\equation
\LetLtxMacro\endLWR@equationless\endequation
\csletcs{LWR@equationlessstar}{equation*}
\csletcs{LWR@endequationlessstar}{endequation*}
}
```

78.6 \displaymathnormal and \displaymathother

\displaymathnormal

By default, or when selecting \displaymathnormal, MathJax math display environments print their contents as text into HTML, 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
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 Ti*k*z 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 Ti*k*z pictures are more likely to compile successfully.

\displaymathnormal
simple math objects

Use when display math environments have simple math which is to sent to MathJax or included in HTML alt tags.

```
\newcommand*{\displaymathnormal}{%
\ifbool{LWR@origmathjax}{\booltrue{mathjax}}{\boolfalse{mathjax}}%
\LetLtxMacro\[\LWR@openbracketnormal%
\LetLtxMacro\]\LWR@closebracketnormal%
\LetLtxMacro\displaymath\LWR@displaymathnormal%
\LetLtxMacro\enddisplaymath\endLWR@displaymathnormal%
\LetLtxMacro\equation\LWR@equationnormal%
\LetLtxMacro\endequation\endLWR@equationnormal%
\csletcs{equation*}{LWR@equationnormalstar}%
\csletcs{endequation*}{LWR@endequationnormalstar}%
}
```

\displaymathother
complicated math objects

Use when display math environments have complicated objects which will not work with MathJax or should not be included in HTML alt tags. Complicated contents are more likely to compile correctly.

```
\newcommand*{\displaymathother}{%
\boolfalse{mathjax}%
\LetLtxMacro\displaymath\LWR@displaymathother%
\LetLtxMacro\enddisplaymath\endLWR@displaymathother%
\LetLtxMacro\[\LWR@displaymathother%
\LetLtxMacro\]\endLWR@displaymathother%
```

```
\LetLtxMacro\equation\LWR@equationother%
\LetLtxMacro\endequation\endLWR@equationother%
\csletcs{equation*}{displaymath}%
\csletcs{endequation*}{enddisplaymath}%
}
```

```
\end{warpHTML}
```

for PRINT output:

```
\begin{warpprint}
```

Print-mode versions:

```
\newcommand*{\displaymathnormal}{}
\newcommand*{\displaymathother}{}
```

```
\end{warpprint}
```

for HTML output:

```
\begin{warpHTML}
```

78.7 AMS Math environments

78.7.1 Support macros

Bool `LWR@amsmultline` 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`.

```
\newbool{LWR@amsmultline}
\boolfalse{LWR@amsmultline}
```

`\LWR@htmlmathlabel` {⟨*label*⟩}

lwarp points `\ltx@label` here. This is used by `\label` when inside a LaTeX AMS math environment's math display environment.

`\LWR@origltx@label` points to the LaTeX original, modified by lwarp, then by amsmath, then by cleveref.

```
\newcommand*{\LWR@htmlmathlabel}[1]{%
\LWR@traceinfo{LWR@htmlmathlabelb #1}%
```

If `mathjax` or `FormatWP`, print the LaTeX expression:

```
\ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
{%
```

The combined LaTeX & HTML label is printed in a `\text` field:

```
    \text{%
```

Shift the label over to the right side of the environment to avoid over-printing the math:

```
    \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.)

```
    \LWR@htmlclosecomment%
    \LWR@origltx@label{#1}%
    \LWR@htmlopencomment%
    }% text
}% mathjax
{%
    \LWR@origltx@label{#1}%
}%
}
```

`\LWR@beginhideamsmath` Starts hiding LaTeX math inside an HTML comment.

```
\newcommand*{\LWR@beginhideamsmath}{
\LWR@stoppars
\LWR@origtilde\LWR@orignewline
\LWR@htmlopencomment

\begingroup
\LWR@restoreorigformatting
}
```

`\LWR@endhideamsmath` Ends hiding LaTeX math inside an HTML comment.

```
\newcommand*{\LWR@endhideamsmath}{
\endgroup

\LWR@htmlclosecomment
\LWR@orignewline
\LWR@startpars
}
```

78.7.2 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 envrionment 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 syched with LaTeX's equation numbers, typeset with LaTeX inside an HTML comment, then printed to HTML output for MathJax to process.

Env eqnarray 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.

```
\let\LWR@origeqnarray\eqnarray
\let\LWR@origendeqnarray\endeqnarray
```

To remember whether the starred environment was used, and thus whether to number the equations:

```
\newbool{LWR@numbereqnarray}
\booltrue{LWR@numbereqnarray}
```

Common code used by eqnarray and Beqnarray (from fancybox):

```
\newcommand{\LWR@eqnarrayfactor}{%
```

If mathjax or FormatWP, print the LaTeX expression:

```
\ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
{%
```

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:

```
    \LWR@syncmathjax
    \boolfalse{LWR@amsmultline}
    \ifbool{LWR@numbereqnarray}
    {
```

If numbering the equations, execute a copy inside an HTML comment block:

```
        \LWR@beginhideamsmath
        \LWR@origeqnarray
        \BODY
        \LWR@origendeqnarray
        \LWR@endhideamsmath
```

Then print the (sanitized) contents to the output for MathJax to interpret:

```
        \LWR@addmathjax{eqnarray}{\BODY}
    }%
    {% not LWR@numbereqnarray
```

If not numbering equations, just create the contents for MathJax:

```
    \LWR@addmathjax{eqnarray*}{\BODY}
    }% LWR@numbereqnarray
```

```
}% mathjax
{% not mathjax
    \ifbool{LWR@numbereqnarray}
    {
```

For numbered SVG equations, first create a `lateximage` with an `alt` attribute containing sanitized copy of the source code:

```
        \begin{BlockClass}{displaymathnumbered}%
        \LWR@newautoidanchor%
        \booltrue{LWR@indisplaymathimage}%
        \begin{lateximage}[(\LWR@startingequationtag--\LWR@equationtag)
            \LWR@addmathjax{eqnarray}{\BODY}]
```

Support for xfakebold:

```
        \LWR@applyxfakebold%
```

Create the image contents using an actual `eqnarray`:

```
        \LWR@origeqnarray
        \BODY
        \LWR@origendeqnarray
        \end{lateximage}
        \end{BlockClass}
    }%
    {% not LWR@numbereqnarray
```

If not numbered, do the same, but an extra `\nonumber` seems to be required:

```
        \begin{BlockClass}{displaymath}
        \LWR@newautoidanchor%
        \booltrue{LWR@indisplaymathimage}%
        \begin{lateximage}[\LWR@addmathjax{eqnarray*}{\BODY}]
```

Support for xfakebold:

```
        \LWR@applyxfakebold%

        \LWR@origeqnarray
        \BODY
        \nonumber
        \LWR@origendeqnarray
        \end{lateximage}
        \end{BlockClass}
    }% LWR@numbereqnarray
}% not mathjax
```

Default to number equations in the future:

```
\booltrue{LWR@numbereqnarray}
}
```

eqnarray itself is made with a blank line before and after to force it to be on its own line:

```
\RenewEnviron{eqnarray}
{%

\LWR@eqnarrayfactor

}
```

The starred version is patched to turn off the numbering:

```
\csgpreto{eqnarray*}{\boolfalse{LWR@numbereqnarray}}

\end{warpHTML}
```

79 Lateximages

79.1 Description

Env lateximage

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 77.)

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 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 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 \mathimagename, and the image is not reused.

This 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 `\packagediagramname`. 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 `lateximages`.

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}
```

79.2 Support counters and macros

for HTML output:

```
9753 \begin{warpHTML}
```

Ctr `LWR@lateximagenumber`

Sequence the images.

```
9754 \newcounter{LWR@lateximagenumber}
9755 \setcounter{LWR@lateximagenumber}{0}
```

Ctr `LWR@lateximagedepth`

Do not create `\lateximage` inside of `\lateximage`.

```
9756 \newcounter{LWR@lateximagedepth}
9757 \setcounter{LWR@lateximagedepth}{0}
```

A few utility macros to write special characters:

```
9758 \edef\LWR@hashmark{\string#} % for use in \write
9759 \edef\LWR@percent{\@percentchar} % for use in \write
```

Ctr `LWR@LIpage`

Used to reference the PDF page number of a lateximage to be written into `<project>-images.txt`.

```
9760 \newcounter{LWR@LIpage}
```

```
9761 \end{warpHTML}
```

79.3 Font size

for HTML & PRINT:

```
9762 \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.*

```
9763 \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 .75, but it may be redefined as needed depending on the HTML font.

```
9764 \newcommand*{\LateximageFontScale}{.75}
```

```
9765 \end{warpall}
```

79.4 Sanitizing math expressions for HTML

for HTML output:

```
9766 \begin{warpHTML}
```

`\LWR@HTMLsanitize` {⟨*text*⟩}

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.

Two versions follow, depending on expansion needs. There may be a better way…

```
9767 \newrobustcmd{\LWR@HTMLsanitize}[1]{%
```

Cancel French babel character handling, and fully expand the strings:

```
9768 \begingroup%
9769 \LWR@FBcancel%
9770 \fullexpandarg%
```

The &, <, and > may be interpreted by the browser:

```
9771 \protect\StrSubstitute{\detokenize{#1}}%
9772 {\detokenize{&}}{\detokenize{&amp;}}[\LWR@strresult]%

9773 \protect\StrSubstitute{\detokenize\expandafter{\LWR@strresult}}%
9774 {\detokenize{<}}{\detokenize{&lt;}}[\LWR@strresult]%

9775 \protect\StrSubstitute{\detokenize\expandafter{\LWR@strresult}}%
9776 {\detokenize{>}}{\detokenize{&gt;}}[\LWR@strresult]%
```

The double quote occasionally causes problems.

```
\protect\StrSubstitute{\detokenize\expandafter{\LWR@strresult}}%
{\detokenize{"}}{\detokenize{&quot;}}[\LWR@strresult]%
```

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.

```
\protect\StrSubstitute{\detokenize\expandafter{\LWR@strresult}}%
{\detokenize{##}}{\LWR@origpound}[\LWR@strresult]%

\LWR@strresult%
\endgroup%
}
```

\LWR@HTMLsanitizeexpand {⟨*text*⟩}

This version expands the argument before sanitizing it.

```
\newrobustcmd{\LWR@HTMLsanitizeexpand}[1]{%
```

Cancel French babel character handling, and fully expand the strings:

```
\begingroup%
\LWR@FBcancel%
\fullexpandarg%
```

The difference between this and \LWR@HTMLsanitize (without "expand") is the following \expandafter:

```
\protect\StrSubstitute{\detokenize\expandafter{#1}}%
{\detokenize{&}}{\detokenize{&amp;}}[\LWR@strresult]%

\protect\StrSubstitute{\detokenize\expandafter{\LWR@strresult}}%
{\detokenize{<}}{\detokenize{&lt;}}[\LWR@strresult]%

\protect\StrSubstitute{\detokenize\expandafter{\LWR@strresult}}%
{\detokenize{>}}{\detokenize{&gt;}}[\LWR@strresult]%

\protect\StrSubstitute{\detokenize\expandafter{\LWR@strresult}}%
{\detokenize{"}}{\detokenize{&quot;}}[\LWR@strresult]%
```

\LWR@HTMLsanitizeexpand is not used for defining new MathJax macros, so the ## conversion is not needed here.

```
\LWR@strresult%
\endgroup%
}
```

79.5 Equation numbers

Ctr LWR@startingequation — For use with lateximage and multi-line numbered equations. Remembers the next equation number so that it may be printed in the alt tag.

```
\newcounter{LWR@startingequation}

\@ifundefined{chapter}
{
\renewcommand{\theLWR@startingequation}{%
\arabic{LWR@startingequation}%
}
}
{% chapter defined
\renewcommand{\theLWR@startingequation}{%
\ifnumcomp{\value{chapter}}{>}{0}{\arabic{chapter}.}{}%
\arabic{LWR@startingequation}%
}
}
```

Bool LWR@isstartingequation — True for the first equation tag, false for later tags in the same environment.

```
\newbool{LWR@isstartingequation}
```

\LWR@startingequationtag — Prints the starting equation number or tag.

```
\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.

```
\newcommand*{\LWR@equationtag}{}
```

Only if SVG math, patch \tag after packages have loaded, in case someone else modified \tag.

```
\AtBeginDocument{

\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.

```
\NewDocumentCommand{\LWR@remembertag}{m}{%
\ifbool{LWR@isstartingequation}%
{%
    \global\boolfalse{LWR@isstartingequation}%
    \xdef\LWR@startingequationtag{#1}%
```

```
}{}%
\xdef\LWR@equationtag{#1}%
}%

}% not mathjax
}% AtBeginDocument
```

79.6 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.

```
\newcommand*{\LWR@amsmathbody}[1]
{%
\textbackslash\{begin\}\{#1\} % extra space
\LWR@HTMLsanitizeexpand{\detokenize\expandafter{\the\@envbody}}%
\textbackslash\{end\}\{#1\}%
}
```

\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.

```
\newcommand*{\LWR@amsmathbodynumbered}[1]
{%
\ifnumcomp{\value{LWR@startingequation}}{=}{\value{equation}}%
{(\LWR@equationtag)}%
{(\LWR@startingequationtag--\LWR@equationtag)} % extra space
\LWR@amsmathbody{#1} % extra space
}
```

79.7 lateximage environment

Env lateximage * [⟨*2:* <alt> *tag*⟩] [⟨*3: add'l hashing*⟩] [⟨*4: css style*⟩]

Typesets the contents and then renders the result as an SVG file. Star causes the image to be hashed for reuse.

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 duplicated in the additional hashing argument.

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.

File *_html.aux 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.

File *-images.txt 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.

```
\catcode`\$=\active%

\NewDocumentEnvironment{lateximage}{s O{(image)} O{} O{}}
{%
\LWR@traceinfo{lateximage: starting on \jobname.pdf page \arabic{page}}%
\LWR@traceinfo{lateximage: entering depth is \arabic{LWR@lateximagedepth}}%
```

Nested lateximages remain one large lateximage:

```
\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.

```
{%
%     \addtocounter{LWR@lateximagedepth}{1}%
    \global\advance\c@LWR@lateximagedepth 1\relax% Due to AmS \text macro.
}%
```

Otherwise, this is the outer-most lateximage:

```
{% start of outer-most lateximage
```

Remember the next equation number to be allocated, in case it must be printed in a multi-equation environment:

```
\LWR@traceinfo{lateximage: starting outer-most lateximage}%
    \setcounter{LWR@startingequation}{\value{equation}}%
    \addtocounter{LWR@startingequation}{1}%
    \booltrue{LWR@isstartingequation}%
    \let\LWR@startingequationtag\theLWR@startingequation%
```

The default equation tag, unless overwritten by \tag:

```
    \let\LWR@equationtag\theequation%
```

Starting a new lateximage:

```
    \addtocounter{LWR@lateximagenumber}{1}%
  \LWR@traceinfo{lateximage: LWR@lateximagenumber is \arabic{LWR@lateximagenumber}}%
```

While inside a lateximage, locally do not use mathjax:

```
    \boolfalse{mathjax}%
```

Be sure that are doing a paragraph:

```
    \LWR@ensuredoingapar%
```

Next file:

```
    \addtocounter{LWR@externalfilecnt}{1}%
  \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`:

```
    \setcounterpageref{LWR@LIpage}{%
        LWRlateximage-\BaseJobname-\arabic{LWR@lateximagenumber}%
    }%
    \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:

```
    \LWR@htmltag{span\LWR@indentHTML%
     id="lateximage-\BaseJobname-\arabic{LWR@lateximagenumber}"\LWR@indentHTML
        class="lateximagesource"\LWR@orignewline
    }%
```

Write instructions to the `<ImagesDirectory>.txt` file:

```
    \LWR@traceinfo{lateximage: about to write to \BaseJobname-images.txt}%
    \IfBooleanTF{#1}% starred
    {% hash
        \LWR@traceinfo{lateximage: hash true, adding %
            !\detokenize\expandafter{#2}!\detokenize\expandafter{#3}!}%
```

Compute and save the hashed file name for later use:

```
        \edef\LWR@hashedname{%
            \LWR@mdfive{\detokenize\expandafter{#2}-!-#3}%
        }%
        \LWR@traceinfo{lateximage: hash is \LWR@hashedname}%
```

Write the page, hashing, and hashed name:

```
        \immediate\write\LWR@lateximagesfile{%
            |\arabic{LWR@LIpage}|true|\LWR@hashedname|%
```

```
        }%
    }% hash
    {% no hash
```

No hash, so write the page, no hashing, and the image number:

```
        \LWR@traceinfo{lateximage: hash false}%
        \immediate\write\LWR@lateximagesfile{%
        |\arabic{LWR@LIpage}|false|\LWR@ImagesName\arabic{LWR@externalfilecnt}|%
        }%
    }% no hash
```

Place an open comment tag. This will hide any traces of the lateximage PDF page which were picked up by *pdftotext*.

```
    \LWR@traceinfo{lateximage: about to create open comment}%
    \LWR@htmlopencomment%
```

One level deeper. At this outer-most `lateximage`, it is known that this is not being used inside an $\mathcal{AMS}$ `\text`, since the outer-most level will never be in math mode.

```
     \addtocounter{LWR@lateximagedepth}{1}%
```

Start the new PDF page:

```
    \LWR@traceinfo{lateximage: about to create a new page}%
    \LWR@orignewpage%
```

Typeset the image in a "standard" width page and font size:

```
    \LWR@traceinfo{lateximage: about to create minipage}%
    \LWR@print@minipage{6in}%
    \@nameuse{LWR@print@\LateximageFontSizeName}%
```

Temporarily restore formatting to its PDF definitions: Do not produce HTML tags for \hspace, etc. inside a lateximage.

```
    \LWR@traceinfo{lateximage: about to temporarily restore formatting}%
    \LWR@restoreorigformatting%
```

Use full-page footnotes instead of minipage footnotes. These become HTML footnotes.

```
    \def\@mpfn{footnote}%
    \def\thempfn{\thefootnote}%
    \LetLtxMacro\@footnotetext\LWR@footnotetext%
```

Create the `LWRlateximage<number>` label:

```
    \LWR@traceinfo{lateximage: about to create label}%
    \LWR@orig@label{LWRlateximage-\BaseJobname-\arabic{LWR@lateximagenumber}}%
    \LWR@traceinfo{lateximage: finished creating the label}%
```

Enable print-mode math functions:

```
    \LetLtxMacro$\LWR@origdollar%
    \catcode`\$=3% math shift
    \LetLtxMacro\(\LWR@origopenparen%
    \LetLtxMacro\)\LWR@origcloseparen%
```

Only enable print-mode display math if are not already inside display math:

```
    \ifbool{LWR@indisplaymathimage}{}{% not in display math
        \LetLtxMacro\[\LWR@origopenbracket%
        \LetLtxMacro\]\LWR@origclosebracket%
        \let\equation\LWR@origequation%
        \let\endequation\LWR@origendequation%
        \csletcs{equation*}{LWR@origequation*}%
        \csletcs{endequation*}{LWR@origendequation*}%
    }% not in display math
```

For chemformula:

```
    \LetLtxMacro\LWR@newsingledollar$%
    \LetLtxMacro\LWR@newsingledollar$% syntax highlighting

}% end of outer-most lateximage
\LWR@traceinfo{lateximage: finished start of environment}%
}% end of \begin{lateximage}
```

\endlateximage When the environment closes:

```
{% start of \end{lateximage}
\LWR@traceinfo{lateximage: starting end of lateximage}%
```

Nested more than one deep?

```
\LWR@traceinfo{lateximage: internal depth was \arabic{LWR@lateximagedepth}}%
\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.

```
{%
    \LWR@traceinfo{lateximage: unnesting}%
    \global\advance\c@LWR@lateximagedepth -1\relax%
}%
```

If this is the outer-most lateximage:

```
{% end of outer-most lateximage
```

Finish the lateximage minipage and start a new PDF page:

```
\LWR@traceinfo{lateximage: ending outer-most lateximage}%
    \endLWR@print@minipage%
    \LWR@orignewpage%
    \LWR@print@scriptsize%
```

Close the HTML comment which encapsulated any traces of the lateximage picked up by *pdftotext*:

```
    \LWR@print@vspace*{.5\baselineskip}%
    \LWR@htmlclosecomment%
    \LWR@traceinfo{lateximage: The page after the image is \arabic{page}}%
```

Create a link to the lateximage, allowing its natural height:

```
    \IfBooleanTF{#1}% starred
    {% hash
        \LWR@subinlineimage[#2]{lateximage}%
        {%
            \LWR@ImagesDirectory\OSPathSymbol%
            \LWR@print@mbox{\LWR@hashedname}%
        }{svg}{#4}%
    }% hash
    {% no hash
        \LWR@subinlineimage[#2]{lateximage}%
        {%
            \LWR@ImagesDirectory\OSPathSymbol%
            \LWR@print@mbox{\LWR@ImagesName\theLWR@externalfilecnt}%
        }{svg}{#4}%
    }% no hash
```

Be sure that are doing a paragraph:

```
    \LWR@ensuredoingapar%
```

Close the HTML span which has the *pdftotext* comment and also the link to the `.svg` image:

```
    \LWR@htmltag{/span}%
    \ifbool{HTMLDebugComments}{%
        \LWR@htmlcomment{End of lateximage}%
    }{}%
```

Undo one lateximage level. This is not inside an $\mathcal{AMS}$ `\text`, so regular `\addtocounter` may be used here.

```
    \addtocounter{LWR@lateximagedepth}{-1}%
}% end of outer-most lateximage
\LWR@traceinfo{lateximage: exiting depth is \arabic{LWR@lateximagedepth}}%
\LWR@traceinfo{lateximage: done}%
}%
\catcode`\$=3% math shift
\end{warpHTML}
```

for PRINT output:

```
\begin{warpprint}
```

Env `lateximage` [⟨<alt> *tag*⟩] [⟨*css style*⟩]

`varwidth` is used to create a box of the natural width of its contents.

```
\NewDocumentEnvironment{lateximage}{s o o o}
    {\begin{varwidth}[b]{\linewidth}}
    {\end{varwidth}}

\end{warpprint}
```

80 center, flushleft, flushright

for HTML output:

```
\begin{warpHTML}
```

Env center Replace center functionality with CSS tags:

```
\renewenvironment*{center}
{
\LWR@forcenewpage
\ifbool{FormatWP}
{\BlockClass[\LWR@print@mbox{text-align:center}]{center}}
{\BlockClass{center}}
}
{\endBlockClass}
```

Env flushright

```
\renewenvironment*{flushright}
{
\LWR@forcenewpage
\ifbool{FormatWP}
{\BlockClass[\LWR@print@mbox{text-align:right}]{flushright}}
{\BlockClass{flushright}}
}
{\endBlockClass}
```

Env flushleft

```
\renewenvironment*{flushleft}
{
\LWR@forcenewpage
\ifbool{FormatWP}
{\BlockClass[\LWR@print@mbox{text-align:left}]{flushleft}}
{\BlockClass{flushleft}}
}
{\endBlockClass}
```

\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

```
\newcommand*{\LWR@HTML@centering}{%
\ifbool{HTMLDebugComments}{%
    \LWR@htmlcomment{centering}%
}{}%
}
\LWR@formatted{centering}
```

\raggedleft

```
\newcommand*{\LWR@HTML@raggedleft}{%
\ifbool{HTMLDebugComments}{%
    \LWR@htmlcomment{raggedleft}%
}{}%
}
\LWR@formatted{raggedleft}
```

\raggedright

```
\newcommand*{\LWR@HTML@raggedright}{%
\ifbool{HTMLDebugComments}{%
    \LWR@htmlcomment{raggedright}%
}{}%
}
\LWR@formatted{raggedright}
```

\leftline {⟨*text*⟩}

```
\renewcommand{\leftline}[1]{\begin{flushleft}#1\end{flushleft}}
```

\centerline {⟨*text*⟩}

```
\renewcommand{\centerline}[1]{\begin{center}#1\end{center}}
```

\rightline {⟨*text*⟩}

```
\renewcommand{\rightline}[1]{\begin{flushright}#1\end{flushright}}

\end{warpHTML}
```

81 Preloaded packages

for HTML output:

```
\begin{warpHTML}
```

If the given package was loaded before or by lwarp, load the lwarp version as well.

\LWR@PreloadedPackage {⟨*packagename*⟩}

```
\newcommand*{\LWR@PreloadedPackage}[1]{%
    \@ifpackageloaded{#1}%
        {%
            \AtBeginDocument{
                \LWR@origRequirePackage{lwarp-#1}%
            }
        }%
        {}%
}
```

If textcomp was loaded before lwarp, perhaps as part of the font-related packages, explicitly load the lwarp patches now:

```
\LWR@PreloadedPackage{textcomp}
```

If xunicode was loaded before lwarp, perhaps as part of the font-related packages, explicitly load the lwarp patches now:

```
\LWR@PreloadedPackage{xunicode}
```

If graphics or graphicx were loaded before lwarp, perhaps by xunicode, explicitly load the lwarp patches now:

```
\LWR@PreloadedPackage{graphics}
\LWR@PreloadedPackage{graphicx}
```

ulem may be preloaded by ctex, ctexart, and related classes.

```
\LWR@PreloadedPackage{ulem}
```

xetexko-vertical may be preloaded by xetexko.

```
\LWR@PreloadedPackage{xetexko-vertical}
```

geometry is preloaded by lwarp, and perhaps by various classes.

```
\LWR@PreloadedPackage{geometry}
```

plext is preloaded by some CJK classes.

```
\LWR@PreloadedPackage{plext}
```

stfloats is preloaded by ltj* classes.

```
\LWR@PreloadedPackage{stfloats}
```

lltjext is preloaded by ltj* classes.

```
\LWR@PreloadedPackage{lltjext}
```

```
\end{warpHTML}
```

82 siunitx

Pkg siunitx The lwarp core passes a few options to siunitx.

fractions Due to *pdftotext* limitations, fraction output is replaced by symbol output for `per-mode` and `quotient-mode`.

⚠ math mode required Some units will require that the expression be placed inside math mode.

NOTE: As of this writing, the siunitx extension for MathJax is not currently hosted at any public CDN, thus siunitx is not usable with MathJax unless a local copy of this extension is created first.

⚠ tabular Tabular `S` columns are rendered as simple `c` columns, and tabular `s` columns are not supported. These may be replaced by `c` columns with each cell contained in `\num` or `\si`.

for HTML output:

```
\begin{warpHTML}
```

Options for siunitx:

```
\newrobustcmd{\LWR@siunitx@textcelsius}{\HTMLentity{deg}C}
\newrobustcmd{\LWR@siunitx@textdegree}{\HTMLentity{deg}}
\newrobustcmd{\LWR@siunitx@textprime}{\HTMLunicode{2032}}
\newrobustcmd{\LWR@siunitx@textdblprime}{\HTMLunicode{2033}}
\newrobustcmd{\LWR@siunitx@textplanckbar}{\text{\textit{\HTMLunicode{0127}}}}

\appto\LWR@restoreorigformatting{%
\renewrobustcmd{\LWR@siunitx@textcelsius}{\text{\ensuremath{^\circ}C}}%
\renewrobustcmd{\LWR@siunitx@textdegree}{\text{\ensuremath{^\circ}}}%
\renewrobustcmd{\LWR@siunitx@textprime}{\text{\ensuremath{^\prime}}}%
\renewrobustcmd{\LWR@siunitx@textdblprime}{\text{\ensuremath{^{\prime\prime}}}}%
\renewrobustcmd{\LWR@siunitx@textplanckbar}{\text{\ensuremath{\hbar}}}%
}

\PassOptionsToPackage{
    detect-mode=true,
    per-mode=symbol,% fraction is not seen by pdftotext
    text-celsius = {\LWR@siunitx@textcelsius},
    text-degree = {\LWR@siunitx@textdegree},
    text-arcminute = {\LWR@siunitx@textprime} ,
    text-arcsecond = {\LWR@siunitx@textdblprime} ,
}{siunitx}

\end{warpHTML}
```

83 Graphics print-mode modifications

83.1 General limitations

⚠ scale

Avoid using the `\includegraphics` `scale` option. Change:

```
\includegraphics[scale=<xx>]{ . . . }
```

to:

```
\includegraphics[width=<yy>\linewidth]{ . . . }
```

\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
```

Prog pdftocairo

PDF to SVG

To convert a PDF image to SVG, use the utility *pdftocairo*:

Enter ⇒ **`pdftocairo -svg filename.pdf`**

Prog lwarpmk pdftosvg

For a large number of images, use *lwarpmk*:

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

Prog lwarpmk epstopdf

Prog epstopdf

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.

⚠ 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 viewports are not supported by lwarp — the entire image will be shown.

⚠ viewport

⚠ viewport 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.

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 of "`(image)`".

`\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.

83.2 Print-mode modifications

for PRINT output:

For print output, accept and then discard the new `class` key:

```
\begin{warpprint}
\define@key{Gin}{class}{}
\define@key{Gin}{alt}{}
```

Print-mode additions for the overpic package. See section 333 for the HTML version.

```
\AtBeginDocument{
\@ifpackageloaded{overpic}{
\newcommand*{\overpicfontsize}{12}
\newcommand*{\overpicfontskip}{14}
}{}
}
\end{warpprint}
```

84 xcolor boxes

Pkg xcolor 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:
```
\begin{warpall}
```

After xparse may have been loaded …

```
\AtBeginDocument{
```

… and *only* if xcolor was loaded:

```
\@ifpackageloaded{xcolor}{
\LWR@traceinfo{patching xcolor}
```

The print version:

\colorboxBlock \colorboxBlock is the same as \colorbox:

```
\LetLtxMacro\colorboxBlock\colorbox
```

The original definition is reused by the new versions:

```
\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.)

```
\newsavebox{\LWR@colorminipagebox}

\NewDocumentCommand{\LWR@print@fcolorbox}{o m o m +m}{%
\LWR@traceinfo{LWR@print@fcolorbox #2 #4}%
```

Pre-load the contents into an LR box so that they can be used inside a \fcolorbox:

```
\begin{lrbox}{\LWR@colorminipagebox}%
#5%
\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.

```
\ifstrequal{#4}{none}%
{% #4 none
    \LWR@traceinfo{background is none}%
    {% scope the \colorlet
        \colorlet{LWR@currentcolor}{.}%
        \color{#2}%
        \fbox{%
            \color{LWR@currentcolor}%
            \usebox{\LWR@colorminipagebox}%
        }% fbox
    }% colorlet
}% #4 none
{% #4 not none
\LWR@traceinfo{background not none}%
\IfValueTF{#1}%
{%
    \IfValueTF{#3}%
   {\LWR@orig@print@fcolorbox[#1]{#2}[#3]{#4}{\usebox{\LWR@colorminipagebox}}}%
    {\LWR@orig@print@fcolorbox[#1]{#2}{#4}{\usebox{\LWR@colorminipagebox}}}%
}%
{% no value #1
    \IfValueTF{#3}%
    {\LWR@orig@print@fcolorbox{#2}[#3]{#4}{\usebox{\LWR@colorminipagebox}}}%
    {\LWR@orig@print@fcolorbox{#2}{#4}{\usebox{\LWR@colorminipagebox}}}%
}% no value #1
}% #4 not none
\LWR@traceinfo{LWR@print@fcolorbox done}%
}

\renewcommand*{\fcolorbox}{\LWR@print@fcolorbox}
```

\fcolorboxBlock [⟨*framemodel*⟩] {⟨*framecolor*⟩} [⟨*boxmodel*⟩] {⟨*boxcolor*⟩} {⟨*text*⟩}

In print mode, \fcolorboxBlock is the same as \fcolorbox.

```
\newcommand*{\LWR@print@fcolorboxBlock}{\LWR@print@fcolorbox}
\newcommand*{\fcolorboxBlock}{\LWR@print@fcolorboxBlock}
```

Env fcolorminipage [⟨*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:

```
\NewDocumentEnvironment{LWR@print@fcolorminipage}{o m o m O{c} O{} o m}
{%
\LWR@traceinfo{*** fcolorminipage: #2 #4 #8}%
```

Pre-load the contents into an LR box so that they can be used inside a \fcolorbox:

```
\begin{lrbox}{\LWR@colorminipagebox}%
```

If inner alignment is not given, use the outer alignment instead:

```
\IfValueTF{#7}%
{\begin{minipage}[#5][#6][#7]{#8}}%
{\begin{minipage}[#5][#6][#5]{#8}}%
}%
{%
\end{minipage}%
\end{lrbox}%
\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.

```
\ifstrequal{#4}{none}%
{% #4 none
    {% scope the \colorlet
        \colorlet{LWR@currentcolor}{.}%
        \color{#2}%
        \fbox{%
            \color{LWR@currentcolor}%
            \usebox{\LWR@colorminipagebox}%
        }% fbox
    }% colorlet
}% #4 none
{% #4 not none
    \IfValueTF{#1}%
    {%
    \IfValueTF{#3}%
   {\LWR@orig@print@fcolorbox[#1]{#2}[#3]{#4}{\usebox{\LWR@colorminipagebox}}}%
    {\LWR@orig@print@fcolorbox[#1]{#2}{#4}{\usebox{\LWR@colorminipagebox}}}%
    }%
    {% no value #1
    \IfValueTF{#3}%
    {\LWR@orig@print@fcolorbox{#2}[#3]{#4}{\usebox{\LWR@colorminipagebox}}}%
    {\LWR@orig@print@fcolorbox{#2}{#4}{\usebox{\LWR@colorminipagebox}}}%
    }% no value #1
}% #4 not none
\LWR@traceinfo{*** finished end fcolorminipage}%
}

\newenvironment*{fcolorminipage}
    {\LWR@print@fcolorminipage}
    {\endLWR@print@fcolorminipage}

\LWR@traceinfo{xcolor patches done}
}{}% xcolor loaded
}% AtBeginDocument
```

```
\end{warpall}
```

85 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:

```
\begin{warpprint}

\AtBeginDocument{
\@ifpackageloaded{chemmacros}{
```

Env polymerdelims

```
\DeclareDocumentEnvironment{polymerdelims}{}
    {}{}
```

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 163.4.

```
\DeclareDocumentEnvironment{redoxreaction}{m m}
    {\rule{0pt}{#1}}{\rule[-#2]{0pt}{#2}}

}{}% chemmacros
}% AtBeginDocument

\end{warpprint}
```

86 cleveref

Pkg cleveref cleveref package is used as-is with minor patches.

⚠ 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 529 to redefine the message which is printed for page number references.

loading order

cleveref and the following 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.

Table 12 on page 451 shows the data structure of the label/reference system as revised by lwarp and cleveref.

A few patches allow cleveref to work as-is:

for HTML output:

```
\begin{warpHTML}
```

`\AtEndPreable` forces cleveref to be loaded last:

```
\AtEndPreamble{
\RequirePackage{cleveref}
}
```

The following patches are applied after cleveref has loaded, and after `\AtBeginDocument`. Print-mode versions are not required since they all come down to `\ref` eventually, and `\ref` has a print-mode version.

```
\AfterEndPreamble{
\LWR@traceinfo{Patching cleveref.}
```

`\@@@setcref` {⟨*kindofref*⟩} {⟨*label*⟩}

`\@templabel` becomes the section number.

```
\def\LWR@orig@@@setcref#1#2{\cref@getlabel{#2}{\@templabel}#1{\@templabel}{}{}}%

\ifdefequal{\@@setcref}{\LWR@orig@@@setcref}{% before v0.21
    \renewcommand*{\@@setcref}[2]{#1{\ref{#2}}{}{}}
}{
    \ifdefequal{\@@@setcref}{\LWR@orig@@@setcref}{% as of v0.21
        \renewcommand*{\@@@setcref}[2]{#1{\ref{#2}}{}{}}
    }{
        \PackageWarning{lwarp-cleveref}{
            Unknown version of cleveref.
            \protect\cref\space will fail.
        }%
```

```
    }
}
```

`\@@@setcrefrange` {⟨*text*⟩} {⟨*label*⟩} {⟨*label*⟩}

```
\def\LWR@orig@@@setcrefrange#1#2#3{%
  \cref@getlabel{#2}{\@labela}%
  \cref@getlabel{#3}{\@labelb}%
  #1{\@labela}{\@labelb}{}{}{}{}}%

\ifdefequal{\@@setcrefrange}{\LWR@orig@@@setcrefrange}{
    \renewcommand{\@@setcrefrange}[3]{%
        #1{\ref{#2}}{\ref{#3}}{}{}{}{}%
    }
}{
    \ifdefequal{\@@@setcrefrange}{\LWR@orig@@@setcrefrange}{
        \renewcommand{\@@@setcrefrange}[3]{%
            #1{\ref{#2}}{\ref{#3}}{}{}{}{}%
        }
    }{
        \PackageWarning{lwarp-cleveref}{
            Unknown version of cleveref.
            \protect\crefrange\space will fail.
        }
    }
}

```

`\cpagerefFor` Redefinable word between “page(s)” and the page numbers.

```
\newcommand*{\cpagerefFor}{for}
```

`\@@@setcpageref` {⟨*typeofref*⟩} {⟨*label*⟩}, where typeofref is “page” or “pages”

```
\def\LWR@orig@@setcpageref#1#2{% before v0.21
  \cref@getpageref{#2}{\@temppage}#1{\@temppage}{}{}}%

\def\LWR@orig@@@setcpageref#1#2{% as of v0.21
  \cpageref@getlabel{#2}{\@temppage}#1{\@temppage}{}{}}%

\ifdefequal{\@@setcpageref}{\LWR@orig@@setcpageref}{
    \renewcommand*{\@@setcpageref}[2]{%
        #1{\cpagerefFor\ \cref{#2}}{}{}%
    }
}{
    \ifdefequal{\@@@setcpageref}{\LWR@orig@@@setcpageref}{
        \renewcommand*{\@@@setcpageref}[2]{%
            #1{\cpagerefFor\ \cref{#2}}{}{}%
        }
    }
    {
        \PackageWarning{lwarp-cleveref}{
```

```
            Unknown version of cleveref.
            \protect\cpageref\space will fail.
        }
    }
}

\def\LWR@orig@@setcpagerefrange#1#2#3{% before v0.21
  \cref@getpageref{#2}{\@pagea}%
  \cref@getpageref{#3}{\@pageb}%
  #1{\@pagea}{\@pageb}{}{}{}{}}%

\def\LWR@orig@@@setcpagerefrange#1#2#3{% as of v0.21
  \cpageref@getlabel{#2}{\@pagea}%
  \cpageref@getlabel{#3}{\@pageb}%
  #1{\@pagea}{\@pageb}{}{}{}{}}%

\ifdefequal{\@@setcpagerefrange}{\LWR@orig@@setcpagerefrange}{
    \renewcommand*{\@@setcpagerefrange}[3]{%
        #1{\cpagerefFor\ \cref{#2}}{\cref{#3}}{}{}{}{}%
    }
}{
    \ifdefequal{\@@@setcpagerefrange}{\LWR@orig@@@setcpagerefrange}{
        \renewcommand*{\@@@setcpagerefrange}[3]{%
            #1{\cpagerefFor\ \cref{#2}}{\cref{#3}}{}{}{}{}%
        }
    }
    {
        \PackageWarning{lwarp-cleveref}{
            Unknown version of cleveref.
            \protect\cpagerefrange\space will fail.
        }
    }
}

}% AfterEndPreamble
```

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.

```
\LetLtxMacro\LWR@orig@label\label
\RenewDocumentCommand{\label}{}{\LWR@new@label}

\LetLtxMacro\LWR@orig@pageref\pageref
\RenewDocumentCommand{\pageref}{}{\LWR@new@pageref}
\end{warpHTML}
```

87 picture environment

Env picture The picture environment is enclosed inside a \lateximage.

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

Env `picture`

```
\BeforeBeginEnvironment{picture}{\begin{lateximage}[(picture)]}

\AfterEndEnvironment{picture}{\end{lateximage}}
```

```
\end{warpHTML}
```

88 Boxes and Minipages

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 . An HTML cannot appear inside a . While in a , `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 `\linewidth`, `\textwidth`, and `\textheight`, widths and heights in HTML are scaled proportionally to a 6×9 inch text area, and inside a `multicols` `\linewidth` is divided by the specified number of columns.

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 `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

`\UseMinipageWidths`
`\IgnoreMinipageWidths`

the local scope, and \IgnoreMinipageWidths may be used to tell lwarp to ignore the specified widths.

⚠ 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:
```
\begin{warpHTML}
```

88.1 Counters and lengths

Ctr LWR@minipagedepth Used to only reset the line width at the outermost minipage.

```
\newcounter{LWR@minipagedepth}
\setcounter{LWR@minipagedepth}{0}
```

Len \LWR@minipagewidth Used to convert the width into printable units.

```
\newlength{\LWR@minipagewidth}
```

Len \LWR@minipageheight Used to convert the height into printable units.

```
\newlength{\LWR@minipageheight}
```

88.2 Footnote handling

Also see section 58 for other forms of footnotes. Minipage footnotes are gathered in section 58.5, and then placed into the document in section 88.3.

88.3 Minipage handling

Bool LWR@minipagefullwidth Should the next minipage have no HTML width?

```
\newbool{LWR@minipagefullwidth}
\boolfalse{LWR@minipagefullwidth}
```

Bool LWR@forceminipagefullwidth 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.

```
\newbool{LWR@forceminipagefullwidth}
\boolfalse{LWR@forceminipagefullwidth}
```

\minipagefullwidth Requests that the next minipage have no width tag in HTML:

for HTML output:
```
\newcommand*{\minipagefullwidth}{\global\booltrue{LWR@minipagefullwidth}}
```

\UseMinipageWidths Locally requests that minipage widths be honored.

```
\newcommand*{\UseMinipageWidths}{\boolfalse{LWR@forceminipagefullwidth}}
```

\IgnoreMinipageWidths Locally requests that minipage widths be honored.

```
\newcommand*{\IgnoreMinipageWidths}{\booltrue{LWR@forceminipagefullwidth}}
\end{warpHTML}
```

for PRINT output:

```
\begin{warpprint}
\newcommand*{\minipagefullwidth}{}
\newcommand*{\UseMinipageWidths}{}
\newcommand*{\IgnoreMinipageWidths}{}
\end{warpprint}
```

for HTML output:

```
\begin{warpHTML}
```

Bool LWR@minipagethispar Has a minipage been seen this paragraph? If true, prevents paragraph tags around horizontal space between minipages.

```
\newbool{LWR@minipagethispar}
\boolfalse{LWR@minipagethispar}
```

Env minipage [⟨*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.

```
\NewDocumentEnvironment{LWR@HTML@sub@minipage}{m m m m}
{%
\LWR@traceinfo{minipage}%
```

Temporarily open a group, 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.

The following used to be an actual LaTeX minipage.

```
\begingroup
```

Compute width, adjusted for frames:

```
\setlength{\LWR@minipagewidth}{#4}%
\ifthenelse{\cnttest{\value{LWR@minipagedepth}}{=}{0}}{%
```

Only create a new page if not yet nested:

```
    \LWR@orignewpage%
```

Adjust virtual page size:

```
    \addtolength{\LWR@minipagewidth}{3em}% room for frames
    \setlength{\linewidth}{6in}%
    \setlength{\textwidth}{6in}%
    \setlength{\textheight}{9in}%
}{}%
\LWR@traceinfo{computed width is \LWR@printlength{\LWR@minipagewidth}}%
```

Compute height:

```
\setlength{\LWR@minipageheight}{\textheight}% default unless specified
\ifblank{#2}{}{\setlength{\LWR@minipageheight}{#2}}%
```

Track nesting depth:

```
\addtocounter{LWR@minipagedepth}{1}%
```

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.

```
\ifbool{FormatWP}{\newline}{}%
\LWR@stoppars%
```

If `FormatWP`, add a text frame:

```
\ifbool{FormatWP}{%

\addtocounter{LWR@thisautoidWP}{1}%
\LWR@htmltag{%
    div id="\LWR@print@mbox{autoidWP-\arabic{LWR@thisautoidWP}}" %
    class="wpminipage"%
}%

}{}%
```

Create the tag with optional alignment style:

```
\LWR@traceinfo{minipage: creating div class}%
\LWR@htmltag{div class="minipage" style="%
\ifthenelse{\equal{#1}{t}}{\LWR@print@mbox{vertical-align:bottom} ; }{}%
\ifthenelse{\equal{#1}{c}}{\LWR@print@mbox{vertical-align:middle} ; }{}%
\ifthenelse{\equal{#1}{b}}{\LWR@print@mbox{vertical-align:top} ; }{}%
\ifthenelse{\equal{#3}{t}}{\LWR@print@mbox{justify-content:flex-start} ; }{}%
\ifthenelse{\equal{#3}{c}}{\LWR@print@mbox{justify-content:center} ; }{}%
\ifthenelse{\equal{#3}{b}}{\LWR@print@mbox{justify-content:flex-end} ; }{}%
\ifthenelse{\equal{#3}{s}}{\LWR@print@mbox{justify-content:space-between} ; }{}%
```

Print the width and optional height styles:

```
\LWR@traceinfo{minipage: about to print the width of \LWR@printlength{\LWR@minipagewidth}}%
\ifbool{LWR@minipagefullwidth}%
{\global\boolfalse{LWR@minipagefullwidth}}%
{%
    \ifbool{LWR@forceminipagefullwidth}%
```

```
        {}%
        {%
            \ifthenelse{\lengthtest{#4}=\linewidth}%
                {}%
                {width:\LWR@printlength{\LWR@minipagewidth} ; }%
        }%
}%
\LWR@traceinfo{minipage: about to print the height}%
\ifblank{#2}{}{height:\LWR@printlength{\LWR@minipageheight} ; }%
"}%
```

Finish with an empty line to start the contents on a new line.

```

% 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.

```
\setlength{\linewidth}{#4}% the original width
\setlength{\textwidth}{6in}%
\setlength{\textheight}{9in}%
```

`\raggedright` cancels hyphenation, which will be done by HTML instead.

```
\LWR@print@raggedright%
```

Set minipage footnotes:

```
\def\@mpfn{mpfootnote}%
\def\thempfn{\thempfootnote}\c@mpfootnote\z@%
\let\@footnotetext\@mpfootnotetext%
```

Resume paragraph tag handling for the contents of the minipage:

```
\LWR@startpars%
\ifboolexpr{bool{FormatWP} and bool{WPMarkMinipages}}{%

=== begin minipage ===

}{}%
\LWR@traceinfo{minipage: finished starting the minipage}%
}% finished \minipage
{% \endminipage
```

Print pending minipage footnotes:

```
\LWR@printpendingmpfootnotes%
```

End the environment with closing tag:

```
\ifboolexpr{bool{FormatWP} and bool{WPMarkMinipages}}{%

```

```
=== end minipage ===

}{}%
\LWR@stoppars%
```

The following used to be an actual LaTeX minipage.

```
\endgroup%

\ifbool{FormatWP}{%

\LWR@htmlelementend{div}%

}{}%
\LWR@htmldivclassend{minipage}%

\addtocounter{LWR@minipagedepth}{-1}%
\LWR@startpars%
\ifbool{FormatWP}{\newline}{}%
```

Prevent paragraph tags around horizontal white space until the start of the next paragraph:

```
\global\booltrue{LWR@minipagethispar}%
\LWR@traceinfo{LWR@minipage: done}%
}

\NewDocumentEnvironment{LWR@HTML@minipage}{O{t} O{} O{t} m}
    {\LWR@HTML@sub@minipage{#1}{#2}{#3}{#4}}
    {\endLWR@HTML@sub@minipage}

\LWR@formattedenv{minipage}
```

88.4 \parbox, \mbox, \makebox, \framebox, \fbox, \raisebox

for HTML output:

\parbox [⟨*pos*⟩] [⟨*height*⟩] [⟨*inner-pos*⟩] {⟨*width*⟩} {⟨*text*⟩}

A parbox uses the minipage code:

```
\NewDocumentCommand{\LWR@HTML@parbox}{O{t} O{} O{t} m +m}
{
\LWR@traceinfo{parbox of width #4}%
\begin{minipage}[#1][#2][#3]{#4}%
#5
\end{minipage}%
}

\LWR@formatted{parbox}
```

`\mbox` {⟨*text*⟩} Nullified for HTML.

```
\newcommand*{\LWR@HTML@mbox}[1]{#1}

\LWR@formatted{mbox}
```

`\makebox` (⟨(⟩)posn) [⟨*width*⟩] [⟨*pos*⟩] {⟨*text*⟩}

```
\NewDocumentCommand{\LWR@HTML@makebox}{d() o o m}{%
```

Check for the optional width:

```
\IfValueTF{#2}%
{%
```

Check for the horizontal text alignment. For stretched, the best HTML can do is `justified` alignment.

```
    {% scope
    \def\LWR@align{center}%
    \ifstrequal{#3}{l}{\def\LWR@align{left}}{}%
    \ifstrequal{#3}{r}{\def\LWR@align{right}}{}%
    \ifstrequal{#3}{s}{\def\LWR@align{justify}}{}%
```

To print the `width` argument:

```
    \setlength{\LWR@tempwidth}{#2}%
```

`inline-block` allows `width` and `text-alignment` to be used in a .

```
    \InlineClass[%
        \LWR@print@mbox{display:inline-block} ; %
        \LWR@print@mbox{text-align}:\LWR@align\ ; %
        width:\LWR@printlength{\LWR@tempwidth}%
    ]%
    {makebox}%


    {#4}%
    }% scope
}%
```

Without a width argument, the text is simply used inline:

```
{#4}% no width
}

\LWR@formatted{makebox}
```

`\framebox` [⟨*width*⟩] [⟨*pos*⟩] {⟨*text*⟩}

```
\LetLtxMacro\LWR@origframebox\framebox
```

```
10436
10437 \NewDocumentCommand{\LWR@HTML@framebox}{o o m}{%
10438 \fbox{\makebox[#1][#2]{#3}}%
10439 }
10440
10441 \LWR@formatted{framebox}
```

\LWR@forceminwidth {⟨*legth*⟩}

Sets \LWR@atleastonept to be at least 1pt.

```
10442 \newlength{\LWR@atleastonept}
10443
10444 \newcommand*{\LWR@forceminwidth}[1]{%
10445 \setlength{\LWR@atleastonept}{#1}%
10446 \ifthenelse{%
10447     \lengthtest{\LWR@atleastonept>0pt}\AND%
10448     \lengthtest{\LWR@atleastonept<1pt}%
10449 }%
10450     {\setlength{\LWR@atleastonept}{1pt}}%
10451     {}%
10452 }
```

\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.

```
10453 \newcommand*{\LWR@fboxstyle}{%
10454 \LWR@findcurrenttextcolor%
10455 border:\LWR@printlength{\LWR@atleastonept} solid \LWR@origpound\LWR@tempcolor ; %
10456 padding:\LWR@printlength{\fboxsep} ; %
10457 color:\LWR@origpound\LWR@tempcolor%
10458 }
```

\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:

```
10459 \newcommand{\LWR@HTML@fbox}[1]{%
10460 \LWR@traceinfo{HTML fbox}%
10461 \LWR@forceminwidth{\fboxrule}%
10462 \InlineClass[%
10463 \LWR@fboxstyle%
10464 ]{fbox}{#1}%
10465 }
```

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:

```
10466 \AtBeginDocument{\LWR@formatted{fbox}}
```

\fboxBlock {⟨*text*⟩} Creates a framed HTML of the text.

First, a print-mode version. This is newly defined for print mode, so it is defined inside warpall.

for HTML & PRINT:
```
\end{warpHTML}
\begin{warpall}
\let\fboxBlock\fbox
\end{warpall}
\begin{warpHTML}
```

for HTML output: Next, an HTML version:

```
\newcommand{\LWR@HTML@fboxBlock}[1]{%
\LWR@forceminwidth{\fboxrule}%
\begin{BlockClass}[%
\LWR@fboxstyle%
]{fboxBlock}
#1
\end{BlockClass}
}

\LWR@formatted{fboxBlock}

\end{warpHTML}
```

Env fminipage [⟨*align*⟩] [⟨*height*⟩] [⟨*align*⟩] {⟨*width*⟩}

Creates a framed HTML around its contents.

for HTML & PRINT: Print version:

```
\begin{warpall}

\newsavebox{\LWR@fminipagebox}

\NewDocumentEnvironment{LWR@print@fminipage}{O{t} o O{t} m}
{%
```

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:

```
\IfValueTF{#3}%
{\def\LWR@thisalign{#3}}
{\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.

```
\IfValueTF{#2}%
{\minipage[#1][#2+2\fboxsep+2\fboxrule][\LWR@thisalign]{#4+2\fboxsep+2\fboxrule}}%
{\minipage[#1]{#4+2\fboxsep+2\fboxrule}}%
```

Capture the contents of the environment:

```
\begin{lrbox}{\LWR@fminipagebox}%
```

Nest the contents inside an inner minipage of the desired size:

```
\IfValueTF{#2}%
{\minipage[#1][#2][\LWR@thisalign]{#4}}%
{\minipage[#1]{#4}}%
}
{%
```

Close the inner minipage and the LR box with the contents:

```
\endminipage%
\end{lrbox}%
```

Create a frame around the contents of the environment:

```
\fbox{\usebox{\LWR@fminipagebox}}%
```

The entire thing is placed inside the outer minipage:

```
\endminipage%
}

\LetLtxMacro\fminipage\LWR@print@fminipage
\LetLtxMacro\endfminipage\endLWR@print@fminipage
% \newenvironment{fminipage}{\LWR@print@fminipage}{\endLWR@print@fminipage}

\end{warpall}
```

for HTML output:

```
\ifLWR@isHTML
\NewDocumentEnvironment{LWR@HTML@fminipage}{O{t} o O{t} m}
{%
\LWR@traceinfo{fminipage #1 #2 #3 #4}%
\LWR@forceminwidth{\fboxrule}%
\setlength{\LWR@tempwidth}{#4}%
\IfValueT{#2}{\setlength{\LWR@tempheight}{#2}}%

\LWR@stoppars%

\begin{BlockClass}[%
\LWR@fboxstyle ; %
\IfValueT{#2}{height:\LWR@printlength{\LWR@tempheight} ; }%
width:\LWR@printlength{\LWR@tempwidth}%
]{fminipage}%
}
{%
\end{BlockClass}%
```

Prevent paragraph tags around horizontal white space until the start of the next paragraph:

```
\global\booltrue{LWR@minipagethispar}%
\LWR@traceinfo{fminipage done}%
}

\LWR@formattedenv{fminipage}

\end{warpHTML}
```

`\raisebox` `{⟨raiselen⟩}` `[⟨height⟩]` `[⟨depth⟩]` `{⟨text⟩}`

```
\begin{warpHTML}

\NewDocumentCommand{\LWR@HTML@raisebox}{m o o m}{%
#4%
}

\LWR@formatted{raisebox}

\end{warpHTML}
```

89 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 51.9.

for HTML output:

```
\begin{warpHTML}
```

`\emph` `{⟨text⟩}`

```
\DeclareRobustCommand{\LWR@HTML@emph}[1]{%
    {%
        \itshape%
        \LWR@htmlspan{em}{#1}%
    }%
}
```

```
\LWR@formatted{emph}

\DeclareRobustCommand{\LWR@null@emph}[1]{#1}
```

\textmd {⟨*text*⟩}

```
\DeclareRobustCommand{\LWR@HTML@textmd}[1]{%
    {%
        \mdseries%
        \InlineClass(font-weight:normal){textmd}{#1}%
    }%
}
\LWR@formatted{textmd}

\DeclareRobustCommand{\LWR@null@textmd}[1]{#1}
```

\textbf {⟨*text*⟩}

```
\DeclareRobustCommand{\LWR@HTML@textbf}[1]{%
    {%
        \bfseries%
        \LWR@htmlspan{b}{#1}%
    }%
}
\LWR@formatted{textbf}

\DeclareRobustCommand{\LWR@null@textbf}[1]{#1}
```

\textrm {⟨*text*⟩}

```
\DeclareRobustCommand{\LWR@HTML@textrm}[1]{%
    {%
        \rmfamily%
        \InlineClass(font-family:serif){textrm}{#1}%
    }%
}
\LWR@formatted{textrm}

\DeclareRobustCommand{\LWR@null@textrm}[1]{#1}
```

\textsf {⟨*text*⟩}

```
\DeclareRobustCommand{\LWR@HTML@textsf}[1]{%
    {%
        \sffamily%
        \InlineClass(font-family:sans){textsf}{#1}%
    }%
}
\LWR@formatted{textsf}

\DeclareRobustCommand{\LWR@null@textsf}[1]{#1}
```

\texttt {⟨*text*⟩}

```
\DeclareRobustCommand{\LWR@HTML@texttt}[1]{%
    {%
        \ttfamily%
        \LWR@htmlspan{kbd}{#1}%
    }%
}
\LWR@formatted{texttt}

\DeclareRobustCommand{\LWR@null@texttt}[1]{#1}
```

\textup {⟨*text*⟩}

```
\DeclareRobustCommand{\LWR@HTML@textup}[1]{%
    {%
        \upshape%
        \InlineClass(font-style:normal){textup}{#1}%
    }%
}
\LWR@formatted{textup}

\DeclareRobustCommand{\LWR@null@textup}[1]{#1}
```

\textit {⟨*text*⟩}

```
\DeclareRobustCommand{\LWR@HTML@textit}[1]{%
    {%
        \itshape%
        \LWR@htmlspan{i}{#1}%
    }%
}
\LWR@formatted{textit}

\DeclareRobustCommand{\LWR@null@textit}[1]{#1}
```

\textsc {⟨*text*⟩}

```
\DeclareRobustCommand{\LWR@HTML@textsc}[1]{%
    {%
        \scshape%
        \InlineClass(
            font-variant: small-caps ;
            font-variant-numeric: oldstyle-nums ;
        ){textsc}{#1}%
    }%
}
\LWR@formatted{textsc}

\DeclareRobustCommand{\LWR@null@textsc}[1]{#1}
```

\textsi {⟨*text*⟩}

```
\@ifundefined{textsi}{
    \LetLtxMacro\LWR@print@textsi\LWR@print@textsc
}{}

\DeclareRobustCommand{\LWR@HTML@textsi}[1]{%
    {%
        \sishape%
        \InlineClass(
            font-style: italic;
            font-variant: small-caps ;
            font-variant-numeric: oldstyle-nums ;
        ){textsi}{#1}%
    }%
}
\LWR@formatted{textsi}

\DeclareRobustCommand{\LWR@null@textsi}[1]{#1}
```

`\textsl` {⟨*text*⟩}

```
\DeclareRobustCommand{\LWR@HTML@textsl}[1]{%
    {%
        \slshape%
        \InlineClass(font-style:oblique){textsl}{#1}%
    }%
}
\LWR@formatted{textsl}

\DeclareRobustCommand{\LWR@null@textsl}[1]{#1}
```

`\textnormal` {⟨*text*⟩}

```
\DeclareRobustCommand{\LWR@HTML@textnormal}[1]{\textmd{\textrm{\textup{#1}}}}
\LWR@formatted{textnormal}

\DeclareRobustCommand{\LWR@null@textnormal}[1]{#1}

\DeclareRobustCommand{\LWR@null@rmfamily}{}
\DeclareRobustCommand{\LWR@null@sffamily}{}
\DeclareRobustCommand{\LWR@null@ttfamily}{}
\DeclareRobustCommand{\LWR@null@bfseries}{}
\DeclareRobustCommand{\LWR@null@mdseries}{}
\DeclareRobustCommand{\LWR@null@upshape}{}
\DeclareRobustCommand{\LWR@null@slshape}{}
\DeclareRobustCommand{\LWR@null@scshape}{}
\DeclareRobustCommand{\LWR@null@itshape}{}
\DeclareRobustCommand{\LWR@null@normalfont}{}


\DeclareRobustCommand{\LWR@null@em}{}
```

`\LWR@nullfonts` Removes formatting during filename operations, file references, and HTML comments.

⚠ Use only inside a group.

The following are *not* made robust, since they must be expanded to their nullified versions.

```
\catcode`\$=\active% redefining $ below
\catcode`\_=12% redefining \_ below
\newcommand*{\LWR@nullfonts}{%
\LetLtxMacro\emph\LWR@null@emph%
\LetLtxMacro\textmd\LWR@null@textmd%
\LetLtxMacro\textbf\LWR@null@textbf%
\LetLtxMacro\textrm\LWR@null@textrm%
\LetLtxMacro\textsf\LWR@null@textsf%
\LetLtxMacro\texttt\LWR@null@texttt%
\LetLtxMacro\textup\LWR@null@textup%
\LetLtxMacro\textit\LWR@null@textit%
\LetLtxMacro\textsc\LWR@null@textsc%
\LetLtxMacro\textsi\LWR@null@textsi%
\LetLtxMacro\textsl\LWR@null@textsl%
\LetLtxMacro\textnormal\LWR@null@textnormal%
\LetLtxMacro\rmfamily\LWR@null@rmfamily%
\LetLtxMacro\sffamily\LWR@null@sffamily%
\LetLtxMacro\ttfamily\LWR@null@ttfamily%
\LetLtxMacro\bfseries\LWR@null@bfseries%
\LetLtxMacro\mdseries\LWR@null@mdseries%
\LetLtxMacro\upshape\LWR@null@upshape%
\LetLtxMacro\slshape\LWR@null@slshape%
\LetLtxMacro\scshape\LWR@null@scshape%
\LetLtxMacro\sishape\LWR@null@sishape%
\LetLtxMacro\itshape\LWR@null@itshape%
\LetLtxMacro\normalfont\LWR@null@normalfont%
\LetLtxMacro\em\LWR@null@em%
```

Various built-in symbols.

```
\renewcommand*{\$}{-}%
\renewcommand*{\%}{-}%
\renewcommand*{\_}{-}%
\renewcommand*{\}}{-}%
\renewcommand*{\{}{-}%
\renewcommand*{\&}{and}%
\renewcommand*{\#}{-}%
\renewcommand*{\,}{-}%
\renewcommand*{~}{-}%
\renewcommand*{\^}[1]{-}%
\renewcommand*{\~}[1]{-}%
\renewcommand*{\newline}{-}%
\renewcommand*{\textasciicircum}{-}%
\renewcommand*{\textasciitilde}{-}%
\renewcommand*{\textasteriskcentered}{-}%
\renewcommand*{\textbackslash}{-}%
\renewcommand*{\textbar}{-}%
\renewcommand*{\textbardbl}{-}%
\renewcommand*{\textbigcircle}{-}%
```

```
\renewcommand*{\textbraceleft}{-}%
\renewcommand*{\textbraceright}{-}%
\renewcommand*{\textbullet}{-}%
\renewcommand*{\textcopyright}{-}%
\renewcommand*{\textdagger}{-}%
\renewcommand*{\textdaggerdbl}{-}%
\renewcommand*{\textdollar}{-}%
\renewcommand*{\textellipsis}{-}%
\renewcommand*{\textemdash}{-}%
\renewcommand*{\textendash}{-}%
\renewcommand*{\textexclamdown}{-}%
\renewcommand*{\textgreater}{-}%
\renewcommand*{\textless}{-}%
\renewcommand*{\textordfeminine}{-}%
\renewcommand*{\textordmasculine}{-}%
\renewcommand*{\textparagraph}{-}%
\renewcommand*{\textperiodcentered}{-}%
\renewcommand*{\textpertenthousand}{-}%
\renewcommand*{\textperthousand}{-}%
\renewcommand*{\textquestiondown}{-}%
\renewcommand*{\textquotedblleft}{-}%
\renewcommand*{\textquotedblright}{-}%
\renewcommand*{\textquoteleft}{-}%
\renewcommand*{\textquoteright}{-}%
\renewcommand*{\textregistered}{-}%
\renewcommand*{\textsection}{-}%
\renewcommand*{\textsterling}{-}%
\renewcommand*{\texttrademark}{-}%
\renewcommand*{\textunderscore}{-}%
\renewcommand*{\textvisiblespace}{-}%
\renewcommand*{\copyright}{-}%
\renewcommand*{\dag}{-}%
\renewcommand*{\ddag}{-}%
\renewcommand*{\dots}{-}%
\renewcommand*{\P}{-}%
\renewcommand*{\pounds}{-}%
\renewcommand*{\S}{-}%
\renewcommand*{\aa}{a}%
\renewcommand*{\AA}{A}%
\renewcommand*{\AE}{AE}%
\renewcommand*{\ae}{ae}%
\renewcommand*{\dh}{d}%
\renewcommand*{\DH}{D}%
\renewcommand*{\DJ}{D}%
\renewcommand*{\dj}{d}%
\renewcommand*{\IJ}{IJ}%
\renewcommand*{\ij}{ij}%
\renewcommand*{\L}{L}%
\renewcommand*{\l}{l}%
\renewcommand*{\NG}{NG}%
\renewcommand*{\ng}{ng}%
\renewcommand*{\O}{O}%
\renewcommand*{\o}{o}%
\renewcommand*{\oe}{oe}%
\renewcommand*{\OE}{OE}%
```

```
\renewcommand*{\ss}{ss}%
\renewcommand*{\SS}{SS}%
\renewcommand*{\th}{th}%
\renewcommand*{\TH}{TH}%
\renewcommand*{\guillemotleft}{}%
\renewcommand*{\guilsinglleft}{}%
\renewcommand*{\quotedblbase}{}%
\renewcommand*{\textquotedbl}{}%
\renewcommand*{\guillemotright}{}%
\renewcommand*{\guilsinglright}{}%
\renewcommand*{\quotesinglbase}{}%

\renewcommand*{\HTMLunicode}[1]{}%
\renewcommand*{\HTMLentity}[1]{}%

\renewcommand{\textsuperscript}[1]{##1}%
\renewcommand{\textsubscript}[1]{##1}%


\renewcommand{\underline}[1]{##1}%

\RenewDocumentCommand{\LWR@htmlspanclass}{o m +m}{##3}%
\DeclareExpandableDocumentCommand{\InlineClass}{D{(}{)}{} o m +m}{##4}%
```

Nullify math macros.

```
\def\(##1\){}%
\def\[##1\]{}%
\RenewDocumentCommand{\LWR@subsingledollar}{s m m m}{}%
\protected\def$##1${}%
```

Nullify logos:

```
\renewcommand*{\TeX}{TeX}%
\renewcommand*{\LaTeX}{LaTeX}%
\renewcommand*{\LaTeXe}{LaTeX2e}%
\renewcommand*{\LuaTeX}{LuaTeX}%
\renewcommand*{\LuaLaTeX}{LuaLaTeX}%
\renewcommand*{\XeTeX}{XeTeX}%
\renewcommand*{\XeLaTeX}{XeLaTeX}%
\renewcommand*{\ConTeXt}{ConTeXt}%
\renewcommand*{\BibTeX}{BibTeX}%
\renewcommand*{\MakeIndex}{MakeIndex}%
\renewcommand*{\AmS}{AmS}%
\renewcommand*{\MiKTeX}{MiKTeX}%
\renewcommand*{\LyX}{LyX}%
```

Use the simpler form with `\texorpdfstring`:

```
\let\texorpdfstring\relax%
\newcommand{\texorpdfstring}[2]{##2}%
}
\catcode`\$=3%
\catcode`\_=8%
```

\FilenameNullify {⟨*redefinitions*⟩}

Adds more nullifying definitions for filename generation.

```
\newcommand*{\FilenameNullify}[1]{%
    \appto{\LWR@nullfonts}{#1}%
}
```

Remembers the current font family, series, and shape.

```
\newcommand*{\LWR@f@family}{rm}
\newcommand*{\LWR@f@series}{md}
\newcommand*{\LWR@f@shape}{up}
```

\LWR@textcurrentfont {⟨*text*⟩}

Prints the text with the current font choices. Avoids nesting repeated font selections.

```
\newcounter{LWR@textcurrentfontdepth}
\setcounter{LWR@textcurrentfontdepth}{0}

\newcommand*{\LWR@textcurrentfont}[1]{%
    \ifnumcomp{\value{LWR@textcurrentfontdepth}}{>}{0}%
        {%
            \addtocounter{LWR@textcurrentfontdepth}{1}%
            #1%
            \addtocounter{LWR@textcurrentfontdepth}{-1}%
        }%
        {%
            \addtocounter{LWR@textcurrentfontdepth}{1}%
            \InlineClass{%
                    text\LWR@f@family\LWR@origtilde{}%
                    text\LWR@f@series\LWR@origtilde{}%
                    text\LWR@f@shape%
                }%
                {#1}%
            \addtocounter{LWR@textcurrentfontdepth}{-1}%
        }%
}
```

Env LWR@blocktextcurrentfont Prints the contents with the current font choices.

```
\newenvironment*{LWR@blocktextcurrentfont}{%
\BlockClass{%
        text\LWR@f@family\LWR@origtilde{}%
        text\LWR@f@series\LWR@origtilde{}%
        text\LWR@f@shape%
    }%
}{\endBlockClass}
```

\mdseries

```
\renewrobustcmd*{\mdseries}{\renewcommand*{\LWR@f@series}{md}}
```

\bfseries

```
\renewrobustcmd*{\bfseries}{\renewcommand*{\LWR@f@series}{bf}}
```

\rmfamily

```
\renewrobustcmd*{\rmfamily}{\renewcommand*{\LWR@f@family}{rm}}
```

\sffamily

```
\renewrobustcmd*{\sffamily}{\renewcommand*{\LWR@f@family}{sf}}
```

\ttfamily

```
\renewrobustcmd*{\ttfamily}{\renewcommand*{\LWR@f@family}{tt}}
```

\upshape

```
\renewrobustcmd*{\upshape}{\renewcommand*{\LWR@f@shape}{up}}
```

\itshape

```
\renewrobustcmd*{\itshape}{\renewcommand*{\LWR@f@shape}{it}}
```

\scshape

```
\renewrobustcmd*{\scshape}{\renewcommand*{\LWR@f@shape}{sc}}
```

\sishape

```
\@ifundefined{sishape}{
    \newrobustcmd*{\sishape}{\renewcommand*{\LWR@f@shape}{si}}
}{
    \renewrobustcmd*{\sishape}{\renewcommand*{\LWR@f@shape}{si}}
}
```

\slshape

```
\renewrobustcmd*{\slshape}{\renewcommand*{\LWR@f@shape}{sl}}
```

\normalfont

```
\renewrobustcmd*{\normalfont}{\rmfamily\mdseries\upshape}
```

\sp {⟨*text*⟩}

For siunitx. Must work in math mode.

```
\renewcommand{\sp}[1]{\text{<sup>#1</sup>}{}}
```

\sb {⟨*text*⟩}

For siunitx. Must work in math mode.

```
\renewcommand{\sb}[1]{\text{<sub>#1</sub>}{}}
```

\textsuperscript {⟨*text*⟩}

```
\renewrobustcmd{\textsuperscript}[1]{\LWR@htmlspan{sup}{#1}}
```

\@textsuperscript {⟨*text*⟩}

```
\renewcommand{\@textsuperscript}[1]{\LWR@htmlspan{sup}{#1}}
```

\textsubscript {⟨*text*⟩}

```
\AtBeginDocument{
\renewrobustcmd{\textsubscript}[1]{\LWR@htmlspan{sub}{#1}}
}
```

\@textsubscript {⟨*text*⟩}

```
\AtBeginDocument{
\renewcommand{\@textsubscript}[1]{\LWR@htmlspan{sub}{#1}}
}
```

\up {⟨*text*⟩} Prints superscript.

This is \let at the beginning of the document in case some other package has changed the definition.

```
\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.

```
\AtBeginDocument{\let\fup\textsuperscript}
```

\underline {⟨*text*⟩}

```
\renewcommand{\underline}[1]{%
    \InlineClass%
        (text-decoration:underline; text-decoration-skip: auto)%
        {underline}{#1}%
}
```

\LWR@overline {⟨*text*⟩}

```
\newcommand{\LWR@overline}[1]{%
    \InlineClass%
        (text-decoration:overline; text-decoration-skip: auto)%
        {overline}{#1}%
}
```

\LWR@currenttextcolor The color to use for text and \rule, defaulting to black:

```
\newcommand*{\LWR@currenttextcolor}{black}
```

\LWR@tempcolor \LWR@tempcolortwo The color converted to HTML colorspace.

```
\newcommand*{\LWR@tempcolor}{}
\newcommand*{\LWR@tempcolortwo}{}
```

\LWR@findcurrenttextcolor Sets \LWR@tempcolor to the current color.

```
\newcommand*{\LWR@findcurrenttextcolor}{%
\renewcommand{\LWR@tempcolor}{000000}%
}
```

\LWR@textcurrentcolor {⟨*text*⟩} Like \textcolor but uses the current \color instead.

```
\NewDocumentCommand{\LWR@textcurrentcolor}{m}{%
    \renewcommand*{\LWR@currenttextcolor}{black}%
    #1%
}
```

```
\end{warpHTML}
```

for PRINT output:

```
\begin{warpprint}
```

\LWR@textcurrentfont {⟨*text*⟩}

Prints the text with the current font choices.

```
\newcommand*{\LWR@textcurrentfont}[1]{#1}
```

Env LWR@blocktextcurrentfont Prints the contents with the current font choices.

```
\newenvironment*{LWR@blocktextcurrentfont}{}{}
```

\FilenameNullify {⟨*macros to nullify*⟩}

```
\newcommand*{\FilenameNullify}[1]{}
```

```
\end{warpprint}
```

90 Skips, spaces, font sizes

for HTML output:

```
\begin{warpHTML}
```

`\,` and `\thinspace` may be redefined by other packages, so are redefined `\AtBeginDocument` here.

Direct-formatting space commands become HTML entities:

```
\AtBeginDocument{
\renewrobustcmd*{\,}{\HTMLunicode{202f}} % HTML thin non-breakable space
\renewrobustcmd*{\thinspace}{\HTMLunicode{202f}}   % HTML thin non-breakable space
\renewrobustcmd*{\negthinspace}{\HTMLunicode{202f}}   % HTML thin non-breakable space
\renewrobustcmd*{~}{\HTMLentity{nbsp}}
\renewrobustcmd*{\textellipsis}{\HTMLunicode{2026}}
}
```

Direct-formatting font sizes are ignored:

```
\newrobustcmd*{\LWR@HTML@normalsize}{}
\LWR@formatted{normalsize}

\newrobustcmd*{\LWR@HTML@small}{}
\LWR@formatted{small}

\newrobustcmd*{\LWR@HTML@footnotesize}{}
\LWR@formatted{footnotesize}

\newrobustcmd*{\LWR@HTML@scriptsize}{}
\LWR@formatted{scriptsize}

\newrobustcmd*{\LWR@HTML@tiny}{}
\LWR@formatted{tiny}

\newrobustcmd*{\LWR@HTML@large}{}
\LWR@formatted{large}

\newrobustcmd*{\LWR@HTML@Large}{}
\LWR@formatted{Large}

\newrobustcmd*{\LWR@HTML@LARGE}{}
\LWR@formatted{LARGE}

\newrobustcmd*{\LWR@HTML@huge}{}
\LWR@formatted{huge}

\newrobustcmd*{\LWR@HTML@Huge}{}
\LWR@formatted{Huge}


\DeclareDocumentCommand{\onecolumn}{}{}

\DeclareDocumentCommand{\twocolumn}{O{}}{
```

```
#1
}
```

\hfill

```
\newcommand*{\LWR@HTML@hfill}{\qquad}
\LWR@formatted{hfill}
```

\hrulefill

```
\newcommand*{\LWR@HTML@hrulefill}{\rule{1in}{1pt}}
\LWR@formatted{hrulefill}
```

\dotfill

```
\newcommand*{\LWR@HTML@dotfill}{\dots}
\LWR@formatted{dotfill}
```

\newpage

```
\renewcommand*{\newpage}{
}
```

\newline Uses the HTML <br /> element.

```
\newrobustcmd*{\LWR@newlinebr}{\unskip\LWR@htmltag{br /}\LWR@orignewline}%
\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.

```
\LetLtxMacro\LWR@origendofline\\
\NewDocumentCommand{\LWR@endofline}{s O{0pt}}
{%
\newline%

\setlength{\LWR@templengthone}{#2}%
\ifdimgreater{\LWR@templengthone}{0pt}{\newline}{}%
}
```

\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 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.

```
\newcommand*{\LWR@minipagestartpars}{%
\ifbool{LWR@minipagethispar}{\LWR@startpars}{}%
}
```

\LWR@minipagestoppars

Placed just after \hspace, \quad, or \qquad's HTML output.

```
\newcommand*{\LWR@minipagestoppars}{%
\ifbool{LWR@minipagethispar}{\LWR@stoppars}{}%
}
```

\quad

Handles special minipage & horizontal space interactions. Uses 2003 EM SPACE to pass validation.

```
\renewrobustcmd*{\quad}{%
\LWR@minipagestoppars%
\HTMLunicode{2003}%
\LWR@minipagestartpars%
}
```

\qquad

Handles special minipage & horizontal space interactions.

```
\renewrobustcmd*{\qquad}{\quad\quad}
```

\enskip

Handles special minipage & horizontal space interactions.

```
\renewrobustcmd*{\enskip}{%
\LWR@minipagestoppars%
\HTMLunicode{2002}%
\LWR@minipagestartpars%
}
```

Len \LWR@tempwidth
Len \LWR@tempheight
Len \LWR@tempraise

Used to compute span width, height, raise for \hspace and \rule:

```
\newlength{\LWR@tempwidth}
```

```
\newlength{\LWR@tempheight}
\newlength{\LWR@tempraise}
```

`\LWR@select@html@hspace` * {⟨*length*⟩} * {⟨*length*⟩}
`\hspace`

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`.

```
\newcommand{\LWR@select@html@hspace}{%
\RenewDocumentCommand{\hspace}{s m}{%
\setlength{\LWR@tempwidth}{##2}%
```

If `\fill`, change to `\qquad`:

```
\ifnum\gluestretchorder\LWR@tempwidth>0%
\setlength{\LWR@tempwidth}{2em}%
\fi%
```

Only if the width is not zero:

```
\ifdimcomp{\LWR@tempwidth}{=}{0pt}{}{%
```

If had a minipage this paragraph, try to inline the white space without generating paragraph tags:

```
    \LWR@minipagestoppars%
```

Support the HTML thin wrappable space:

```
    \ifdimcomp{\LWR@tempwidth}{=}{.16667em}%
    {%
        \HTMLunicode{2009}% thin breakable space
    }%
```

Print the span with the converted width. Not rounded.

```
    {%
        \LWR@htmltagc{%
            span style="width:\LWR@printlength{\LWR@tempwidth};  % extra space
                display:inline-block"%
        }%
```

If formatting for a word processor, approximate with a number of `\quad`s, in case a span of a given width is not supported:

```
        \ifbool{FormatWP}{%
            \setlength{\LWR@templengthone}{\LWR@tempwidth}%
            \whiledo{\lengthtest{\LWR@templengthone>1em}}{%
                \quad%
                \addtolength{\LWR@templengthone}{-1em}%
```

```
11002             }%
11003         }{}%
```

Close the span:

```
11004         \LWR@htmltagc{/span}%
11005     }%
```

If had a minipage this paragraph, try to inline the white space without generating paragraph tags:

```
11006     \LWR@minipagestartpars%
11007 }% width not 0
11008 }%
11009 }
```

`\LWR@select@html@nohspace` * {⟨*length*⟩}
`\hspace`

Used to disable `\hspace` while creating `description` `\item`s.

```
11010 \newcommand{\LWR@select@html@nohspace}{%
11011     \RenewDocumentCommand{\hspace}{s m}{}%
11012 }
```

`\LWR@select@print@hspace`

```
11013 \newcommand*{\LWR@select@print@hspace}{%
11014     \renewrobustcmd\hspace{\@ifstar\@hspacer\@hspace}%
11015 }
```

`\hspace` * {⟨*length*⟩}

Handles special minipage & horizontal space interactions.

```
11016 \LWR@select@html@hspace
```

`\LWR@vspace` * {⟨*length*⟩} Nullified vspace.

```
11017 \NewDocumentCommand{\LWR@HTML@vspace}{s m}{}
11018
11019 \LWR@formatted{vspace}
```

`\linebreak` [⟨*num*⟩] Inserts an HTML `br` tag.

```
11020 \renewcommand*{\linebreak}[1][]{\newline}
```

`\nolinebreak` [⟨*num*⟩]

```
11021 \renewcommand*{\nolinebreak}[1][]{}
```

`\pagebreak` `[`⟨*num*⟩`]` Starts a new paragraph.

```
\renewcommand*{\pagebreak}[1][]{
}
```

`\nopagebreak` `[`⟨*num*⟩`]`

```
\renewcommand*{\nopagebreak}[1][]{}
```

`\enlargethispage` `*` `{`⟨*len*⟩`}`

```
\RenewDocumentCommand{\enlargethispage}{s m}{}
```

`\clearpage`
`\cleardoublepage`

```
\renewcommand*{\clearpage}{}
\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.

```
\newcommand*{\LWR@HTML@rule}[3][]{%
```

The width is copied into a temporary LaTeX length, from which comparisons and conversions may be made:

```
\setlength{\LWR@tempwidth}{#2}%
```

If it's zero-width then skip the entire rule:

```
\ifthenelse{\lengthtest{\LWR@tempwidth=0pt}}%
{}% zero- width
{% non-zero width
```

If it's non-zero width, set a minimal thickness so that it more reliably shows in the browser:

```
    \ifthenelse{%
        \lengthtest{\LWR@tempwidth>0pt}\AND%
        \lengthtest{\LWR@tempwidth<1pt}%
    }%
        {\setlength{\LWR@tempwidth}{1pt}}%
        {}%
```

Likewise with height:

```
    \setlength{\LWR@tempheight}{#3}%
    \ifthenelse{%
        \lengthtest{\LWR@tempheight>0pt}\AND%
        \lengthtest{\LWR@tempheight<1pt}%
    }%
        {\setlength{\LWR@tempheight}{1pt}}%
        {}%
```

If had a minipage this paragraph, try to inline the rule without generating paragraph tags:

```
    \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.

```
    \LWR@findcurrenttextcolor%
    \LWR@htmltagc{%
    span\LWR@indentHTML%
    style="%
```

The background color is used to draw the filled rule. The color may be changed by `\textcolor`.

```
    \ifbool{FormatWP}{}{background:\LWR@currenttextcolor ; }%
```

The width and height are printed, converted to PT:

```
    width:\LWR@printlength{\LWR@tempwidth} ; %
    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.

```
    \ifblank{#1}%
    {}%
    {%
        \setlength{\LWR@tempraise}{0pt-#1}%
        \setlength{\LWR@tempraise}{\LWR@tempraise*2}%
        \LWR@indentHTML%
        -ms-transform: translate(0pt,\LWR@printlength{\LWR@tempraise}); %
        \LWR@indentHTML%
        -webkit-transform: translate(0pt,\LWR@printlength{\LWR@tempraise}); %
        \LWR@indentHTML%
        transform: translate(0pt,\LWR@printlength{\LWR@tempraise}); %
        \LWR@indentHTML%
    }%
```

Display inline-block to place the span inline with the text:

```
    display:inline-block;"\LWR@orignewline%
    }%
```

If formatting for a word processor, approximate with a number of underscores, in case a span of a given width is not supported:

```
        \ifbool{FormatWP}{%
            \setlength{\LWR@templengthone}{\LWR@tempwidth}%
            \whiledo{\lengthtest{\LWR@templengthone>1em}}{%
                \_{}%
                \addtolength{\LWR@templengthone}{-1em}%
            }%
        }{}%
```

Close the span:

```
    \LWR@htmltagc{/span}%
```

If had a minipage this paragraph, try to inline the white space without generating paragraph tags:

```
    \LWR@minipagestartpars%
}% non-zero width
}

\LWR@formatted{rule}

\end{warpHTML}
```

91 \phantomsection

for HTML output:

```
\begin{warpHTML}
```

\phantomsection

Emulate the hyperref \phantomsection command, often used to insert the bibliography into the table of contents. Ignores \ForceHTMLTOC.

```
\DeclareDocumentCommand{\phantomsection}{}{%
\begingroup%
\boolfalse{LWR@forcinghtmltoc}%
\section*{}%
\endgroup%
}

\end{warpHTML}
```

92 \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 `\let` in `\LWR@LwarpStart`.

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.

92.1 HTML logos

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

`\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.

```
\LetLtxMacro\LWR@origTeX\TeX

\newrobustcmd*{\LWR@TeX}
{%
    \InlineClass{latexlogofont}%
    {%
        \InlineClass{latexlogo}%
        {%
            T%
            \InlineClass{latexlogosub}{e}%
            X%
        }%
    }%
}
```

`\LaTeX` LATEX, LATEX 2_ε
`\LaTeXe`

```
\LetLtxMacro\LWR@origLaTeX\LaTeX

\newrobustcmd*{\LWR@LaTeX}
{%
    \InlineClass{latexlogofont}%
    {%
        \InlineClass{latexlogo}%
        {%
            L%
            \InlineClass{latexlogosup}{a}%
            T%
            \InlineClass{latexlogosub}{e}%
            X%
        }%
    }%
```

```
}

\LetLtxMacro\LWR@origLaTeXe\LaTeXe

\renewrobustcmd*{\LaTeXe}
{%
    \LaTeX%
    \InlineClass{latexlogofont}{%
        \InlineClass{latexlogotwoe}{%
            2%
            \InlineClass{latexlogotwoesub}{\HTMLunicode{03B5}}%
        }%
    }%
}
```

\LuaTeX \LuaLaTeX LuaTEX, LuaLATEX

```
\newrobustcmd*{\LWR@LuaTeX}{\InlineClass{latexlogofont}{Lua}\TeX}
\newrobustcmd*{\LWR@LuaLaTeX}{\InlineClass{latexlogofont}{Lua}\LaTeX}
```

\XeTeX \XeLaTeX XETEX, XELATEX

xetexlogo is a css class which aligns the backwards E in XETEX and spaces TEX appropriately.

xelatexlogo is a css class which aligns the backwards E in XELATEX and spaces LATEX appropriately.

```
\newrobustcmd*{\Xe}
    {%
        X%
        \InlineClass{xelatexlogosub}{\HTMLunicode{18e}}%
    }

\newrobustcmd*{\LWR@XeTeX}{\InlineClass{xelatexlogo}{\Xe}\TeX}

\newrobustcmd*{\LWR@XeLaTeX}{\InlineClass{xelatexlogo}{\Xe}\LaTeX}
```

\ConTeXt ConTEXt

```
\newrobustcmd*{\LWR@ConTeXt}
{\InlineClass{latexlogofont}{Con}\TeX{}%
\InlineClass{latexlogofont}{t}}
```

\BibTeX \MakeIndex BIBTEX, *MakeIndex*

```
\providerobustcmd*{\BibTeX}
{\InlineClass{latexlogofont}{B\textsc{ib}}\TeX}

\newrobustcmd*{\MakeIndex}
{\InlineClass{latexlogofont}{\textit{MakeIndex}}}
```

\AmS $\mathcal{AMS}$

amslogo is a css class used for the $\mathcal{AMS}$ logo.

```
\AtBeginDocument{%
\DeclareDocumentCommand{\AmS}{}
{%
    \InlineClass{amslogo}{%
        \textit{%
            A%
            \InlineClass{latexlogosub}{M}%
            S%
        }%
    }%
}%
}
```

\MiKTeX MiKTEX

```
\newrobustcmd*{\MiKTeX}{\InlineClass{latexlogofont}{MiK}\TeX}
```

\LyX LyX

lyxlogo is a css class used for the LyX logo.

```
\newrobustcmd*{\LyX}{\InlineClass{lyxlogo}{LyX}}
```

```
\end{warpHTML}
```

92.2 Print logos

for PRINT output:

```
\begin{warpprint}

\newrobustcmd*{\XeTeXrevE}
    {\hspace{-.1667em}\raisebox{-.5ex}{E}\hspace{-.125em}}

\AtBeginDocument{
\@ifpackageloaded{graphics}{
    \renewrobustcmd*{\XeTeXrevE}
        {\hspace{-.1667em}\raisebox{-.5ex}{\reflectbox{E}}\hspace{-.125em}}
}{
    \PackageWarning{lwarp}{Load graphicx or graphics
        for improved XeTeX logo}
}
}

\providerobustcmd*{\XeTeX}{\mbox{X\XeTeXrevE\TeX}}
\providerobustcmd*{\XeLaTeX}{\mbox{X\XeTeXrevE\LaTeX}}
\providerobustcmd*{\AmS}{%
    \leavevmode\hbox{$\mathcal A\kern-.2em\lower.376ex%
    \hbox{$\mathcal M$}\kern-.2em\mathcal S$}%
}
```

```
\newrobustcmd*{\LyX}{\textsf{LyX}}
\providerobustcmd*{\LuaTeX}{\mbox{Lua\TeX}}
\providerobustcmd*{\LuaLaTeX}{\mbox{Lua\LaTeX}}
\providerobustcmd*{\BibTeX}{\mbox{B\textsc{ib}\TeX}}
\providerobustcmd*{\MakeIndex}{\mbox{\textit{MakeIndex}}}
\providerobustcmd*{\ConTeXt}{\mbox{Con\TeX{}t}}
\providerobustcmd*{\MiKTeX}{\mbox{MiK\TeX}}
\end{warpprint}
```

93 \AtBeginDocument, \AtEndDocument

for HTML output:

```
\begin{warpHTML}
```

\LWR@LwarpStart
\LWR@LwarpEnd

Automatically sets up the HTML-related actions for the start and end of the document.

```
\AfterEndPreamble{\LWR@LwarpStart}
\AtEndDocument{\LWR@LwarpEnd}

\end{warpHTML}
```

94 Loading KOMA-SCRIPT class patches

Load patches to koma-script.

for HTML output:

```
\begin{warpHTML}

\@ifclassloaded{scrbook}{\RequirePackage{lwarp-patch-komascript}}{}
\@ifclassloaded{scrartcl}{\RequirePackage{lwarp-patch-komascript}}{}
\@ifclassloaded{scrreprt}{\RequirePackage{lwarp-patch-komascript}}{}

\end{warpHTML}
```

95 Loading MEMOIR class patches

Load patches to memoir.

for HTML output:

```
\begin{warpHTML}

\@ifclassloaded{memoir}{\RequirePackage{lwarp-patch-memoir}}{}

\end{warpHTML}
```

96 ut* class patches

Load patches to uj* and ut* classes, as well as ltj* classes.

for HTML output:

```
\begin{warpHTML}
\newcommand*{\LWR@patchujtclasses}{
```

uj/t does not use \partname

```
    \def\@partnameformat{}

    \def\@partcntformat##1{%
        \prepartname%
        \csname the##1\endcsname%
        \postpartname%
        \quad%
    }
    \@ifundefined{chapter}{}{
        \def\@chapcntformat##1{%
            \prechaptername%
            \csname the##1\endcsname%
            \postchaptername%
            \quad%
        }
    }
```

Use decimal points instead of centered dots:

```
    \renewcommand{\thepart}{\@Roman\c@part}
    \@ifundefined{chapter}{
        \renewcommand{\thesection}{\@arabic\c@section}
    }{
        \renewcommand{\thechapter}{\@arabic\c@chapter}
        \renewcommand{\thesection}{\thechapter.\@arabic\c@section}
    }
    \renewcommand{\thesubsection}{\thesection.\@arabic\c@subsection}
    \renewcommand{\thesubsubsection}{%
    \thesubsection.\@arabic\c@subsubsection}
    \renewcommand{\theparagraph}{%
    \thesubsubsection.\@arabic\c@paragraph}
    \renewcommand{\thesubparagraph}{%
    \theparagraph.\@arabic\c@subparagraph}
    \@ifundefined{chapter}{
        \renewcommand{\thefigure}{\@arabic\c@figure}
        \renewcommand{\thetable}{\@arabic\c@table}
    }{
        \renewcommand{\thefigure}{%
        \ifnum\c@chapter>\z@\thechapter.\fi\@arabic\c@figure}
        \renewcommand{\thetable}{%
        \ifnum\c@chapter>\z@\thechapter.\fi\@arabic\c@table}
    }
}
```

```
\@ifclassloaded{ujarticle}{\LWR@patchujtclasses}{}
\@ifclassloaded{ujbook}{\LWR@patchujtclasses}{}
\@ifclassloaded{ujreport}{\LWR@patchujtclasses}{}
\@ifclassloaded{utarticle}{\LWR@patchujtclasses}{}
\@ifclassloaded{utbook}{\LWR@patchujtclasses}{}
\@ifclassloaded{utreport}{\LWR@patchujtclasses}{}
\@ifclassloaded{ltjarticle}{\LWR@patchujtclasses}{}
\@ifclassloaded{ltjbook}{\LWR@patchujtclasses}{}
\@ifclassloaded{ltjreport}{\LWR@patchujtclasses}{}
\@ifclassloaded{ltjsarticle}{\LWR@patchujtclasses}{}
\@ifclassloaded{ltjsbook}{\LWR@patchujtclasses}{}
\@ifclassloaded{ltjsreport}{\LWR@patchujtclasses}{}
\@ifclassloaded{ltjskiyou}{\LWR@patchujtclasses}{}
\@ifclassloaded{ltjspf}{\LWR@patchujtclasses}{}
\@ifclassloaded{ltjtarticle}{\LWR@patchujtclasses}{}
\@ifclassloaded{ltjtbook}{\LWR@patchujtclasses}{}
\@ifclassloaded{ltjtreport}{\LWR@patchujtclasses}{}

\end{warpHTML}
```

97 CTEX patches

Patches for ctex and related classes, which are loaded before lwarp.

All CTEX classes and the ctex package seem to load ctexpatch, so its presence is used to decide whether to have lwarp patch CTEX.

for HTML output:

```
\begin{warpHTML}
```

`\AtBeginDocument` in case the user set `FileSectionNames` in the preamble.

```
\AtBeginDocument{
    \@ifpackageloaded{ctexpatch}{%
        \def\@partcntformat#1{%
            \LWR@isolate{\CTEX@partname}~%
            \CTEX@part@aftername%
        }%

        \def\@partnameformat{}

        \def\@chapcntformat#1{%
            \LWR@isolate{\CTEX@chaptername}~%
            \CTEX@chapter@aftername%
        }%
    }{}
}

\end{warpHTML}
```

98 kotexutf patches

Patch for kotexutf, which is loaded before lwarp.

kotexutf's `\@setref` was conflicting with lwarp's cross references.

for HTML output:

```
\begin{warpHTML}
```

If kotexutf's version of `\@setref` is detected, it is reverted to the original.

```
\AtBeginDocument{
\@ifpackageloaded{kotexutf}{%
    \def\LWR@kotexutf@setref#1#2#3{%
        \@setref@dhucs@orig{#1}{#2}{#3}%
      \ifx#1\relax\else
        \bgroup
        \dhucs@make@cjkchar@null
        \edef\@temp{\expandafter#2#1}\global\josatoks\expandafter{\@temp}%
        \egroup
      \fi%
    }%

    \ifdefequal{\@setref}{\LWR@kotexutf@setref}{
        \let\@setref\@setref@dhucs@orig
    }{}
}{}
}

\end{warpHTML}
```

File 2 **lwarp-2in1.sty**

§ 99 Package 2in1

Pkg 2in1 2in1 is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{2in1}
```

File 3 **lwarp-2up.sty**

§ 100 Package 2up

Pkg 2up 2up is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{2up}[2010/05/15]

\def\source#1#2#3{}
\def\target#1#2#3{}
\def\targetlayout#1{}
\newdimen\pageseplength
\newdimen\pagesepwidth
\newdimen\pagesepoffset
\def\twoupemptypage{}
\def\twoupclearpage{}
\def\twoupeject{}
\def\twouparticle{}
\def\twoupplain{}
\def\twouplegaltarget{}
\def\twouplandscape{}
\def\TwoupWrites{}
```

File 4 **lwarp-a4.sty**

§ 101 Package a4

Pkg a4 a4 is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{a4}[2004/04/15]

\newcommand*{\WideMargins}{}
```

File 5 **lwarp-a4wide.sty**

§ 102 Package a4wide

Pkg a4wide a4wide is ignored.

for HTML output:
```
\LWR@ProvidesPackageDrop{a4wide}[1994/08/30]
```

File 6 **lwarp-a5comb.sty**

§ 103 Package a5comb

Pkg a5comb a5comb is ignored.

for HTML output:
```
\LWR@ProvidesPackageDrop{a5comb}
```

File 7 **lwarp-abstract.sty**

§ 104 Package abstract

(*Emulates or patches code by* Peter Wilson.)

Pkg abstract 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:

```
\ProvideDocumentEnvironment{abstract}{}{}{}
```

Accept all options for lwarp-abstract:

```
\LWR@ProvidesPackagePass{abstract}[2009/06/08]

\AtBeginDocument{
\BeforeBeginEnvironment{abstract}{
\LWR@forcenewpage
\BlockClass{abstract}
}
```

```
\AfterEndEnvironment{abstract}{\endBlockClass}
}

\renewcommand{\@bsrunintitle}{%
\hspace*{\abstitleskip}%
{\abstractnamefont%
\InlineClass{abstractrunintitle}{\abstractname}%
\@bslabeldelim}%
}

\@ifclassloaded{memoir}
{
  \renewenvironment{abstract}{%
%     \titlepage
    \null\vfil
    \@beginparpenalty\@lowpenalty
    \if@bsrunin
    \else
      \if@bsstyle
        \abstitlestyle{\BlockClassSingle{abstracttitle}{\abstractname}}
      \else
        \ifnumber@bs
          \num@bs
        \else
          \begin{\absnamepos}%
  \abstractnamefont \BlockClassSingle{abstracttitle}{\abstractname}
            \@endparpenalty\@M
          \end\absnamepos%
%%        \vspace{\abstitleskip}%
        \fi
      \fi
      \vspace{\abstitleskip}%
    \fi
    \put@bsintoc%
    \begin{@bstr@ctlist}\if@bsrunin\@bsrunintitle\fi\abstracttextfont}%
    {\par\end{@bstr@ctlist}\vfil\null%\endtitlepage
    }
}{% not memoir
\if@titlepage
  \renewenvironment{abstract}{%
%     \titlepage
    \null\vfil
    \@beginparpenalty\@lowpenalty
    \if@bsrunin
    \else
      \if@bsstyle
        \abstitlestyle{\BlockClassSingle{abstracttitle}{\abstractname}}
      \else
        \ifnumber@bs
          \num@bs
        \else
          \begin{\absnamepos}%
  \abstractnamefont \BlockClassSingle{abstracttitle}{\abstractname}
            \@endparpenalty\@M
          \end\absnamepos%
```

```
%%        \vspace{\abstitleskip}%
        \fi
      \fi
      \vspace{\abstitleskip}%
    \fi
    \put@bsintoc%
    \begin{@bstr@ctlist}\if@bsrunin\@bsrunintitle\fi\abstracttextfont}%
    {\par\end{@bstr@ctlist}\vfil\null%\endtitlepage
    }
\else
  \renewenvironment{abstract}{%
    \if@bsrunin
    \else
      \if@bsstyle
        \abstitlestyle{\BlockClassSingle{abstracttitle}{\abstractname}}
      \else
        \ifnumber@bs
          \num@bs
        \else
\begin{\absnamepos}%
\abstractnamefont\BlockClassSingle{abstracttitle}{\abstractname}%
\end\absnamepos%
%%        \vspace{\abstitleskip}%
        \fi
      \fi
      \vspace{\abstitleskip}%
    \fi
    \put@bsintoc%
    \begin{@bstr@ctlist}\if@bsrunin\@bsrunintitle\fi\abstracttextfont}%
    {\par\end{@bstr@ctlist}}
\fi
}% not memoir
```

File 8 **lwarp-academicons.sty**

§ 105 Package

academicons

(*Emulates or patches code by* Diogo A. B. Fernandes.)

Pkg academicons 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:

```
\LWR@ProvidesPackagePass{academicons}[2018/06/27]

\LetLtxMacro\LWR@orig@symbol\symbol

\let\LWR@academicons@orig@AI\AI

\newcommand*{\LWR@academicons@symbol}[1]{%
    \begin{lateximage}*[(academicon)][academicons#1]%
    \begingroup%
```

```
    \LWR@academicons@orig@AI%
    \LWR@orig@symbol{#1}%
    \endgroup%
    \end{lateximage}%
}

\renewcommand*{\AI}{%
    \LetLtxMacro\symbol\LWR@academicons@symbol%
}

\renewcommand*{\aiicon}[1]
{%
    \begin{lateximage}*[(#1 icon)][academicons#1]%
    \AI\csname aiicon@#1\endcsname%
    \end{lateximage}%
}
```

File 9 **lwarp-accsupp.sty**

§ 106 Package accsupp

Pkg accsupp accsupp is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{accsupp}[2018/03/28]

\newcommand*{\BeginAccSupp}[1]{}
\newcommand*{\EndAccSupp}[1]{}
```

File 10 **lwarp-acro.sty**

§ 107 Package acro

(*Emulates or patches code by* Clemens Niederberger.)

Pkg acro acro is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{acro}[2017/01/30]
```

\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:

```
\ExplSyntaxOn
\NewDocumentCommand \LWR@DeclareAcronym {mm}
{
    \acro_declare_acronym:nn {#1} {#2}
    \catcode`\$=3% lwarp
}
\ExplSyntaxOff

```

```
\RenewDocumentCommand{\DeclareAcronym}{}{
    \catcode`\$=\active% lwarp
    \LWR@DeclareAcronym
}
```

Modified to activate the current font:

```
\ExplSyntaxOn
\cs_gset_protected:Npn \acro_write_short:nn #1#2
  {
    \mode_if_horizontal:F { \leavevmode }
    \group_begin:
      \bool_if:NTF \l__acro_custom_format_bool
        { \l__acro_custom_format_tl }
        { \l__acro_short_format_tl }
      {\LWR@textcurrentfont{#2}}% lwarp
    \group_end:
  }

\cs_gset_protected:Npn \acro_write_alt:nn #1#2
  {
    \mode_if_horizontal:F { \leavevmode }
    \group_begin:
      \bool_if:NTF \l__acro_custom_format_bool
        { \l__acro_custom_format_tl }
        { \l__acro_alt_format_tl }
      {\LWR@textcurrentfont{#2}}% lwarp
    \group_end:
  }

\cs_gset_protected:Npn \acro_write_long:nn #1#2
  {
    \mode_if_horizontal:F { \leavevmode }
    \group_begin:
      \bool_if:NTF \l__acro_custom_long_format_bool
        { \l__acro_custom_long_format_tl }
        { \use:n }
      {
        \use:x
          {
            \exp_not:n {#1}
            {
              \bool_if:NTF \l__acro_first_upper_bool
                { \exp_not:N \__acro_first_upper_case:n { \exp_not:n {
                     \LWR@textcurrentfont{#2}% lwarp
                } } }
                { \exp_not:n {\LWR@textcurrentfont{#2}} }% lwarp
            }
          }
      }
    \group_end:
  }
\ExplSyntaxOff
```

File 11 **lwarp-acronym.sty**

§ 108 Package

acronym

(*Emulates or patches code by* Tobias Oetiker.)

Pkg acronym 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:

```
\LWR@ProvidesPackagePass{acronym}[2015/03/21]
```

Uses `\textit` instead of `\itshape`:

```
\renewcommand{\acfia}[1]{%
  {\textit{\AC@acl{#1}}} (\ifAC@starred\acs*{#1}\else\acs{#1}\fi)}
```

Removes the mbox to allow math inside:

```
\renewcommand*\AC@acs[1]{%
%     \mbox{
 \expandafter\AC@get\csname fn@#1\endcsname\@firstoftwo{#1}}
% }
```

Fix for acronym labels in the captions of floats.

```
\renewcommand{\@starttoc}[1]{
\LWR@htmlelementclass{nav}{#1}
\LetLtxMacro\@verridelabel\@gobble
\LWR@orig@starttoc{#1}
\LWR@htmlelementclassend{nav}{#1}
}
```

Modified for cleveref and lwarp:

```
\renewcommand*\AC@und@newl@bel[3]{%
    \@ifundefined{#1@#3}%
    {%
        \global\expandafter\let\csname#2@#3\endcsname\@nnil
        \global\expandafter\let\csname#2@#3@lwarp\endcsname\@nnil% lwarp
        \global\expandafter\let\csname#2@#3@cref\endcsname\@nnil% lwarp
    }%
    {%
        \global\expandafter\let\csname#1@#3\endcsname\relax
        \global\expandafter\let\csname#1@#3@lwarp\endcsname\relax% lwarp
        \global\expandafter\let\csname#1@#3@cref\endcsname\relax% lwarp
    }%
}%
```

File 12 **lwarp-adjmulticol.sty**

§ 109 Package

adjmulticol

(*Emulates or patches code by* Boris Veytsman.)

Pkg adjmulticol 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:

```
\LWR@ProvidesPackageDrop{adjmulticol}[2012/01/20]
\RequirePackage{multicol}
```

adjmulticols * {⟨*numcols*⟩} {⟨*left margi*⟩} {⟨*right margin*⟩}

```
\NewDocumentEnvironment{adjmulticols}{s m m m}
{%
```

Compute the margins, and limit to positive only:

```
\setlength{\LWR@templengthone}{#3}%
\ifdimcomp{\LWR@templengthone}{<}{0pt}{\setlength{\LWR@templengthone}{0pt}}{}%
\setlength{\LWR@templengthtwo}{#4}
\ifdimcomp{\LWR@templengthtwo}{<}{0pt}{\setlength{\LWR@templengthtwo}{0pt}}{}%
```

If one column is specified, use a of class singlecolumn, else use multicols:

```
\newcommand*{\LWR@mcolstype}{multicols}%
\ifnumcomp{#2}{=}{1}{\renewcommand*{\LWR@mcolstype}{singlecolumn}}{}%
```

Help avoid page overflow:

```
\LWR@forcenewpage%
```

Create the with the given margin and class:

```
\BlockClass[%
\LWR@print@mbox{margin-left:\LWR@printlength{\LWR@templengthone}} ; %
\LWR@print@mbox{margin-right:\LWR@printlength{\LWR@templengthtwo}}%
]{\LWR@mcolstype}%
}
{\endBlockClass}
```

File 13 **lwarp-addlines.sty**

§ 110 Package addlines

(*Emulates or patches code by* Will Robertson.)

Pkg addlines addlines is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{addlines}[2018/12/05]

\newcommand\addlines{\@ifstar\addlines@a\addlines@a}
\newcommand\addlines@a[1][1]{}
\let\addline\addlines
\newcommand\removelines{\@ifstar\removelines@a\removelines@a}
\newcommand\removelines@a[1][1]{}
\let\removeline\removelines
\newcommand\squeezepage[1][0]{}
```

File 14 **lwarp-afterpage.sty**

§ 111 Package afterpage

(*Emulates or patches code by* David Carlisle.)

Pkg afterpage Emulated.

for HTML output: Discard all options for lwarp-afterpage:

```
\LWR@ProvidesPackageDrop{afterpage}[2014/10/28]

\newcommand{\afterpage}[1]{#1}
```

File 15 **lwarp-algorithm2e.sty**

§ 112 Package algorithm2e

(*Emulates or patches code by* Christophe Fiorio.)

Pkg algorithm2e 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:

```
\LWR@ProvidesPackagePass{algorithm2e}[2017/07/18]
```

For the list-of entries:

```
\renewcommand{\l@algocf}[2]{\hypertocfloat{1}{algocf}{loa}{#1}{#2}}
```

Select the lwarp float style according to the algorithm2e style:

```
\newcommand*{\LWR@floatstyle@algocf}{ruled}

\ifdefstring{\algocf@style}{boxed}{%
\renewcommand*{\LWR@floatstyle@algocf}{boxed}
}{}

\ifdefstring{\algocf@style}{boxruled}{%
\renewcommand*{\LWR@floatstyle@algocf}{boxruled}
}{}

\ifdefstring{\algocf@style}{plain}{%
\renewcommand*{\LWR@floatstyle@algocf}{plain}
}{}
```

Paragraph handling to allow line numbers under certain conditions:

```
\newbool{LWR@algocf@dopars}
\booltrue{LWR@algocf@dopars}

\renewcommand{\algocf@everypar}{%
\ifbool{LWR@algocf@dopars}{%
    \ifbool{LWR@doingstartpars}{%
        \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
            {}%
            {%
                \algocf@everyparnl\algocf@everyparhanging%
            }%
    }{}%
}{}%
}
```

lwarp caption handling:

```
\renewcommand{\algocf@makecaption}[2]{%
\LWR@HTML@caption@begin{algocf}%
\LWR@isolate{\algocf@captiontext{#1}{#2}}%
\LWR@HTML@caption@end%
}
```

Print any caption where it is declared:

```
\renewcommand{\algocf@makecaption@plain}[2]{%
    \LWR@HTML@caption@begin{algocf}%
    \LWR@isolate{\algocf@captiontext{#1}{#2}}%
    \LWR@HTML@caption@end%
}

\renewcommand{\algocf@makecaption@boxed}[2]{%
    \LWR@HTML@caption@begin{algocf}%
```

```
    \LWR@isolate{\algocf@captiontext{#1}{#2}}%
    \LWR@HTML@caption@end%
}

\renewcommand{\algocf@makecaption@ruled}[2]{%
    \LWR@HTML@caption@begin{algocf}%
    \LWR@isolate{\algocf@captiontext{#1}{#2}}%
    \LWR@HTML@caption@end%
}
```

Turn off line numbering while making the caption:

```
\long\def\algocf@latexcaption#1[#2]#3{% original definition of caption
\boolfalse{LWR@algocf@dopars}%  lwarp
  \par%
  \addcontentsline{\csname ext@#1\endcsname}{#1}%
 {\protect\numberline{\csname the#1\endcsname}{\ignorespaces \LWR@isolate{#2}}}%
  \begingroup%
  \@parboxrestore%
  \if@minipage%
    \@setminipage%
  \fi%
  \normalsize%
  \@makecaption{\csname fnum@#1\endcsname}{\ignorespaces #3}\par%
  \endgroup%
\booltrue{LWR@algocf@dopars}%   lwarp
}
```

Line numbers are printed in a of class `alg2elinenumber`:

```
\renewcommand{\algocf@printnl}[1]{%
    \InlineClass{alg2elinenumber}{\NlSty{#1}}~%
}%
```

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.

```
\preto\@algocf@init{%
  \edef\LWR@floatstyle@figure{\LWR@floatstyle@algocf}%
}
```

For lwarp, the algorithm is not assembled inside a box, since `latexi`mages would not work, so the captions are printed where declared.

```
\renewcommand{\@algocf@start}{%
    \let\@mathsemicolon=\;\def\;{\ifmmode\@mathsemicolon\else\@endalgoln\fi}%
%     \raggedright%
    \AlFnt{}%
    \booltrue{LWR@algocf@dopars}% lwarp
}

\renewcommand{\@algocf@finish}{%
    \boolfalse{LWR@algocf@dopars}% lwarp
    \lineskip\normallineskip\setlength{\skiptotal}{\@defaultskiptotal}%
```

```
    \let\;=\@mathsemicolon%
    \let\]=\@emathdisplay%
}
```

Use an HTML break:

```
\renewcommand{\BlankLine}{%
\LWR@stoppars%
\LWR@htmltagc{br /}%
\LWR@startpars%
}
```

Simplified for HTML. The paragraph handling must be preserved.

```
\renewcommand{\SetKwInOut}[2]{%
  \algocf@newcommand{#1}[1]{%
    \ifthenelse{\boolean{algocf@hanginginout}}%
        {\relax}%
        {\algocf@seteveryparhanging{\relax}}%
    \ifthenelse{\boolean{algocf@inoutnumbered}}%
        {\relax}%
        {\algocf@seteveryparnl{\relax}}%
    {%
            \KwSty{#2\algocf@typo:}%
        ~##1\par%
    }%
    \algocf@linesnumbered% reset the numbering of the lines
    \ifthenelse{\boolean{algocf@hanginginout}}%
        {\relax}%
        {\algocf@reseteveryparhanging}%
  }%
}%

\renewcommand{\ResetInOut}[1]{}%
```

Each of the following creates a of a given class, and turns off line numbering while creating the tags:

```
\renewcommand{\algocf@Vline}[1]{%
\boolfalse{LWR@algocf@dopars}%
\begin{BlockClass}{alg2evline}
\booltrue{LWR@algocf@dopars}%
#1
\boolfalse{LWR@algocf@dopars}%
\end{BlockClass}
\booltrue{LWR@algocf@dopars}%
}

\renewcommand{\algocf@Vsline}[1]{%
\boolfalse{LWR@algocf@dopars}%
\begin{BlockClass}{alg2evsline}
\booltrue{LWR@algocf@dopars}%
#1
\boolfalse{LWR@algocf@dopars}%
\end{BlockClass}
```

```
\booltrue{LWR@algocf@dopars}%
}

\renewcommand{\algocf@Noline}[1]{%
\boolfalse{LWR@algocf@dopars}%
\begin{BlockClass}{alg2enoline}
\booltrue{LWR@algocf@dopars}%
#1
\boolfalse{LWR@algocf@dopars}%
\end{BlockClass}
\booltrue{LWR@algocf@dopars}%
}
```

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.

```
\LetLtxMacro\algocf@Here\algocf
\LetLtxMacro\endalgocf@Here\endalgocf
```

File 16 **lwarp-algorithmicx.sty**

§ 113 Package

algorithmicx

(*Emulates or patches code by* Szász János.)

Pkg algorithmicx algorithmicx is supported with minor adjustments.

for HTML output:

```
\LWR@ProvidesPackagePass{algorithmicx}[2005/04/27]
```

Inside the algorithmic environment, level indenting is converted to a of the required length, and comments are placed inside a which is floated right.

⚠ package conflicts If using \newfloat, trivfloat, and/or algorithmicx together, see section 450.1.

for HTML output:

```
\begin{warpHTML}

\AtBeginEnvironment{algorithmic}{%
%
\let\origALG@doentity\ALG@doentity%
%
\renewcommand*{\ALG@doentity}{%
\origALG@doentity%
\LWR@htmltagc{%
span style="width:\LWR@printlength{\ALG@thistlm}; display:inline-block;"%
}%
\ifbool{FormatWP}{%
\setlength{\LWR@templengthone}{\the\ALG@thistlm}%
\whiledo{\lengthtest{\LWR@templengthone>1em}}{%
\quad%
\addtolength{\LWR@templengthone}{-1em}%
}%
```

```
}{}%
\LWR@htmltagc{/span}%
}%

\let\LWR@origComment\Comment%

\renewcommand{\Comment}[1]{%
    \InlineClass{floatright}{\LWR@origComment{#1}}%
}%
}

\renewcommand\algorithmiccomment[1]{%
\hfill\HTMLunicode{25B7} #1% white right triangle
}%

\end{warpHTML}
```

File 17 **lwarp-alltt.sty**

§ 114 Package alltt

(*Emulates or patches code by* Johannes Braams.)

Pkg alltt alltt is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{alltt}[1997/06/16]

\AfterEndPreamble{
\LWR@traceinfo{Patching alltt.}
\AtBeginEnvironment{alltt}{%
\LWR@forcenewpage
\LWR@atbeginverbatim{3}{alltt}%
}
\AfterEndEnvironment{alltt}{%
    \LWR@afterendverbatim{2}%
}
}
```

File 18 **lwarp-amsmath.sty**

§ 115 Package amsmath

(*Emulates or patches code by* American Mathematical Society, LaTeX3 Project.)

Pkg amsmath amsmath is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{amsmath}[2017/09/02]
```

Patches to allow `\eqref` inside a caption:

```
\def\maketag@@@#1{\text{#1}}
\def\tagform@#1{\maketag@@@{(\ignorespaces#1\unskip)}}
```

Patches for $\mathcal{AMS}$ math \tag macro to remember the first tag:

```
\ifbool{mathjax}{}{% not mathjax

\LetLtxMacro\LWR@origmake@df@tag@@\make@df@tag@@
\LetLtxMacro\LWR@origmake@df@tag@@@\make@df@tag@@@

\renewcommand*{\make@df@tag@@}[1]{%
\LWR@remembertag{#1}%
\LWR@origmake@df@tag@@{#1}%
}

\renewcommand*{\make@df@tag@@@}[1]{%
\LWR@remembertag{#1}%
\LWR@origmake@df@tag@@@{#1}%
}

}% not mathjax
```

The following $\mathcal{AMS}$ environments are more easily patched in-place:

Env multline

```
\BeforeBeginEnvironment{multline}{

\ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
{
    \LWR@syncmathjax
    \booltrue{LWR@amsmultline}
    \LWR@beginhideamsmath
}
{
    \begin{BlockClass}{displaymathnumbered}
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[\LWR@amsmathbodynumbered{multline}]
    \LWR@applyxfakebold%
}
}

\AfterEndEnvironment{multline}{

\ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
{
    \LWR@endhideamsmath
    \boolfalse{LWR@amsmultline}
    \LWR@addmathjax{multline}{\the\@envbody}
}
{\end{lateximage}\end{BlockClass}}

}
```

Env multline*

```
\BeforeBeginEnvironment{multline*}{

\ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
{
    \LWR@syncmathjax
    \booltrue{LWR@amsmultline}
    \LWR@beginhideamsmath
}
{
    \begin{BlockClass}{displaymath}
    \LWR@newautoidanchor
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[\LWR@amsmathbody{multline*}]
    \LWR@applyxfakebold%
}
}

\AfterEndEnvironment{multline*}{

\ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
{
    \LWR@endhideamsmath
    \boolfalse{LWR@amsmultline}
    \LWR@addmathjax{multline*}{\the\@envbody}
}
{\end{lateximage}\end{BlockClass}}

}

```

Env gather

```
\BeforeBeginEnvironment{gather}{

\ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
{
    \LWR@syncmathjax
    \boolfalse{LWR@amsmultline}
    \LWR@beginhideamsmath
}
{
    \begin{BlockClass}{displaymathnumbered}
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[\LWR@amsmathbodynumbered{gather}]
    \LWR@applyxfakebold%
}
}

\AfterEndEnvironment{gather}{

\ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
{
```

```
    \LWR@endhideamsmath
    \LWR@addmathjax{gather}{\the\@envbody}
}
{\end{lateximage}\end{BlockClass}}

}
```

Env gather*

```
\BeforeBeginEnvironment{gather*}{

\ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
{
    \LWR@syncmathjax
    \boolfalse{LWR@amsmultline}
    \LWR@beginhideamsmath
}
{
    \begin{BlockClass}{displaymath}
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[\LWR@amsmathbody{gather*}]
    \LWR@applyxfakebold%
}
}

\AfterEndEnvironment{gather*}{

\ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
{
    \LWR@endhideamsmath
    \LWR@addmathjax{gather*}{\the\@envbody}
}
{\end{lateximage}\end{BlockClass}}

}
```

Env align

```
\BeforeBeginEnvironment{align}{

\ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
{
    \LWR@syncmathjax
    \boolfalse{LWR@amsmultline}
    \LWR@beginhideamsmath
}
{
    \begin{BlockClass}{displaymathnumbered}
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[\LWR@amsmathbodynumbered{align}]
    \LWR@applyxfakebold%
}
```

```
}

\AfterEndEnvironment{align}{

\ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
{
    \LWR@endhideamsmath
    \LWR@addmathjax{align}{\the\@envbody}
}
{\end{lateximage}\end{BlockClass}}

}
```

Env `align*`

```
\BeforeBeginEnvironment{align*}{

\ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
{
    \LWR@syncmathjax
    \boolfalse{LWR@amsmultline}
    \LWR@beginhideamsmath
}
{
    \begin{BlockClass}{displaymath}
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[\LWR@amsmathbody{align*}]
    \LWR@applyxfakebold%
}
}

\AfterEndEnvironment{align*}{

\ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
{
    \LWR@endhideamsmath
    \LWR@addmathjax{align*}{\the\@envbody}
}
{\end{lateximage}\end{BlockClass}}

}
```

Env `flalign`

```
\BeforeBeginEnvironment{flalign}{

\ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
{
    \LWR@syncmathjax
    \boolfalse{LWR@amsmultline}
    \LWR@beginhideamsmath
}
{
```

```
    \begin{BlockClass}{displaymathnumbered}
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[\LWR@amsmathbodynumbered{flalign}]
    \LWR@applyxfakebold%
}
}

\AfterEndEnvironment{flalign}{

\ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
{
    \LWR@endhideamsmath
    \LWR@addmathjax{flalign}{\the\@envbody}
}
{\end{lateximage}\end{BlockClass}}

}
```

Env flalign*

```
\BeforeBeginEnvironment{flalign*}{

\ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
{
    \LWR@syncmathjax
    \boolfalse{LWR@amsmultline}
    \LWR@beginhideamsmath
}
{
    \begin{BlockClass}{displaymath}
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[\LWR@amsmathbody{flalign*}]
    \LWR@applyxfakebold%
}
}

\AfterEndEnvironment{flalign*}{

\ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
{
    \LWR@endhideamsmath
    \LWR@addmathjax{flalign*}{\the\@envbody}
}
{\end{lateximage}\end{BlockClass}}

}
```

Env alignat

```
\BeforeBeginEnvironment{alignat}{

\ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
```

```
{
    \LWR@syncmathjax
    \boolfalse{LWR@amsmultline}
    \LWR@beginhideamsmath
}
{
    \begin{BlockClass}{displaymathnumbered}
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[\LWR@amsmathbodynumbered{alignat}]
    \LWR@applyxfakebold%
}
}

\AfterEndEnvironment{alignat}{

\ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
{
    \LWR@endhideamsmath
    \LWR@addmathjax{alignat}{\the\@envbody}
}
{\end{lateximage}\end{BlockClass}}

}
```

Env `alignat*`

```
\BeforeBeginEnvironment{alignat*}{

\ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
{
    \LWR@syncmathjax
    \boolfalse{LWR@amsmultline}
    \LWR@beginhideamsmath
}
{
    \begin{BlockClass}{displaymath}
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[\LWR@amsmathbody{alignat*}]
    \LWR@applyxfakebold%
}
}

\AfterEndEnvironment{alignat*}{

\ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
{
    \LWR@endhideamsmath
    \LWR@addmathjax{alignat*}{\the\@envbody}
}
{\end{lateximage}\end{BlockClass}}

}
```

File 19 **lwarp-amsthm.sty**

§ 116 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.

Pkg amsthm amsthm is patched for use by lwarp.

Table 14: amsthm package — css styling of theorems and proofs

Theorem: of class amsthmbody<theoremstyle>

Theorem Name: of class amsthmname<theoremtyle>

Theorem Number: of class amsthmnumber<theoremstyle>

Theorem Note: of class amsthmnote<theoremstyle>

Proof: of class amsthmproof

Proof Name: of class amsthmproofname

where <theoremstyle> is plain, definition, etc.

for HTML output: amsthm must be loaded before mdframed:

```
\@ifpackageloaded{mdframed}{
    \PackageError{lwarp}
    {Package mdframed must be loaded after package amsthm.}
    {%
     Move \detokenize{\usepackage}{amsthm} before \detokenize{\usepackage}{mdframed}.
       Package amsthm may be loaded by something else, which must also be moved
        before mdframed.%
    }
}
{}

\LWR@ProvidesPackagePass{amsthm}[2017/10/31]
```

Storage for the style being used for new theorems:

```
\newcommand{\LWR@newtheoremstyle}{plain}
```

Patched to remember the style being used for new theorems:

```
\renewcommand{\theoremstyle}[1]{%
  \@ifundefined{th@#1}{%
    \PackageWarning{amsthm}{Unknown theoremstyle `#1'}%
```

```
    \thm@style{plain}%
    \renewcommand{\LWR@newtheoremstyle}{plain}% lwarp
  }{%
    \thm@style{#1}%
    \renewcommand{\LWR@newtheoremstyle}{#1}% lwarp
  }%
}
```

Patched to remember the style for this theorem type:

```
\def\@xnthm#1#2{%
  \csedef{LWR@thmstyle#2}{\LWR@newtheoremstyle}% lwarp
  \let\@tempa\relax
  \@xp\@ifdefinable\csname #2\endcsname{%
    \global\@xp\let\csname end#2\endcsname\@endtheorem
    \ifx *#1% unnumbered, need to get one more mandatory arg
      \edef\@tempa##1{%
        \gdef\@xp\@nx\csname#2\endcsname{%
          \@nx\@thm{\@xp\@nx\csname th@\the\thm@style\endcsname}%
            {}{##1}}}%
    \else % numbered theorem, need to check for optional arg
      \def\@tempa{\@oparg{\@ynthm{#2}}[]}%
    \fi
   \AtBeginEnvironment{#2}{\edef\LWR@thisthmstyle{\@nameuse{LWR@thmstyle#2}}}% lwarp
  }%
  \@tempa
}
```

Patched to enclose with css:

```
\newcommand{\LWR@haveamsthmname}{
\renewcommand{\thmname}[1]{\InlineClass{amsthmname\LWR@thisthmstyle}{##1}}
}

\newcommand{\LWR@haveamsthmnumber}{
\renewcommand{\thmnumber}[1]{\InlineClass{amsthmnumber\LWR@thisthmstyle}{##1}}
}

\newcommand{\LWR@haveamsthmnote}{
\renewcommand{\thmnote}[1]{\InlineClass{amsthmnote\LWR@thisthmstyle}{##1}}
}

\LWR@haveamsthmname
\LWR@haveamsthmnumber
\LWR@haveamsthmnote
```

Patches for css:

```
\def\@begintheorem#1#2[#3]{%
    \item[
%   \deferred@thm@head{
%     \the\thm@headfont \thm@indent
    \@ifempty{#1}{\let\thmname\@gobble}{\LWR@haveamsthmname}% lwarp
    \@ifempty{#2}{\let\thmnumber\@gobble}{\LWR@haveamsthmnumber}% lwarp
    \@ifempty{#3}{\let\thmnote\@gobble}{\LWR@haveamsthmnote}% lwarp
```

```
    \thm@swap\swappedhead\thmhead{#1}{#2}{#3}%
    \the\thm@headpunct~
    \thmheadnl % possibly a newline.
    \hskip\thm@headsep
%   }%
    ]
  \ignorespaces}
```

Patched for css:

```
\def\@thm#1#2#3{%
  \ifhmode\unskip\unskip\par\fi
  \normalfont
  \LWR@forcenewpage% lwarp
  \BlockClass{amsthmbody\LWR@thisthmstyle}% lwarp
  \trivlist
  \let\thmheadnl\relax
  \let\thm@swap\@gobble
  \thm@notefont{\fontseries\mddefault\upshape}%
  \thm@headpunct{.}% add period after heading
  \thm@headsep 5\p@ plus\p@ minus\p@\relax
  \thm@space@setup
  #1% style overrides
  \@topsep \thm@preskip               % used by thm head
  \@topsepadd \thm@postskip           % used by \@endparenv
  \def\@tempa{#2}\ifx\@empty\@tempa
    \def\@tempa{\@oparg{\@begintheorem{#3}{}}[]}%
  \else
    \refstepcounter{#2}%
    \def\@tempa{\@oparg{\@begintheorem{#3}{\csname the#2\endcsname}}[]}%
  \fi
  \@tempa
}
```

cleveref patches `\@thm` to do `\cref@thmoptarg` if an optional argument is given. lwarp then patches `\cref@thmoptarg` `\AtBeginDocument`.

```
\AtBeginDocument{%
\def\cref@thmoptarg[#1]#2#3#4{%
    \ifhmode\unskip\unskip\par\fi%
    \normalfont%
    \LWR@forcenewpage% lwarp
    \BlockClass{amsthmbody\LWR@thisthmstyle}% lwarp
    \trivlist%
    \let\thmheadnl\relax%
    \let\thm@swap\@gobble%
    \thm@notefont{\fontseries\mddefault\upshape}%
    \thm@headpunct{.}% add period after heading
    \thm@headsep 5\p@ plus\p@ minus\p@\relax%
    \thm@space@setup%
    #2% style overrides
    \@topsep \thm@preskip               % used by thm head
    \@topsepadd \thm@postskip           % used by \@endparenv
    \def\@tempa{#3}\ifx\@empty\@tempa%
        \def\@tempa{\@oparg{\@begintheorem{#4}{}}[]}%
```

```
    \else%
        \refstepcounter[#1]{#3}%  <<< cleveref modification
        \def\@tempa{\@oparg{\@begintheorem{#4}{\csname the#3\endcsname}}[]}%
    \fi%
    \@tempa
}%
}% AtBeginDocument

\def\@endtheorem{\endtrivlist\endBlockClass\@endpefalse }
```

Proof QED symbol:

```
\AtBeginDocument{
\@ifundefined{LWR@orig@openbox}{
\LetLtxMacro\LWR@orig@openbox\openbox
\LetLtxMacro\LWR@orig@blacksquare\blacksquare
\LetLtxMacro\LWR@orig@Box\Box

\def\openbox{\text{\HTMLunicode{25A1}}}% UTF-8 white box
\def\blacksquare{\text{\HTMLunicode{220E}}}% UTF-8 end-of-proof
\def\Box{\text{\HTMLunicode{25A1}}}% UTF-8 white box

\appto\LWR@restoreorigformatting{%
\LetLtxMacro\openbox\LWR@orig@openbox%
\LetLtxMacro\blacksquare\LWR@orig@blacksquare%
\LetLtxMacro\Box\LWR@orig@Box%
}% appto
}{}% @ifundefined
}% AtBeginDocument
```

Patched for css:

```
\renewenvironment{proof}[1][\proofname]{\par
\LWR@forcenewpage% lwarp
    \BlockClass{amsthmproof}% lwarp
  \pushQED{\qed}%
  \normalfont \topsep6\p@\@plus6\p@\relax
  \trivlist
  \item[
        \InlineClass{amsthmproofname}{#1\@addpunct{.}}]\ignorespaces% changes
}{%
  \InlineClass{theoremendmark}{\popQED}\endtrivlist%
  \endBlockClass% lwarp
  \@endpefalse
}
```

File 20 **lwarp-anonchap.sty**

§ 117 Package anonchap

(*Emulates or patches code by* Peter Wilson.)

Pkg anonchap anonchap is emulated.

Pkg tocloft

⚠ tocloft & other packages

If using tocloft with tocbibind, anonchap, fncychap, or other packages which change chapter title formatting, load tocloft with its `titles` option, which tells tocloft to use standard LaTeX commands to create the titles, allowing other packages to work with it.

The code is shared by tocbibind.

for HTML output:

```
\LWR@ProvidesPackageDrop{anonchap}[2009/08/03]

\newcommand{\simplechapter}[1][\@empty]{%
    \def\@chapcntformat##1{%
        #1~\csname the##1\endcsname\simplechapterdelim\quad%
    }%
}

\newcommand{\restorechapter}{%
\let\@chapcntformat\@seccntformat%
}
```

File 21 **lwarp-anysize.sty**

§ 118 Package

anysize

(*Emulates or patches code by* Michael Salzenberg, Thomas Esser.)

Pkg anysize

anysize is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{anysize}[1994/08/13]

\def\papersize#1#2{}
\def\marginsize#1#2#3#4{}
```

File 22 **lwarp-appendix.sty**

§ 119 Package

appendix

(*Emulates or patches code by* Peter Wilson.)

Pkg appendix

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:

```
\LWR@ProvidesPackagePass{appendix}[2009/09/02]

\renewcommand*{\@chap@pppage}{%
\part*{\appendixpagename}
 \if@dotoc@pp
 \addappheadtotoc
```

```
 \fi
}

\renewcommand*{\@sec@pppage}{%
\part*{\appendixpagename}
 \if@dotoc@pp
 \addappheadtotoc
 \fi
}
```

File 23 **lwarp-ar.sty**

§ 120 Package **ar**

(*Emulates or patches code by* Agostino De Marco.)

Pkg ar ar is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{ar}[2012/01/23]
```

Measure and print the width of the supplied glyph.

```
\newlength{\LWR@ar@width}

\newcommand*{\LWR@ar@printwidth}[1]{%
    \setlength{\LWR@ar@width}{\widthof{#1}}%
    width:%
    \LWR@convertto{em}{\the\LWR@ar@width}em%
}
```

The HTML version of \AR:

```
\newrobustcmd*{\LWR@HTML@AR}{%
```

Start a hashed lateximage, additionally hashed by the font series, with a width depending on the given glyph:

```
    \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:

```
    \ifmmode\else\csuse{LWR@orig\LWR@f@series series}\fi%
```

Print the original glyph using the newly set font series:

```
    \LWR@print@AR%
```

Done.

```
    \end{lateximage}%
}
```

Combine the print and HTML versions:

```
\LWR@formatted{AR}

\newrobustcmd*{\LWR@HTML@ARb}{%
    \begin{lateximage}*[AR][b][\LWR@ar@printwidth{\LWR@print@ARb}]%
    \LWR@print@ARb%
    \end{lateximage}%
}
\LWR@formatted{ARb}

\newrobustcmd*{\LWR@HTML@ARss}{%
  \begin{lateximage}*[ARss][\LWR@f@series][\LWR@ar@printwidth{\LWR@print@ARss}]%
    \ifmmode\else\csuse{LWR@orig\LWR@f@series series}\fi%
    \LWR@print@ARss%
    \end{lateximage}%
}
\LWR@formatted{ARss}

\newrobustcmd*{\LWR@HTML@ARssb}{%
    \begin{lateximage}*[AR][ssb][\LWR@ar@printwidth{\LWR@print@ARssb}]%
    \LWR@print@ARssb%
    \end{lateximage}%
}
\LWR@formatted{ARssb}

\newrobustcmd*{\LWR@HTML@ARtt}{%
    \begin{lateximage}*[AR][tt][\LWR@ar@printwidth{\LWR@print@ARtt}]%
    \LWR@print@ARtt%
    \end{lateximage}%
}
\LWR@formatted{ARtt}
```

File 24 **lwarp-arabicfront.sty**

§ 121 Package arabicfront

Pkg arabicfront arabicfront is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{arabicfront}[2006/09/03]
```

File 25 **lwarp-array.sty**

§ 122 Package array

Pkg array 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: Remove the default nullfied macros:

```
\let\firsthline\relax
\let\lasthline\relax

\LWR@ProvidesPackagePass{array}[2018/12/30]

\newcommand*{\LWR@HTML@firsthline}{\LWR@HTMLhline}%
\LWR@expandableformatted{firsthline}

\newcommand*{\LWR@HTML@lasthline}{\LWR@HTMLhline}%
\LWR@expandableformatted{lasthline}
```

File 26 **lwarp-arydshln.sty**

§ 123 Package

arydshln

(*Emulates or patches code by* Hiroshi Nakashima.)

Pkg arydshln 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 HTML output: array is required to allow `\newcolumn` below.

```
\RequirePackage{array}

\LWR@ProvidesPackageDrop{arydshln}[2018/09/26]
```

Ignored, but included for source compatibility:

```
\newdimen\dashlinedash \dashlinedash4pt %
\newdimen\dashlinegap \dashlinegap4pt %
\let\hdashlinewidth\dashlinedash
\let\hdashlinegap\dashlinegap

\def\ADLnullwide{}
\def\ADLsomewide{}
\def\ADLnullwidehline{}
\def\ADLsomewidehline{}

\def\ADLactivate{}
\def\ADLinactivate{}
\newcommand*{\ADLdrawingmode}[1]{}
\newcommand*{\ADLnoshorthanded}{}
\newcommand*{\dashgapcolor}[2][]{}
\newcommand*{\nodashgapcolor}{}
```

In a `lateximage`, revert to solid vertical rules:

```
\appto\LWR@restoreorigformatting{%
\newcolumntype{:}{|}%
\newcolumntype{;}[1]{|}%
\LetLtxMacro\hdashline\hline%
}
```

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`:

```
\def\hdashline{
%     \adl@hdashline\adl@ihdashline
    \adl@hdashline\adl@inactivehdl
}
\def\adl@hdashline#1{\noalign{\ifnum0=`}\fi
%         \ifadl@zwhrule \vskip-\arrayrulewidth
%         \else
%             \adl@hline\adl@connect\arrayrulewidth
                \hrule \@height \arrayrulewidth% lwarp
%         \fi
        \@ifnextchar[%]
                    {#1}%
                    {#1[%
%                         \dashlinedash/\dashlinegap
                       1pt/1pt
                    ]}}
% \def\adl@ihdashline[#1/#2]{\ifnum0=`{\fi}%
%         \multispan{\adl@columns}\unskip \adl@hcline\z@[#1/#2]%
%         \noalign{\ifnum0=`}\fi
%         \futurelet\@tempa\adl@xhline}
\def\adl@inactivehdl[#1/#2]{
%         \ifadl@zwhrule \vskip-\arrayrulewidth \fi
        \hrule\@height\arrayrulewidth
        \futurelet\@tempa\adl@xhline}
\def\adl@xhline{\ifx\@tempa\hline \adl@ixhline\fi
        \ifx\@tempa\hdashline \adl@ixhline\fi
        \ifnum0=`{\fi}}
\def\adl@ixhline{\vskip\doublerulesep \adl@hline\relax\doublerulesep}
\def\adl@hline#1#2{%
% \@tempcnta#2
%         \global\advance\adl@totalheight\@tempcnta
%         \xdef\adl@rowsL{\adl@rowsL
%                 (#1/\number\@tempcnta);}%
%         \xdef\adl@rowsR{\adl@rowsR
%                 (#1/\number\@tempcnta);}
}

\def\cdashline#1{\noalign{\ifnum0=`}\fi
        \@ifnextchar[%]
%                     {\adl@cdline[#1]}%
%                     {\adl@cdline[#1][\dashlinedash/\dashlinegap]}
                    {\adl@inactivecdl[#1]}%
```

```
                    {\adl@inactivecdl[#1][\dashlinedash/\dashlinegap]}
}

\def\adl@inactivecdl[#1-#2][#3]{\ifnum0=‘{\fi}\cline{#1-#2}}
```

File 27 **lwarp-asymptote.sty**

§ 124 Package

asymptote

(*Emulates or patches code by* Andy Hammerlindl, John Bowman, Tom Prince.)

Pkg asymptote 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:

```
\LWR@ProvidesPackagePass{asymptote}[2016/11/26]

\BeforeBeginEnvironment{asy}{\begin{lateximage}[(-asymptote-~\packagediagramname)]}
\AfterEndEnvironment{asy}{\end{lateximage}}

\xpatchcmd{\asyinclude}
    {\begingroup}
    {\begin{lateximage}[(-asymptote-~\packagediagramname)]}
    {}
    {\LWR@patcherror{asymptote}{asyinclude-begingroup}}

\xpatchcmd{\asyinclude}
    {\endgroup}
    {\end{lateximage}}
    {}
    {\LWR@patcherror{asymptote}{asyinclude-endgroup}}
```

File 28 **lwarp-atbegshi.sty**

§ 125 Package

atbegshi

(*Emulates or patches code by* Heiko Oberdiek.)

Pkg atbegshi Emulated.

for HTML output: Discard all options for lwarp-atbegshi:

```
\LWR@ProvidesPackageDrop{atbegshi}[2011/10/05]

\newcommand*{\AtBeginShipout}[1]{}
\newbox\AtBeginShipoutBox
\newcommand*{\AtBeginShipoutNext}[1]{}
\newcommand*{\AtBeginShipoutFirst}[1]{}
\newcommand*{\AtBeginShipoutDiscard}{}
\newcommand*{\AtBeginShipoutInit}{}
\newcommand*{\AtBeginShipoutAddToBox}[1]{}
\newcommand*{\AtBeginShipoutAddToBoxForeground}[1]{}
\newcommand*{\AtBeginShipoutUpperLeft}[1]{}
\newcommand*{\AtBeginShipoutUpperLeftForeground}[1]{}
\newcommand*{\AtBeginShipoutOriginalShipout}[1]{}
\def\AtBeginShipoutBoxWidth{0pt}
\def\AtBeginShipoutBoxHeight{0pt}
\def\AtBeginShipoutBoxDepth{0pt}

```

File 29 **lwarp-attachfile.sty**

§ 126 Package

attachfile

(*Emulates or patches code by* Scott Pakin.)

Pkg attachfile attachfile is patched for use by lwarp.

⚠ Metadata is ignored for now.

for HTML output:

```
\LWR@ProvidesPackagePass{attachfile}[2016/09/18]
```

Encloses each icon:

```
\newenvironment*{LWR@attachfile@icon}
{
    \begin{lateximage}*%
        [-attachfile-]%
        [%
            \detokenize\expandafter{\atfi@icon@icon}-%
```

```
            \detokenize\expandafter{\atfi@color@rgb}%
        ]%
}
{
    \end{lateximage}
}
```

Each icon is enclosed inside a `LWR@attachfile@icon` environment:

```
\xpretocmd{\atfi@acroGraph}{\LWR@attachfile@icon}{}{}
\xapptocmd{\atfi@acroGraph}{\endLWR@attachfile@icon}{}{}

\xpretocmd{\atfi@acroPaperclip}{\LWR@attachfile@icon}{}{}
\xapptocmd{\atfi@acroPaperclip}{\endLWR@attachfile@icon}{}{}

\xpretocmd{\atfi@acroPushPin}{\LWR@attachfile@icon}{}{}
\xapptocmd{\atfi@acroPushPin}{\endLWR@attachfile@icon}{}{}

\xpretocmd{\atfi@acroTag}{\LWR@attachfile@icon}{}{}
\xapptocmd{\atfi@acroTag}{\endLWR@attachfile@icon}{}{}
```

Disable PDF file embedding:

```
\DeclareRobustCommand{\atfi@embedfile}[1]{}
```

The displayed output for an `\attachfile` reference:

```
\newcommand*{\LWR@attachfile@appearance}{}

\DeclareRobustCommand{\atfi@set@appearance}[1]{%
    \def\LWR@attachfile@appearance{#1}%
}
```

A file annotation becomes a reference:

```
\DeclareRobustCommand{\atfi@insert@file@annot}[1]{%
    \href{#1}{\LWR@attachfile@appearance}%
}
```

File 30 **lwarp-attachfile2.sty**

§ 127 Package

attachfile2

(*Emulates or patches code by* Heiko Oberdiek.)

Pkg attachfile2 attachfile2 is patched for use by lwarp.

⚠ Metadata is ignored for now.

for HTML output:

```
\LWR@ProvidesPackagePass{attachfile2}[2016/05/16]
```

Adds memory of the selected color:

```
\def\LWR@attachfiletwo@color{}%

\define@key{AtFi}{color}{%
    \def\LWR@attachfiletwo@color{#1}%   lwarp
  \HyColor@AttachfileColor{#1}%
          \atfi@color@tex\atfi@color@inline\atfi@color@annot
          {attachfile2}{color}%
}
```

Encloses each icon:

```
\newenvironment*{LWR@attachfile@icon}
{
    \begin{lateximage}*%
        [-attachfile-]%
        [%
            \detokenize\expandafter{\atfi@icon@icon}-%
            \detokenize\expandafter{\LWR@attachfiletwo@color}%
        ]%
}
{
    \end{lateximage}
}
```

Each icon is enclosed inside a `LWR@attachfile@icon` environment:

```
\xpretocmd{\atfi@acroGraph}{\LWR@attachfile@icon}{}{}
\xapptocmd{\atfi@acroGraph}{\endLWR@attachfile@icon}{}{}

\xpretocmd{\atfi@acroPaperclip}{\LWR@attachfile@icon}{}{}
\xapptocmd{\atfi@acroPaperclip}{\endLWR@attachfile@icon}{}{}

\xpretocmd{\atfi@acroPushPin}{\LWR@attachfile@icon}{}{}
\xapptocmd{\atfi@acroPushPin}{\endLWR@attachfile@icon}{}{}

\xpretocmd{\atfi@acroTag}{\LWR@attachfile@icon}{}{}
\xapptocmd{\atfi@acroTag}{\endLWR@attachfile@icon}{}{}
```

Disable PDF file embedding:

```
\DeclareRobustCommand{\atfi@embedfile}[1]{}
```

The displayed output for an `\attachfile` reference:

```
\newcommand*{\LWR@attachfile@appearance}{}

\def\atfi@set@appearance@icon{%
    \atfi@set@appearance{\csname atfi@acro\atfi@icon@icon\endcsname}%
}

\DeclareRobustCommand{\atfi@set@appearance}[1]{%
    \def\LWR@attachfile@appearance{#1}%
}
```

A file annotation becomes a reference:

```
\DeclareRobustCommand{\atfi@insert@file@annot}[1]{%
    \href{#1}{\LWR@attachfile@appearance}%
}
```

Modified for text color:

```
\DeclareRobustCommand{\notextattachfile}[2][]{%
  \begingroup
    \atfi@setup{#1}%
    \ifatfi@print
      \leavevmode
      \begingroup
        \HyColor@UseColor\atfi@color@tex
        \LWR@textcurrentcolor{#2}%      lwarp
% \strut
      \endgroup
%     \else
%       \sbox\ltx@zero{#2\strut}%
%       \makebox[\wd0]{}%
    \fi
  \endgroup
}
```

Modified to draw the icon:

```
\DeclareRobustCommand{\noattachfile}[1][]{%
  \begingroup
    \atfi@setup{#1}%
    \atfi@set@appearance@icon
    \ifatfi@print
        \LWR@attachfile@appearance%     lwarp
%       \expandafter
%       \atfi@refxform\csname atfi@appobj@\atfi@icon@icon\endcsname
%     \else
%       \makebox[\atfi@appearancewidth]{}%
    \fi
  \endgroup
}
```

File 31 **lwarp-authblk.sty**

§ 128 Package

authblk

(*Emulates or patches code by* Patrick W. Daly.)

Pkg authblk

authblk is patched for HTML.

package support

⚠ load order

lwarp supports the native LaTeX titling commands, and also supports the packages authblk and titling. If both are used, authblk should be loaded before titling.

\published and \subtitle

If using the titling package, additional titlepage fields for \published and \subtitle

may be added by using \AddSubtitlePublished in the preamble. See section 65.8.

(*Emulates or patches code by* Patrick W. Daly.)

for HTML output: Require that authblk be loaded before titling:

```
\@ifpackageloaded{titling}{
\PackageError{lwarp-authblk}
{Package authblk must be loaded before titling}
{Titling appends authblk's author macro, so authblk must be loaded first.}
}
{}
```

Load authblk:

```
\LWR@ProvidesPackagePass{authblk}[2001/02/27]
```

Patch to add a class for the affiliation:

```
\LetLtxMacro\LWRAB@affil\affil

\renewcommand{\affil}[2][]{%
\LWRAB@affil[#1]{\protect\InlineClass{affiliation}{#2}}
}
```

Create an HTML break for an \authorcr:

```
\renewcommand*{\authorcr}{\protect\LWR@newlinebr}
```

File 32 **lwarp-axessibility.sty**

§ 129 Package axessibility

Pkg axessibility axessibility is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{axessibility}

\newcommand{\wrap}[1]{}
```

File 33 **lwarp-axodraw2.sty**

§ 130 Package axodraw2

(*Emulates or patches code by* John C. Collins, J.A.M. Vermaseren.)

Pkg axodraw2 axodraw2 is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{axodraw2}[2018/02/15]
```

```
\BeforeBeginEnvironment{axopicture}{\begin{lateximage}[(-axopicture-~\packagediagramname)]}

\AfterEndEnvironment{axopicture}{\end{lateximage}}
```

File 34 **lwarp-backref.sty**

§ 131 Package backref

(*Emulates or patches code by* David Carlisle and Sebastian Rahtz.)

Pkg backref 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:

```
\LWR@ProvidesPackagePass{backref}[2016/05/21]
```

Force the hyperref option:

```
\def\backref{}\let\backrefxxx\hyper@section@backref
```

File 35 **lwarp-balance.sty**

§ 132 Package balance

(*Emulates or patches code by* Patrick W. Daly.)

Pkg balance Emulated.

for HTML output: Discard all options for lwarp-balance:

```
\LWR@ProvidesPackageDrop{balance}[1999/02/23]

\newcommand*{\balance}{}
\newcommand*{\nobalance}{}
```

File 36 **lwarp-bbding.sty**

§ 133 Package bbding

(*Emulates or patches code by* Karel Horak, Peter Møller Neergaard.)

Pkg bbding bbding is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{bbding}[1999/04/15]
```

```
\newcommand*{\LWR@bbdingsymbol}[2]{\HTMLunicode{#2}}

\newcommand{\LWR@HTML@ScissorRightBrokenBottom}{\LWR@bbdingsymbol{000}  {2701}}
\newcommand{\LWR@HTML@ScissorRight}{\LWR@bbdingsymbol{001}              {2702}}
\newcommand{\LWR@HTML@ScissorRightBrokenTop}{\LWR@bbdingsymbol{002}    {2703}}
\newcommand{\LWR@HTML@ScissorLeftBrokenBottom}{\LWR@bbdingsymbol{003}  {2701}}
\newcommand{\LWR@HTML@ScissorLeft}{\LWR@bbdingsymbol{004}              {2702}}
\newcommand{\LWR@HTML@ScissorLeftBrokenTop}{\LWR@bbdingsymbol{005}     {2703}}
\newcommand{\LWR@HTML@ScissorHollowRight}{\LWR@bbdingsymbol{006}       {2704}}
\newcommand{\LWR@HTML@ScissorHollowLeft}{\LWR@bbdingsymbol{007}        {2704}}
\newcommand{\LWR@HTML@Phone}{\LWR@bbdingsymbol{010}                    {260E}}
\newcommand{\LWR@HTML@PhoneHandset}{\LWR@bbdingsymbol{011}             {2706}}
\newcommand{\LWR@HTML@Tape}{\LWR@bbdingsymbol{012}                     {2707}}
\newcommand{\LWR@HTML@Plane}{\LWR@bbdingsymbol{013}                    {2708}}
\newcommand{\LWR@HTML@Envelope}{\LWR@bbdingsymbol{014}                 {2709}}
\newcommand{\LWR@HTML@HandCuffRight}{\LWR@bbdingsymbol{015}            {261B}}
\newcommand{\LWR@HTML@HandCuffLeft}{\LWR@bbdingsymbol{016}             {261A}}
\newcommand{\LWR@HTML@HandCuffRightUp}{\LWR@bbdingsymbol{017}          {261D}}
\newcommand{\LWR@HTML@HandCuffLeftUp}{\LWR@bbdingsymbol{020}           {261F}}
\newcommand{\LWR@HTML@HandRight}{\LWR@bbdingsymbol{021}                {261E}}
\newcommand{\LWR@HTML@HandLeft}{\LWR@bbdingsymbol{022}                 {261C}}
\newcommand{\LWR@HTML@HandRightUp}{\LWR@bbdingsymbol{023}              {261D}}
\newcommand{\LWR@HTML@HandLeftUp}{\LWR@bbdingsymbol{024}               {261F}}
\newcommand{\LWR@HTML@Peace}{\LWR@bbdingsymbol{025}                    {270C}}
\newcommand{\LWR@HTML@HandPencilLeft}{\LWR@bbdingsymbol{026}           {270D}}
\newcommand{\LWR@HTML@PencilRight}{\LWR@bbdingsymbol{027}              {270F}}
\newcommand{\LWR@HTML@PencilLeft}{\LWR@bbdingsymbol{030}               {270F}}
\newcommand{\LWR@HTML@PencilRightUp}{\LWR@bbdingsymbol{031}            {2710}}
\newcommand{\LWR@HTML@PencilLeftUp}{\LWR@bbdingsymbol{032}             {2710}}
\newcommand{\LWR@HTML@PencilRightDown}{\LWR@bbdingsymbol{033}          {270E}}
\newcommand{\LWR@HTML@PencilLeftDown}{\LWR@bbdingsymbol{034}           {270E}}
\newcommand{\LWR@HTML@NibRight}{\LWR@bbdingsymbol{035}                 {2711}}
\newcommand{\LWR@HTML@NibLeft}{\LWR@bbdingsymbol{036}                  {2711}}
\newcommand{\LWR@HTML@NibSolidRight}{\LWR@bbdingsymbol{037}            {2712}}
\newcommand{\LWR@HTML@NibSolidLeft}{\LWR@bbdingsymbol{040}             {2712}}
\newcommand{\LWR@HTML@Checkmark}{\LWR@bbdingsymbol{041}                {2713}}
\newcommand{\LWR@HTML@CheckmarkBold}{\LWR@bbdingsymbol{042}            {2714}}
\newcommand{\LWR@HTML@XSolid}{\LWR@bbdingsymbol{043}                   {2715}}
\newcommand{\LWR@HTML@XSolidBold}{\LWR@bbdingsymbol{044}               {2716}}
\newcommand{\LWR@HTML@XSolidBrush}{\LWR@bbdingsymbol{045}              {2717}}
\newcommand{\LWR@HTML@PlusOutline}{\LWR@bbdingsymbol{046}              {2719}}
\newcommand{\LWR@HTML@Plus}{\LWR@bbdingsymbol{047}                     {271A}}
\newcommand{\LWR@HTML@PlusCenterOpen}{\LWR@bbdingsymbol{050}           {271C}}
\newcommand{\LWR@HTML@PlusThinCenterOpen}{\LWR@bbdingsymbol{051}       {271B}}
\newcommand{\LWR@HTML@Cross}{\LWR@bbdingsymbol{052}                    {271D}}
\newcommand{\LWR@HTML@CrossOpenShadow}{\LWR@bbdingsymbol{053}          {271E}}
\newcommand{\LWR@HTML@CrossOutline}{\LWR@bbdingsymbol{054}             {271F}}
\newcommand{\LWR@HTML@CrossBoldOutline}{\LWR@bbdingsymbol{055}         {271F}}
\newcommand{\LWR@HTML@CrossMaltese}{\LWR@bbdingsymbol{056}             {2720}}
\newcommand{\LWR@HTML@DavidStarSolid}{\LWR@bbdingsymbol{057}           {2721}}
\newcommand{\LWR@HTML@DavidStar}{\LWR@bbdingsymbol{060}                {2721}}
\newcommand{\LWR@HTML@FourAsterisk}{\LWR@bbdingsymbol{061}             {2722}}
\newcommand{\LWR@HTML@JackStar}{\LWR@bbdingsymbol{062}                 {2723}}
\newcommand{\LWR@HTML@JackStarBold}{\LWR@bbdingsymbol{063}             {2724}}
\newcommand{\LWR@HTML@CrossClowerTips}{\LWR@bbdingsymbol{064}          {2725}}
```

```
\newcommand{\LWR@HTML@FourStar}{\LWR@bbdingsymbol{065}                  {2726}}
\newcommand{\LWR@HTML@FourStarOpen}{\LWR@bbdingsymbol{066}              {2727}}
\newcommand{\LWR@HTML@FiveStarLines}{\LWR@bbdingsymbol{067}             {2729}}
\newcommand{\LWR@HTML@FiveStar}{\LWR@bbdingsymbol{070}                  {2605}}
\newcommand{\LWR@HTML@FiveStarOpen}{\LWR@bbdingsymbol{071}              {2729}}
\newcommand{\LWR@HTML@FiveStarOpenCircled}{\LWR@bbdingsymbol{072}       {272A}}
\newcommand{\LWR@HTML@FiveStarCenterOpen}{\LWR@bbdingsymbol{073}        {272B}}
\newcommand{\LWR@HTML@FiveStarOpenDotted}{\LWR@bbdingsymbol{074}        {272C}}
\newcommand{\LWR@HTML@FiveStarOutline}{\LWR@bbdingsymbol{075}           {272D}}
\newcommand{\LWR@HTML@FiveStarOutlineHeavy}{\LWR@bbdingsymbol{076}      {272E}}
\newcommand{\LWR@HTML@FiveStarConvex}{\LWR@bbdingsymbol{077}            {272F}}
\newcommand{\LWR@HTML@FiveStarShadow}{\LWR@bbdingsymbol{100}            {2730}}
\newcommand{\LWR@HTML@AsteriskBold}{\LWR@bbdingsymbol{101}              {2731}}
\newcommand{\LWR@HTML@AsteriskCenterOpen}{\LWR@bbdingsymbol{102}        {2732}}
\newcommand{\LWR@HTML@AsteriskThin}{\LWR@bbdingsymbol{103}              {273B}}
\newcommand{\LWR@HTML@AsteriskThinCenterOpen}{\LWR@bbdingsymbol{104}    {273C}}
\newcommand{\LWR@HTML@EightStarTaper}{\LWR@bbdingsymbol{105}            {2733}}
\newcommand{\LWR@HTML@EightStarConvex}{\LWR@bbdingsymbol{106}           {2735}}
\newcommand{\LWR@HTML@SixStar}{\LWR@bbdingsymbol{107}                   {2736}}
\newcommand{\LWR@HTML@EightStar}{\LWR@bbdingsymbol{110}                 {2737}}
\newcommand{\LWR@HTML@EightStarBold}{\LWR@bbdingsymbol{111}             {2738}}
\newcommand{\LWR@HTML@TwelweStar}{\LWR@bbdingsymbol{112}                {2739}}
\newcommand{\LWR@HTML@SixteenStarLight}{\LWR@bbdingsymbol{113}          {273A}}
\newcommand{\LWR@HTML@SixFlowerPetalRemoved}{\LWR@bbdingsymbol{114}     {273B}}
\newcommand{\LWR@HTML@SixFlowerOpenCenter}{\LWR@bbdingsymbol{115}       {273C}}
\newcommand{\LWR@HTML@Asterisk}{\LWR@bbdingsymbol{116}                  {273D}}
\newcommand{\LWR@HTML@SixFlowerAlternate}{\LWR@bbdingsymbol{117}        {273E}}
\newcommand{\LWR@HTML@FiveFlowerPetal}{\LWR@bbdingsymbol{120}           {273F}}
\newcommand{\LWR@HTML@SixFlowerPetalDotted}{\LWR@bbdingsymbol{121}      {2740}}
\newcommand{\LWR@HTML@FiveFlowerOpen}{\LWR@bbdingsymbol{122}            {2740}}
\newcommand{\LWR@HTML@EightFlowerPetal}{\LWR@bbdingsymbol{123}          {2741}}
\newcommand{\LWR@HTML@SunshineOpenCircled}{\LWR@bbdingsymbol{124}       {2742}}
\newcommand{\LWR@HTML@SixFlowerAltPetal}{\LWR@bbdingsymbol{125}         {2743}}
\newcommand{\LWR@HTML@FourClowerOpen}{\LWR@bbdingsymbol{126}            {273F}}
\newcommand{\LWR@HTML@FourClowerSolid}{\LWR@bbdingsymbol{127}           {273F}}
\newcommand{\LWR@HTML@AsteriskRoundedEnds}{\LWR@bbdingsymbol{130}       {2749}}
\newcommand{\LWR@HTML@EightFlowerPetalRemoved}{\LWR@bbdingsymbol{131}   {274A}}
\newcommand{\LWR@HTML@EightAsterisk}{\LWR@bbdingsymbol{132}             {274B}}
\newcommand{\LWR@HTML@SixFlowerRemovedOpenPetal}{\LWR@bbdingsymbol{133}  {2740}}
\newcommand{\LWR@HTML@SparkleBold}{\LWR@bbdingsymbol{134}               {2748}}
\newcommand{\LWR@HTML@Sparkle}{\LWR@bbdingsymbol{135}                   {2747}}
\newcommand{\LWR@HTML@SnowflakeChevron}{\LWR@bbdingsymbol{136}          {2744}}
\newcommand{\LWR@HTML@SnowflakeChevronBold}{\LWR@bbdingsymbol{137}      {2746}}
\newcommand{\LWR@HTML@Snowflake}{\LWR@bbdingsymbol{140}                 {2744}}
\newcommand{\LWR@HTML@CircleSolid}{\LWR@bbdingsymbol{141}               {25CF}}
\newcommand{\LWR@HTML@Ellipse}{\LWR@bbdingsymbol{142}                   {274D}}
\newcommand{\LWR@HTML@EllipseSolid}{\LWR@bbdingsymbol{143}              {25CF}}
\newcommand{\LWR@HTML@CircleShadow}{\LWR@bbdingsymbol{144}              {274D}}
\newcommand{\LWR@HTML@EllipseShadow}{\LWR@bbdingsymbol{145}             {274D}}
\newcommand{\LWR@HTML@Square}{\LWR@bbdingsymbol{146}                    {25A1}}
\newcommand{\LWR@HTML@SquareSolid}{\LWR@bbdingsymbol{147}               {25A0}}
\newcommand{\LWR@HTML@SquareShadowBottomRight}{\LWR@bbdingsymbol{150}   {2751}}
\newcommand{\LWR@HTML@SquareShadowTopRight}{\LWR@bbdingsymbol{151}      {2752}}
\newcommand{\LWR@HTML@SquareShadowTopLeft}{\LWR@bbdingsymbol{152}       {2752}}
\newcommand{\LWR@HTML@SquareCastShadowBottomRight}{\LWR@bbdingsymbol{153} {2751}}
```

```
\newcommand{\LWR@HTML@SquareCastShadowTopRight}{\LWR@bbdingsymbol{154}   {2752}}
\newcommand{\LWR@HTML@SquareCastShadowTopLeft}{\LWR@bbdingsymbol{155}   {2752}}
\newcommand{\LWR@HTML@TriangleUp}{\LWR@bbdingsymbol{156}                {25B2}}
\newcommand{\LWR@HTML@TriangleDown}{\LWR@bbdingsymbol{157}              {25BC}}
\newcommand{\LWR@HTML@DiamondSolid}{\LWR@bbdingsymbol{160}              {25C6}}
\newcommand{\LWR@HTML@OrnamentDiamondSolid}{\LWR@bbdingsymbol{161}     {2756}}
\newcommand{\LWR@HTML@HalfCircleRight}{\LWR@bbdingsymbol{162}           {25D7}}
\newcommand{\LWR@HTML@HalfCircleLeft}{\LWR@bbdingsymbol{163}            {25D6}}
\newcommand{\LWR@HTML@RectangleThin}{\LWR@bbdingsymbol{164}             {2758}}
\newcommand{\LWR@HTML@Rectangle}{\LWR@bbdingsymbol{165}                 {2759}}
\newcommand{\LWR@HTML@RectangleBold}{\LWR@bbdingsymbol{166}             {275A}}
\newcommand{\LWR@HTML@ArrowBoldRightStrobe}{\LWR@bbdingsymbol{167}     {27A0}}
\newcommand{\LWR@HTML@ArrowBoldUpRight}{\LWR@bbdingsymbol{170}         {27A6}}
\newcommand{\LWR@HTML@ArrowBoldDownRight}{\LWR@bbdingsymbol{171}       {27A5}}
\newcommand{\LWR@HTML@ArrowBoldRightShort}{\LWR@bbdingsymbol{172}      {27A7}}
\newcommand{\LWR@HTML@ArrowBoldRightCircled}{\LWR@bbdingsymbol{173}    {27B2}}


\LWR@formatted{ScissorRightBrokenBottom}
\LWR@formatted{ScissorRight}
\LWR@formatted{ScissorRightBrokenTop}
\LWR@formatted{ScissorLeftBrokenBottom}
\LWR@formatted{ScissorLeft}
\LWR@formatted{ScissorLeftBrokenTop}
\LWR@formatted{ScissorHollowRight}
\LWR@formatted{ScissorHollowLeft}
\LWR@formatted{Phone}
\LWR@formatted{PhoneHandset}
\LWR@formatted{Tape}
\LWR@formatted{Plane}
\LWR@formatted{Envelope}
\LWR@formatted{HandCuffRight}
\LWR@formatted{HandCuffLeft}
\LWR@formatted{HandCuffRightUp}
\LWR@formatted{HandCuffLeftUp}
\LWR@formatted{HandRight}
\LWR@formatted{HandLeft}
\LWR@formatted{HandRightUp}
\LWR@formatted{HandLeftUp}
\LWR@formatted{Peace}
\LWR@formatted{HandPencilLeft}
\LWR@formatted{PencilRight}
\LWR@formatted{PencilLeft}
\LWR@formatted{PencilRightUp}
\LWR@formatted{PencilLeftUp}
\LWR@formatted{PencilRightDown}
\LWR@formatted{PencilLeftDown}
\LWR@formatted{NibRight}
\LWR@formatted{NibLeft}
\LWR@formatted{NibSolidRight}
\LWR@formatted{NibSolidLeft}
\LWR@formatted{Checkmark}
\LWR@formatted{CheckmarkBold}
\LWR@formatted{XSolid}
\LWR@formatted{XSolidBold}
```

```
\LWR@formatted{XSolidBrush}
\LWR@formatted{PlusOutline}
\LWR@formatted{Plus}
\LWR@formatted{PlusCenterOpen}
\LWR@formatted{PlusThinCenterOpen}
\LWR@formatted{Cross}
\LWR@formatted{CrossOpenShadow}
\LWR@formatted{CrossOutline}
\LWR@formatted{CrossBoldOutline}
\LWR@formatted{CrossMaltese}
\LWR@formatted{DavidStarSolid}
\LWR@formatted{DavidStar}
\LWR@formatted{FourAsterisk}
\LWR@formatted{JackStar}
\LWR@formatted{JackStarBold}
\LWR@formatted{CrossClowerTips}
\LWR@formatted{FourStar}
\LWR@formatted{FourStarOpen}
\LWR@formatted{FiveStarLines}
\LWR@formatted{FiveStar}
\LWR@formatted{FiveStarOpen}
\LWR@formatted{FiveStarOpenCircled}
\LWR@formatted{FiveStarCenterOpen}
\LWR@formatted{FiveStarOpenDotted}
\LWR@formatted{FiveStarOutline}
\LWR@formatted{FiveStarOutlineHeavy}
\LWR@formatted{FiveStarConvex}
\LWR@formatted{FiveStarShadow}
\LWR@formatted{AsteriskBold}
\LWR@formatted{AsteriskCenterOpen}
\LWR@formatted{AsteriskThin}
\LWR@formatted{AsteriskThinCenterOpen}
\LWR@formatted{EightStarTaper}
\LWR@formatted{EightStarConvex}
\LWR@formatted{SixStar}
\LWR@formatted{EightStar}
\LWR@formatted{EightStarBold}
\LWR@formatted{TwelweStar}
\LWR@formatted{SixteenStarLight}
\LWR@formatted{SixFlowerPetalRemoved}
\LWR@formatted{SixFlowerOpenCenter}
\LWR@formatted{Asterisk}
\LWR@formatted{SixFlowerAlternate}
\LWR@formatted{FiveFlowerPetal}
\LWR@formatted{SixFlowerPetalDotted}
\LWR@formatted{FiveFlowerOpen}
\LWR@formatted{EightFlowerPetal}
\LWR@formatted{SunshineOpenCircled}
\LWR@formatted{SixFlowerAltPetal}
\LWR@formatted{FourClowerOpen}
\LWR@formatted{FourClowerSolid}
\LWR@formatted{AsteriskRoundedEnds}
\LWR@formatted{EightFlowerPetalRemoved}
\LWR@formatted{EightAsterisk}
\LWR@formatted{SixFlowerRemovedOpenPetal}
```

```
\LWR@formatted{SparkleBold}
\LWR@formatted{Sparkle}
\LWR@formatted{SnowflakeChevron}
\LWR@formatted{SnowflakeChevronBold}
\LWR@formatted{Snowflake}
\LWR@formatted{CircleSolid}
\LWR@formatted{Ellipse}
\LWR@formatted{EllipseSolid}
\LWR@formatted{CircleShadow}
\LWR@formatted{EllipseShadow}
\LWR@formatted{Square}
\LWR@formatted{SquareSolid}
\LWR@formatted{SquareShadowBottomRight}
\LWR@formatted{SquareShadowTopRight}
\LWR@formatted{SquareShadowTopLeft}
\LWR@formatted{SquareCastShadowBottomRight}
\LWR@formatted{SquareCastShadowTopRight}
\LWR@formatted{SquareCastShadowTopLeft}
\LWR@formatted{TriangleUp}
\LWR@formatted{TriangleDown}
\LWR@formatted{DiamondSolid}
\LWR@formatted{OrnamentDiamondSolid}
\LWR@formatted{HalfCircleRight}
\LWR@formatted{HalfCircleLeft}
\LWR@formatted{RectangleThin}
\LWR@formatted{Rectangle}
\LWR@formatted{RectangleBold}
\LWR@formatted{ArrowBoldRightStrobe}
\LWR@formatted{ArrowBoldUpRight}
\LWR@formatted{ArrowBoldDownRight}
\LWR@formatted{ArrowBoldRightShort}
\LWR@formatted{ArrowBoldRightCircled}
```

File 37 **lwarp-biblatex.sty**

§ 134 Package

biblatex

(*Emulates or patches code by* Philipp Lehman.)

Pkg biblatex 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.

```
\ifdef{\newfloat@addtocontents@ORI}{
    \let\addtocontents\newfloat@addtocontents@ORI
}{}

\LWR@ProvidesPackagePass{biblatex}[2018/03/04]
```

File 38 **lwarp-bibunits.sty**

§ 135 Package

bibunits

(*Emulates or patches code by* Thorsten Hansen.)

Pkg bibunits bibunits is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{bibunits}[2004/05/12]

\def\bu@bibdata{\BaseJobname}
```

File 39 **lwarp-bigdelim.sty**

§ 136 Package

bigdelim

(*Emulates or patches code by* Piet van Oostrum, Øystein Bache, Jerry Leichter.)

Pkg bigdelim 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{\{}{2}{.25in}[left ] & c & d \\
\mrowcell & e & f \\
<empty> & g & h \\
\end{tabular}
```

<>	a	b
left {	c	d
	e	f
<>	g	h

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:

```
\let\ldelim\relax
\let\rdelim\relax
```

Next, load the package's new definitions:

```
\LWR@ProvidesPackagePass{bigdelim}[2018/08/03]
```

`\ldelim` `\rdelim` {⟨*1:delimiter*⟩} {⟨*2:#rows*⟩} {⟨*3:width*⟩} [⟨*4:text*⟩]

```
\NewDocumentCommand{\LWR@HTML@ldelim}{m m m O{}}{%
\renewcommand{\LWR@multirowborder}{right}%
\multirow{#2}{#3}{#4 \InlineClass{ldelim}{#1}}%
}

\LWR@formatted{ldelim}

\NewDocumentCommand{\LWR@HTML@rdelim}{m m m O{}}{%
\renewcommand{\LWR@multirowborder}{left}%
\multirow{#2}{#3}{\InlineClass{rdelim}{#1} #4}%
}

\LWR@formatted{rdelim}
```

File 40 **lwarp-bigstrut.sty**

§ 137 Package bigstrut

(*Emulates or patches code by* Piet van Oostrum, Øystein Bache, Jerry Leichter.)

`Pkg bigstrut` bigstrut is used as-is for print or `lateximage`, and patched for HTML.

for HTML output:

```
\LWR@ProvidesPackagePass{bigstrut}[2018/08/03]

\LetLtxMacro\LWR@origbigstrut\bigstrut

\renewcommand\bigstrut[1][x]{}

\appto\LWR@restoreorigformatting{%
\LetLtxMacro\bigstrut\LWR@origbigstrut%
}
```

File 41 **lwarp-bitpattern.sty**

§ 138 Package bitpattern

(*Emulates or patches code by* Jean-Marc Bourguet.)

Pkg bitpattern bitpattern is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{bitpattern}[2015/12/11]

\xpatchcmd{\bitpattern}
    {\begingroup}
    {\begin{lateximage}[(-bitpattern-~\packagediagramname)]}
    {}
    {\LWR@patcherror{bitpattern}{bitpattern}}

\xpatchcmd{\bp@Done}
    {\endgroup}
    {\end{lateximage}}
    {}
    {\LWR@patcherror{bitpattern}{bp@Done}}
```

File 42 **lwarp-blowup.sty**

§ 139 Package blowup

Pkg blowup blowup is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{blowup}[2018/01/02]

\newcommand*\blowUp[1]{}
```

File 43 **lwarp-booklet.sty**

§ 140 Package booklet

(*Emulates or patches code by* Peter Wilson.)

Pkg booklet booklet is nullified.

for HTML output:

```
\LWR@ProvidesPackageDrop{booklet}[2009/09/02]

\newdimen\pageseplength
\newdimen\pagesepwidth
\newdimen\pagesepoffset
\newif\ifsidebyside     \sidebysidetrue
\newif\ifuselandscape   \uselandscapefalse
\newif\ifprintoption    \printoptionfalse
\newcommand*{\pagespersignature}[1]{}
\def\magstepminus#1{}
\newcommand*{\target}[3]{}
\newcommand*{\source}[3]{}
\newcommand*{\setpdftargetpages}{}
\newcommand*{\setdvipstargetpages}{}
\newcommand*{\targettopbottom}{}
```

```
\newcommand*{\twoupemptypage}{}
\newcommand*{\twoupclearpage}{}
\newcommand*{\checkforlandscape}{}
```

File 44 **lwarp-bookmark.sty**

§ 141 Package bookmark

(*Emulates or patches code by* Heiko Oberdiek.)

Pkg bookmark bookmark is emulated.

for HTML output: Discard all options for lwarp-bookmark:

```
\LWR@ProvidesPackageDrop{bookmark}[2016/05/17]

\newcommand*{\bookmarksetup}[1]{}
\newcommand*{\bookmarksetupnext}[1]{}
\newcommand*{\bookmark}[2][]{}
\newcommand*{\bookmarkdefinestyle}[2]{}
\newcommand*{\bookmarkget}[1]{}
\newcommand{\BookmarkAtEnd}[1]{}
```

File 45 **lwarp-booktabs.sty**

§ 142 Package booktabs

(*Emulates or patches code by* Simon Fear.)

Pkg booktabs booktabs is emulated during HTML output, and used as-is during print output and inside an HTML `latexmage`.

for HTML output: First, forget the placeholder macros:

```
\LetLtxMacro\toprule\relax
\LetLtxMacro\midrule\relax
\LetLtxMacro\cmidrule\cline
\LetLtxMacro\bottomrule\relax
\LetLtxMacro\addlinespace\relax
\LetLtxMacro\morecmidrules\relax
\LetLtxMacro\specialrule\relax

\LWR@ProvidesPackagePass{booktabs}[2016/04/27]

\DeclareDocumentCommand{\LWR@HTML@toprule}{o d()}%
    {%
        \IfValueTF{#1}%
            {\LWR@docmidrule[#1](){1-\arabic{LWR@tabletotalLaTeXcols}}}%
            {%
                \ifbool{FormatWP}%
```

```
                {\LWR@docmidrule[#1](){1-\arabic{LWR@tabletotalLaTeXcols}}}%
                {\global\booltrue{LWR@doingtbrule}}%
            }%
    \LWR@getmynexttoken}

\LWR@expandableformatted{toprule}

\DeclareDocumentCommand{\LWR@HTML@midrule}{o d()}%
    {%
        \IfValueTF{#1}%
            {\LWR@docmidrule[#1](){1-\arabic{LWR@tabletotalLaTeXcols}}}%
            {%
                \ifbool{FormatWP}%
                {\LWR@docmidrule[#1](){1-\arabic{LWR@tabletotalLaTeXcols}}}%
                {\addtocounter{LWR@hlines}{1}}%
            }%
    \LWR@getmynexttoken}

\LWR@expandableformatted{midrule}

\DeclareDocumentCommand{\LWR@HTML@cmidrule}{O{\LWR@cmidrulewidth} d() m}{%
    \LWR@docmidrule[#1](#2){#3}%
    \LWR@getmynexttoken%
}%

\LWR@expandableformatted{cmidrule}

\DeclareDocumentCommand{\LWR@HTML@bottomrule}{o d()}{%
    \IfValueTF{#1}%
        {\LWR@docmidrule[#1](){1-\arabic{LWR@tabletotalLaTeXcols}}}%
        {%
            \ifbool{FormatWP}%
            {\LWR@docmidrule[#1](){1-\arabic{LWR@tabletotalLaTeXcols}}}%
            {\global\booltrue{LWR@doingtbrule}}%
        }%
    \LWR@getmynexttoken%
}%

\LWR@expandableformatted{bottomrule}

\DeclareDocumentCommand{\LWR@HTML@addlinespace}{o}{}%

\LWR@expandableformatted{addlinespace}

\DeclareDocumentCommand{\LWR@HTML@morecmidrules}{}{}%

\LWR@expandableformatted{morecmidrules}

\DeclareDocumentCommand{\LWR@HTML@specialrule}{m m m d()}%
  {\LWR@docmidrule[#1](){1-\arabic{LWR@tabletotalLaTeXcols}}\LWR@getmynexttoken}%

\LWR@expandableformatted{specialrule}
```

File 46 **lwarp-bophook.sty**

§ 143 Package

bophook

Pkg bophook bophook is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{bophook}[2001/03/29]

\newcommand*{\AtBeginPage}[1]{}
\newcommand*{\PageLayout}[1]{}
```

File 47 **lwarp-bounddvi.sty**

§ 144 Package

bounddvi

Pkg bounddvi bounddvi is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{bounddvi}[2016/12/28]
```

File 48 **lwarp-boxedminipage2e.sty**

§ 145 Package

boxedminipage2e

(*Emulates or patches code by* Scott Pakin.)

Pkg boxedminipage2e boxedminipage2e is emulated.

for HTML output: Discard all options for lwarp-boxedminipage2e:

```
\LWR@ProvidesPackageDrop{boxedminipage2e}[2015/03/09]

\newenvironment{boxedminipage}{%
\begin{BlockClass}{framebox}%
\minipage%
}
{
\endminipage%
\end{BlockClass}
}
```

File 49 **lwarp-breakurl.sty**

§ 146 Package breakurl

(*Emulates or patches code by* Vilar Camara Neto.)

Pkg breakurl breakurl is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{breakurl}[2013/04/10]

\LetLtxMacro\burl\url

\NewDocumentCommand{\LWR@burlaltb}{O{} +m m}{%
\LWR@ensuredoingapar%
\LWR@subhyperref{#2}%
\LWR@subhyperreftext{#3}%
\endgroup% restore catcodes
}

\newrobustcmd*{\burlalt}{%
\begingroup%
\catcode`\#=12%
\catcode`\%=12%
\catcode`\&=12%
\catcode`\~=12%
\catcode`\_=12%
\LWR@burlaltb%
}

\LetLtxMacro\urlalt\burlalt
```

File 50 **lwarp-breqn.sty**

§ 147 Package breqn

(*Emulates or patches code by* Michael J. Downes, Morten Høgholm.)

Pkg breqn breqn is patched for use by lwarp.

⚠ darray darray is not supported.

⚠ MathJax MathJax does not support breqn.

for HTML output:

```
\LWR@ProvidesPackagePass{breqn}[2017/01/27]

\setkeys{breqn}{spread={5pt}}

\def\eqnumside{R}
```

```
% \def\eqnumplace{T}

\BeforeBeginEnvironment{dmath}{
    \begin{BlockClass}{displaymathnumbered}
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[-breqn dmath- \mathimagename]
}

\AfterEndEnvironment{dmath}{
    \end{lateximage}\end{BlockClass}
}

\BeforeBeginEnvironment{dmath*}{
    \begin{BlockClass}{displaymath}
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[-breqn dmath*- \mathimagename]
}

\AfterEndEnvironment{dmath*}{
    \end{lateximage}\end{BlockClass}
}

\BeforeBeginEnvironment{dseries}{
    \begin{BlockClass}{displaymathnumbered}
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[-breqn dseries- \mathimagename]
}

\AfterEndEnvironment{dseries}{
    \end{lateximage}\end{BlockClass}
}

\BeforeBeginEnvironment{dseries*}{
    \begin{BlockClass}{displaymath}
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[-breqn dseries*- \mathimagename]
}

\AfterEndEnvironment{dseries*}{
    \end{lateximage}\end{BlockClass}
}

\BeforeBeginEnvironment{dgroup}{
    \begin{BlockClass}{displaymath}
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[-breqn dgroup- \mathimagename]
}

\AfterEndEnvironment{dgroup}{
    \end{lateximage}\end{BlockClass}
```

```
}

\BeforeBeginEnvironment{dgroup*}{
    \begin{BlockClass}{displaymath}
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[-breqn dgroup*- \mathimagename]
}

\AfterEndEnvironment{dgroup*}{
    \end{lateximage}\end{BlockClass}
}
```

File 51 **lwarp-bsheaders.sty**

§ 148 Package bsheaders

Pkg bsheaders bsheaders is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{bsheaders}[1997/10/06]
```

File 52 **lwarp-bxpapersize.sty**

§ 149 Package bxpapersize

Pkg bxpapersize bxpapersize is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{bxpapersize}[2017/10/08]

\providecommand*\papersizesetup{\bxpapersizesetup}
\newcommand*\bxpapersizesetup[1]{}
```

File 53 **lwarp-bytefield.sty**

§ 150 Package bytefield

(*Emulates or patches code by* Scott Pakin.)

Pkg bytefield bytefield is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{bytefield}[2017/09/15]

\BeforeBeginEnvironment{bytefield}{\begin{lateximage}[(-bytefield-~\packagediagramname)]}

\AfterEndEnvironment{bytefield}{\end{lateximage}}
```

File 54 **lwarp-cancel.sty**

§ 151 Package

cancel

Pkg cancel

cancel is used as-is for SVG math, and emulated for HTML text output.

for HTML output:

```
\LWR@origRequirePackage{lwarp-xcolor}% for \convertcolorspec
\LWR@ProvidesPackagePass{cancel}[2013/04/12]
```

\cancelto is math-only, so is used as-is.

```
\LetLtxMacro\LWR@origcancel\cancel
\LetLtxMacro\LWR@origbcancel\bcancel
\LetLtxMacro\LWR@origxcancel\xcancel

\appto\LWR@restoreorigformatting{%
\LetLtxMacro\cancel\LWR@origcancel%
\LetLtxMacro\bcancel\LWR@origbcancel%
\LetLtxMacro\xcancel\LWR@origxcancel%
}
```

\LWR@cancelcolor {⟨*text*⟩} {⟨*color*⟩} {⟨*class*⟩} {⟨*colorstyle*⟩} {⟨*FormatWPstyle*⟩}

Add colors if not empty:

```
\newcommand{\LWR@cancelcolor}[5]{%
\ifcsempty{#2}%
{\InlineClass(#5){#3}{#1}}%
{\LWR@htmlspanclass[#5;#4:\LWR@origpound\LWR@tempcolor]{#3}{#1}}%
}
```

\cancel {⟨*text*⟩}

```
\DeclareRobustCommand{\cancel}[1]{%
\begingroup%
\CancelColor%
\LWR@findcurrenttextcolor%
\color{black}%
\LWR@cancelcolor{#1}{LWR@tempcolor}{sout}{text-decoration-color}%
    {text-decoration:line-through}%
\endgroup%
}

\LetLtxMacro\bcancel\cancel
\LetLtxMacro\xcancel\cancel
```

File 55 **lwarp-canoniclayout.sty**

§ 152 Package canoniclayout

Pkg canoniclayout canoniclayout is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{canoniclayout}[2011/11/05]

\newcommand*{\currentfontletters}{}
\newcommand*{\charactersperpage}{}
```

File 56 **lwarp-caption.sty**

§ 153 Package caption

(*Emulates or patches code by* Axel Sommerfeldt.)

Pkg caption caption is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{caption}[2018/10/05]

\renewcommand\caption@ibox[3]{%
  \@testopt{\caption@iibox{#1}{#2}{#3}}{%
%         \wd\@tempboxa%
        \linewidth% lwarp
  }%
% \LWR@traceinfo{caption@ibox: done}%
}

\long\def\caption@iibox#1#2#3[#4]{%
  \@testopt{\caption@iiibox{#1}{#2}{#3}{#4}}\captionbox@hj@default
}

\long\def\caption@iiibox#1#2#3#4[#5]#6{%
%   \setbox\@tempboxa\hbox{#6}%
  \begingroup
  #1*% set \caption@position
  \caption@iftop{%
    \LWR@traceinfo{caption@iiibox top}%
    \endgroup
    \parbox[t]{#4}{%
      #1\relax
      \caption@setposition t%
%       \vbox{\caption#2{#3}}%
      {\caption#2{#3}}% lwarp
%       \captionbox@hrule
%       \csname caption@hj@#5\endcsname
%       \unhbox\@tempboxa
```

```
      #6% lwarp
    }%
  }{%
    \LWR@traceinfo{caption@iiibox bottom}%
    \endgroup
    \parbox[b]{#4}{%
      #1\relax
      \caption@setposition b%
%       \csname caption@hj@#5\endcsname
%       \unhbox\@tempboxa
         #6% lwarp
%       \captionbox@hrule
%       \vtop{\caption#2{#3}}}%
      {\caption#2{#3}}% lwarp
     }%
  }%
\LWR@traceinfo{caption@iiibox: done}%
}

\def\caption@caption{%
  \caption@iftype
    {%
     \caption@checkgrouplevel\@empty\caption
     \caption@star
       {\caption@refstepcounter\@captype}%
       {\caption@dblarg{\@caption\@captype}}}%
    {\caption@Error{\noexpand\caption outside float}%
     \caption@gobble}%
}

\long\def\caption@@caption#1[#2]#3{%

  \ifcaption@star \else
    \caption@prepareanchor{#1}{#2}%
    \memcaptioninfo{#1}{\csname the#1\endcsname}{#2}{#3}%
    \@nameuse{nag@hascaptiontrue}%
  \fi

  \par
  \caption@beginex{#1}{#2}{#3}%
    \caption@setfloatcapt{%
      \caption@boxrestore
      \if@minipage
        \@setminipage
      \fi
      \caption@normalsize
      \ifcaption@star
        \let\caption@makeanchor\@firstofone
      \fi
      \@makecaption{\csname fnum@#1\endcsname}%
                   {\ignorespaces\caption@makeanchor{#3}}\par
      \caption@if@minipage\@minipagetrue\@minipagefalse}%
  \caption@end%
}
```

\caption@@@make {⟨*caption label*⟩} {⟨*caption text*⟩}

```
\renewcommand\caption@@@make[2]{%
\LWR@startpars% lwarp
%   \sbox\@tempboxa{#1}%
%   \ifdim\wd\@tempboxa=\z@
%     \let\caption@lsep\relax
%   \fi
  \caption@ifempty{#2}{%
    \let\caption@lsep\@empty
    \let\caption@tfmt\@firstofone
  }%
  \@setpar{\LWR@closeparagraph\@@par}% lwarp
  \caption@applyfont
  \caption@fmt
    {\ifcaption@star\else
       \begingroup
         \captionlabelfont
         #1%
       \endgroup
     \fi}%
    {\ifcaption@star\else
       \begingroup
         \caption@iflf\captionlabelfont
         \relax\caption@lsep
       \endgroup
     \fi}%
    {{\captiontextfont
      \caption@ifstrut
        {\vrule\@height\ht\strutbox\@width\z@}%
        {}%
      \nobreak\hskip\z@skip % enable hyphenation
      \caption@tfmt{#2}
    \LWR@ensuredoingapar% lwarp
      \caption@ifstrut
        {\ifhmode\@finalstrut\strutbox\fi}%
        {}%
      \par}}
\LWR@stoppars% lwarp
}
```

\caption@@make@ {⟨⟩} {⟨⟩}

```
\renewcommand{\caption@@make@}[2]{%
  \caption@stepthecounter
  \caption@beginhook
      \caption@@@make{#1}{#2}%
  \caption@endhook
}

% \DeclareCaptionBox{none}{#2}
\DeclareCaptionBox{parbox}{%
#2%
}
\DeclareCaptionBox{colorbox}{%
#2%
```

```
129 }
```

File 57 **lwarp-cases.sty**

§ 154 Package cases

(*Emulates or patches code by* Donald Arseneau.)

Pkg cases cases is patched for use by lwarp.

⚠ MathJax MathJax does not support cases.

for HTML output:

```
1 \LWR@ProvidesPackagePass{cases}[2002/05/02]

2 \BeforeBeginEnvironment{numcases}{
3     \begin{BlockClass}{displaymathnumbered}
4     \LWR@newautoidanchor%
5     \booltrue{LWR@indisplaymathimage}%
6     \begin{lateximage}[-cases- \mathimagename]
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- \mathimagename]
18 }
19
20 \AfterEndEnvironment{subnumcases}{
21    \end{lateximage}\end{BlockClass}
22 }
```

File 58 **lwarp-changebar.sty**

§ 155 Package changebar

Pkg changebar 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}{}
```

```
\newcommand*{\cbcolor}[1]{}
\newlength{\changebarwidth}
\newlength{\deletebarwidth}
\newlength{\changebarsep}
\newcounter{changebargrey}
```

File 59 **lwarp-changepage.sty**

§ 156 Package changepage

(*Emulates or patches code by* Peter Wilson.)

Pkg changepage changepage is emulated.

for HTML output: Discard all options for lwarp-changepage:

```
\LWR@ProvidesPackageDrop{changepage}[2009/10/20]

\newif\ifoddpage
\DeclareRobustCommand{\checkoddpage}{\oddpagetrue}
\DeclareRobustCommand{\changetext}[5]{}
\DeclareRobustCommand{\changepage}[9]{}

\@ifundefined{adjustwidth}{
\newenvironment{adjustwidth}[2]{}{}
\newenvironment{adjustwidth*}[2]{}{}
}{
\renewenvironment{adjustwidth}[2]{}{}
\renewenvironment{adjustwidth*}[2]{}{}
}

\DeclareDocumentCommand{\strictpagecheck}{}{}
\DeclareDocumentCommand{\easypagecheck}{}{}
```

File 60 **lwarp-changes.sty**

§ 157 Package changes

(*Emulates or patches code by* Ekkart Kleinod.)

Pkg changes changes is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{changes}[2015/04/27]

\xpatchcmd{\Changes@output}
    {#1}
    {\LWR@textcurrentcolor{#1}}
    {}
    {\LWR@patcherror{changes}{Changes@output}}
```

```
\xpatchcmd{\listofchanges}
    {
        \ifthenelse{\equal{\Changes@InID}{\@empty}}
        {\changesauthorname: \changesanonymousname}
        {%
        \changesauthorname: \Changes@InID
        \ifthenelse{\equal{\Changes@InName}{\@empty}}
        {}
        { (\Changes@InName)}
        }
    }
    {
        \LWR@textcurrentcolor{% lwarp
            \ifthenelse{\equal{\Changes@InID}{\@empty}}
            {\changesauthorname: \changesanonymousname}
            {%
            \changesauthorname: \Changes@InID
            \ifthenelse{\equal{\Changes@InName}{\@empty}}
            {}
            { (\Changes@InName)}
            }
        }
    }
    {}
    {\LWR@patcherror{changes}{listofchanges}}

\xpatchcmd{\listofchanges}
    {\parbox{\Changes@Len@summ}}
    {}
    {}
    {\LWR@patcherror{changes}{listofchanges}}

\xpatchcmd{\listofchanges}
    {\parbox{\Changes@Len@summ}}
    {}
    {}
    {\LWR@patcherror{changes}{listofchanges}}

\xpatchcmd{\listofchanges}
    {\parbox{\Changes@Len@summ}}
    {}
    {}
    {\LWR@patcherror{changes}{listofchanges}}
```

File 61 **lwarp-chappg.sty**

§ 158 Package chappg

(*Emulates or patches code by* Robin Fairbairns.)

Pkg chappg chappg is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{chappg}[2006/05/09]

\renewcommand{\pagenumbering}[2][]{}
\providecommand{\chappgsep}{--}
```

File 62 **lwarp-chapterbib.sty**

§ 159 Package chapterbib

(*Emulates or patches code by* Donald Arseneau.)

Pkg chapterbib chapterbib is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{chapterbib}[2010/09/18]

\xdef\@savedjobname{\BaseJobname}
\let\@currentipfile\@savedjobname
```

File 63 **lwarp-chemfig.sty**

§ 160 Package chemfig

(*Emulates or patches code by* Christian Tellechea.)

Pkg chemfig 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-~\packagediagramname)]
\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:

```
\LWR@ProvidesPackagePass{chemfig}

\LetLtxMacro\LWR@chemfig@origchemfig\chemfig

\DeclareDocumentCommand\chemfig{s O{} O{} m}{%
    \begin{lateximage}[(-chemfig-~\packagediagramname)]%
    \IfBooleanTF{#1}{%
        \LWR@chemfig@origchemfig*[#2][#3]{#4}%
    }{%
        \LWR@chemfig@origchemfig[#2][#3]{#4}%
    }
    \end{lateximage}%
```

```
}

\LetLtxMacro\LWR@chemfig@origCF@lewis@b\CF@lewis@b

\def\CF@lewis@b#1#2{%
\begin{lateximage}[(-chemfig-~\packagediagramname)]%
\LWR@chemfig@origCF@lewis@b{#1}{#2}%
\end{lateximage}%
}

\preto{\schemestart}{\begin{lateximage}[(-chemfig-~\packagediagramname)]}
\appto{\CF@schemestop}{\end{lateximage}}

\LetLtxMacro\LWR@chemfig@origchemleft\chemleft

\def\chemleft#1#2\chemright#3{%
\begin{lateximage}[(-chemfig-~\packagediagramname)]%
\LWR@chemfig@origchemleft#1#2\chemright#3%
\end{lateximage}%
}

\LetLtxMacro\LWR@chemfig@origchemup\chemup

\def\chemup#1#2\chemdown#3{%
\begin{lateximage}[(-chemfig-~\packagediagramname)]%
\LWR@chemfig@origchemup#1#2\chemdown#3%
\end{lateximage}%
}
```

File 64 **lwarp-chemformula.sty**

§ 161 Package

chemformula

(*Emulates or patches code by* Clemens Niederberger.)

Pkg chemformula

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:

```
\LWR@ProvidesPackagePass{chemformula}[2017/03/23]

\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.

```
\RenewDocumentCommand \ch { O{}m }
 {%
```

To work inside `align` with `\displaymathother`, a simple version must be used to work with chemformula's adaptation to `align`.

```
   \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%  lwarp
   {
       \chemformula_ch:nn {#1} {#2}%   original
   }
```

If used as the outer level, must temporarily ensure MathJax is disabled:

```
   {
       \begingroup%
       \boolfalse{mathjax}%
```

An inline image is used, adjusted for the baseline:

```
       \LWR@subsingledollar*{% lwarp
           \textbackslash{}ch\{\LWR@HTMLsanitize{#2}\}% alt text
       }{%
        \protect\LWR@HTMLsanitize{\detokenize\expandafter{#1}}% add'l hashing
       }%
       {%
           \chemformula_ch:nn {#1} {#2}%   original
       }%
       \endgroup%
   }
}
```

\chcpd Similar to \ch.

```
\cs_gset_protected:Npn \chemformula_chcpd:nn #1#2
 {
   \begingroup%
   \boolfalse{mathjax}%
   \LWR@subsingledollar*{% lwarp
       \textbackslash{}chcpd\{\LWR@HTMLsanitize{#2}\}%
   }{%
       \protect\LWR@HTMLsanitize{\detokenize\expandafter{#1}}%
   }{% original
   \group_begin:
     \tl_if_blank:nF {#2}
       {
         \keys_set:nn {chemformula} {#1}
         \__chemformula_save_catcodes:
         \__chemformula_sanitize:Nn
           \l__chemformula_chemformula_tmpa_tl
           {#2}
         \__chemformula_input_compound_no_check:NV
           \l__chemformula_compound_tl
           \l__chemformula_chemformula_tmpa_tl
         \__chemformula_prepare_output:N \l__chemformula_compound_tl
```

```
          \chemformula_write:V \l__chemformula_compound_tl
        }
    \group_end:
    }
    \endgroup
}
```

\charrow If standalone, appears in a regular lateximage.

```
\RenewDocumentCommand \charrow { mO{}O{} }
{
    \begin{lateximage}[(-chemformula- charrow)]
    \group_begin:
      \__chemformula_draw_arrow:nnn {#1} {#2} {#3}
    \group_end:
    \end{lateximage}
}
```

\chname If standalone, appears in a regular lateximage, hashed according to contents.

```
\RenewDocumentCommand \chname { R(){}R(){} }
  {
    \begin{lateximage}*[%
        \textbackslash{}chname(\LWR@HTMLsanitize{#1})(\LWR@HTMLsanitize{#2})
    ]%
        \chemformula_chwritebelow:nn {#1} {#2}
    \end{lateximage}
  }
```

\chlewis Placed inline, hashed according to contents and options.

```
\RenewDocumentCommand \chlewis { O{}mm }
  {
    \begingroup%
    \boolfalse{mathjax}%
    \LWR@subsingledollar*{\textbackslash{}chlewis\{#2\}\{#3\}}%
    {
        \protect\LWR@HTMLsanitize{\detokenize\expandafter{#1}}%
    }{
        \chemformula_lewis:nnn {#1} {#2} {#3}
    }
    \endgroup%
  }
```

lwarp redefines the $ character, so special handling is required to escape math expressions inside \ch.

This boolean tracks a new kind of escaped math:

```
\bool_new:N      \l__chemformula_first_last_LWRdollar_bool
```

\chemformula_input_escape_math

Adds additional escaping for the new dollar definition:

```
\cs_gset_protected:Npn \__chemformula_input_escape_math:n #1
  {
    \__chemformula_first_last_math:n {#1}
    \bool_if:NT \l__chemformula_first_last_dollar_bool
      {
        \bool_set_true:N \l__chemformula_first_last_math_bool
        \__chemformula_read_escape_dollar:w #1 \q_nil
      }
    \bool_if:NT \l__chemformula_first_last_mathbraces_bool
      {
        \bool_set_true:N \l__chemformula_first_last_math_bool
        \__chemformula_read_escape_mathbraces:w #1 \q_nil
      }
```

Added by lwarp:

```
    \bool_if:NT \l__chemformula_first_last_LWRdollar_bool%      lwarp
      {
        \bool_set_true:N \l__chemformula_first_last_math_bool%  lwarp
        \__chemformula_read_escape_LWRdollar:w #1 \q_nil%       lwarp
      }
  }
```

\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.

```
\begingroup
\catcode`\$=\active

\cs_new_protected:Npn \__chemformula_read_escape_LWRdollar:w $#1$ \q_nil
{
    \__chemformula_read_escape_math:n {#1}
}

\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 $.

```
\cs_new_protected:Npn \__chemformula_bool_cs_set_if_first_last:NnNN #1#2#3#4
  {
    \int_zero:N \l__chemformula_tmpa_int
    \int_zero:N \l__chemformula_tmpb_int
    \int_set:Nn \l__chemformula_tmpa_int { \tl_count:n {#2} }
    \tl_map_inline:nn {#2}
      {
```

```
        \int_incr:N \l__chemformula_tmpb_int
        \int_compare:nT { \l__chemformula_tmpb_int = 1 }
          {
```

At the start, the cs_ version compares control sequences:

```
            \ifdefstrequal{##1}{#3}% lwarp
                {
                    \bool_set_true:N #1
                }% lwarp
                {}
          }
```

At the end, compare more control sequences:

```
        \int_compare:nT { \l__chemformula_tmpb_int = \l__chemformula_tmpa_int }
          {
            \ifdefstrequal{##1}{#4}
                {}
                {
                    \bool_set_false:N #1
                }
          }
      }
  }
```

\chemformula_first_last_math

Modified to check for the new meaning of $ at first/last:

```
\cs_gset_protected:Npn \__chemformula_first_last_math:n #1
  {
    \bool_set_false:N \l__chemformula_first_last_math_bool
    \bool_set_false:N \l__chemformula_first_last_dollar_bool
    \bool_set_false:N \l__chemformula_first_last_LWRdollar_bool%    lwarp
    \bool_set_false:N \l__chemformula_first_last_mathbraces_bool
    \__chemformula_bool_set_if_first_last:Nnnn
      \l__chemformula_first_last_dollar_bool
      {#1}
      { $ } { $ }
    \bool_if:NF \l__chemformula_first_last_dollar_bool
      {
        \__chemformula_bool_set_if_first_last:Nnnn
          \l__chemformula_first_last_mathbraces_bool
          {#1}
          { \( } { \) }
```

Added by lwarp:

```
          \bool_if:NF \l__chemformula_first_last_mathbraces_bool%   lwarp
            {
                \__chemformula_bool_cs_set_if_first_last:NnNN
                \l__chemformula_first_last_LWRdollar_bool
                {#1}
                { \LWR@newsingledollar } { \LWR@newsingledollar }
            }%  lwarp
      }
  }
```

```
158 \ExplSyntaxOff
```

File 65 **lwarp-chemgreek.sty**

§ 162 Package

chemgreek

(*Emulates or patches code by* Clemens Niederberger.)

Pkg chemgreek chemgreek is patched for use by lwarp.

Greek symbols package selection

⚠ To use text-mode symbols, use packages textalpha or textgreek. Using the other packages supported by chemgreek will result in math-mode greek characters, which will result in SVG images being used. These images will be hashed.

⚠ XeLaTeX, LuaLaTeX If using XeLaTeX or LuaLaTeX, select the fontspec mapping:

```
\selectchemgreekmapping{fontspec}
```

for HTML output:

```
1 \LWR@ProvidesPackagePass{chemgreek}[2016/02/10]

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 66 **lwarp-chemmacros.sty**

§ 163 Package

chemmacros

(*Emulates or patches code by* Clemens Niederberger.)

Pkg chemmacros chemmacros is patched for use by lwarp.

for HTML output:

```
1 \LWR@ProvidesPackagePass{chemmacros}[2017/08/28]
```

SVG file hashing assumes that the relevent options are constant for the entire document.

§ 163.1 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}
```

§ 163.2 Code

§ 163.3 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.

```
\ExplSyntaxOn

\newcommand{\@ifchemmacrosmoduleloaded}[1]{%
\@ifl@aded{\c__chemmacros_module_extension_tl}{\c__chemmacros_module_prefix_tl.#1}%
}

\ExplSyntaxOff
```

§ 163.4 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

```
\DeclareDocumentEnvironment{polymerdelims}{}
{\begin{lateximage}[(-chemmacros- polymer)]}
{\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 85.

```
\DeclareDocumentEnvironment{redoxreaction}{m m}
{\begin{lateximage}[(-chemmacros- redoxreaction)]}
{\end{lateximage}}

\ExplSyntaxOn
```

§ 163.5 Acid-base

```
\AtBeginDocument{
\@ifchemmacrosmoduleloaded{acid-base}{
\PackageInfo{lwarp}{Patching~chemmacros~module~acid-base}

\cs_gset_protected:Npn \chemmacros_p:n #1
  {
    \begingroup
    \boolfalse{mathjax}
    \LWR@subsingledollar*{
        \textbackslash{}p\{\LWR@HTMLsanitize{#1}\}
    }{
        chemmacrosp\protect\LWR@HTMLsanitize{\detokenize\expandafter{#1}}%
    }{
    \group_begin:
      \mbox
        {
          \chemmacros_p_style:n {p}
          \ensuremath {#1}
        }
    \group_end:
    }
    \endgroup
  }

\RenewDocumentCommand \pH  {} {
    \begingroup
    \boolfalse{mathjax}
    \LWR@subsingledollar*{\textbackslash{}pH}{chemmacros}{
        \chemmacros_p:n { \chemmacros_chemformula:n {H} }
    }
    \endgroup
}

\RenewDocumentCommand \pOH {} {
    \begingroup
    \boolfalse{mathjax}
    \LWR@subsingledollar*{\textbackslash{}pOH}{chemmacros}{
        \chemmacros_p:n { \chemmacros_chemformula:n {OH} }
    }
    \endgroup
}

\RenewDocumentCommand \pKa {O{}}
  {
    \begingroup
```

```
    \boolfalse{mathjax}
    \LWR@subsingledollar*{\textbackslash{}pKa{[}#1{]}}{chemmacros #1}{
        \chemmacros_p:n
        {
            \Ka \ifblank {#1} {}
            { {} \c_math_subscript_token { \chemmacros_bold:n {#1} } }
        }
    }
    \endgroup
  }

\RenewDocumentCommand \pKb {O{}}
  {
    \begingroup
    \boolfalse{mathjax}
    \LWR@subsingledollar*{\textbackslash{}pKb{[}#1{]}}{chemmacros #1}{
        \chemmacros_p:n
        {
            \Kb \ifblank {#1} {}
            { {} \c_math_subscript_token { \chemmacros_bold:n {#1} } }
        }
    }
    \endgroup
  }

\LetLtxMacro\LWR@chemmacros@origKa\Ka
\renewcommand*{\Ka}{%
    \begingroup
    \boolfalse{mathjax}
    \LWR@subsingledollar*{\textbackslash{}Ka}{chemmacros}{%
        \LWR@chemmacros@origKa%
    }%
    \endgroup
}

\LetLtxMacro\LWR@chemmacros@origKb\Kb
\renewcommand*{\Kb}{%
    \begingroup
    \boolfalse{mathjax}
    \LWR@subsingledollar*{\textbackslash{}Kb}{chemmacros}{%
        \LWR@chemmacros@origKb%
    }%
    \endgroup
}

\LetLtxMacro\LWR@chemmacros@origKw\Kw
\renewcommand*{\Kw}{%
    \begingroup
    \boolfalse{mathjax}
    \LWR@subsingledollar*{\textbackslash{}Kw}{chemmacros}{
        \LWR@chemmacros@origKw
    }
    \endgroup
}

```

```
}{}% \@ifchemmacrosmoduleloaded
}% AtBeginDocument
```

§ 163.6 Charges

```
\AtBeginDocument{
\@ifchemmacrosmoduleloaded{charges}{
\PackageInfo{lwarp}{Patching~chemmacros~module~charges}

\cs_gset_protected:Npn \fplus  {
    \begingroup
    \boolfalse{mathjax}
    \LWR@subsingledollar*{\textbackslash{}fplus}{chemmacros}
    { \LWR@origensuredmath{\chemformula_fplus:}  }
    \endgroup
}
\cs_gset_protected:Npn \fminus {
    \begingroup
    \boolfalse{mathjax}
    \LWR@subsingledollar*{\textbackslash{}fminus}{chemmacros}
    { \LWR@origensuredmath{\chemformula_fminus:} }
    \endgroup
}

}{}% \@ifchemmacrosmoduleloaded
}% AtBeginDocument
```

§ 163.7 Nomenclature

```
\AtBeginDocument{
\@ifchemmacrosmoduleloaded{nomenclature}{
\PackageInfo{lwarp}{Patching~chemmacros~module~nomenclature}

\cs_gset_protected:Npn \chemmacros_charge:n #1
  {
    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}
    {\chemmacros_chemformula:n { {}^{#1} }}
    {
        \ifmmode
            {\chemmacros_chemformula:n { {}^{#1} }}
        \else
            { \textsuperscript{\ensuremath{#1}} }
        \fi
    }
  }


\LetLtxMacro\LWR@chemmacros@origchemprime\chemprime

\protected\def\chemprime { \HTMLunicode{2032} }

\appto\LWR@restoreorigformatting{%
\LetLtxMacro\chemprime\LWR@chemmacros@origchemprime%
}
```

```
\ChemCompatibilityFrom{5.8}
\cs_gset_protected:Npn \__chemmacros_cip:n #1
  {
    \tl_set:Nn \l__chemmacros_tmpa_tl {#1}
    \int_step_inline:nnnn {0} {1} {9}
      {
        \tl_replace_all:Nnn \l__chemmacros_tmpa_tl
          {##1}
          { { \l__chemmacros_cip_number_tl ##1} }
      }
    {
        \l__chemmacros_cip_inner_tl
        \LWR@textcurrentcolor{\LWR@textcurrentfont{% lwarp
            \l__chemmacros_tmpa_tl
        }}% lwarp
    }
  }
\EndChemCompatibility

\RenewDocumentCommand \Sconf { O{S} } {
\begin{lateximage}[\textbackslash{}Sconf{[}#1{]}]
    \chemmacros_sconf:n {#1}
\end{lateximage}
}

\RenewDocumentCommand \Rconf { O{R} } {
\begin{lateximage}[\textbackslash{}Rconf{[}#1{]}]
    \chemmacros_rconf:n {#1}
\end{lateximage}
}

\cs_gset_protected:Npn \chemmacros_hapto:n #1
  {
    \begingroup
    \boolfalse{mathjax}
    \LWR@subsingledollar*{\textbackslash{}hapto\{#1\}}{chemmacros}{
        \chemmacros_coordination_symbol:nnnn
        { \l__chemmacros_coord_use_hyphen_bool }
        {
            \chemmacros_if_compatibility:nnTF {>} {5.7}
            { \c_true_bool }
            { \c_false_bool }
        }
        { \chemeta }
        {#1}
    }
    \endgroup
  }

\cs_gset_protected:Npn \chemmacros_dento:n #1
  {
    \begingroup
    \boolfalse{mathjax}
    \LWR@subsingledollar*{\textbackslash{}dento\{#1\}}{chemmacros}{
        \chemmacros_coordination_symbol:nnnn
        { \l__chemmacros_coord_use_hyphen_bool }
```

```
        {
            \chemmacros_if_compatibility:nnTF {>} {5.7}
            { \c_true_bool }
            { \c_false_bool }
        }
        { \chemkappa }
        {#1}
    }
    \endgroup
  }

\cs_gset_protected:Npn \chemmacros_bridge:n #1
  {
    \begingroup
    \boolfalse{mathjax}
    \LWR@subsingledollar*{\textbackslash{}bridge\{#1\}}{chemmacros}{
        \chemmacros_coordination_symbol:nnnn
        { \l__chemmacros_coord_use_hyphen_bool }
        { \l__chemmacros_bridge_super_bool }
        { \chemmu }
        {#1}
    }
    \endgroup
  }
}{}% \@ifchemmacrosmoduleloaded
}% AtBeginDocument
```

§ 163.8 Particles

```
\AtBeginDocument{
\@ifchemmacrosmoduleloaded{particles}{
\PackageInfo{lwarp}{Patching~chemmacros~module~particles}

\cs_gset_protected:Npn \chemmacros_declare_nucleophile:Nn #1#2
  {
    \cs_set_protected:cpn {__chemmacros_ \chemmacros_remove_backslash:N #1:}
      {
        \bool_if:NTF \l__chemmacros_nucleophile_elpair_bool
          {
            \chemmacros_elpair:n { #2 }
            \chemmacros_if_compatibility:nnT {>=} {5.3}
              { \skip_horizontal:N \l__chemmacros_nucleophile_dim }
            \chemmacros_chemformula:n { {}^{-} }
          }
          { \chemmacros_chemformula:n { #2^{-} } }
      }
    \DeclareDocumentCommand #1 {o}
      {%
        \begin{lateximage}%
        \group_begin:%
          \IfNoValueF {##1}%
            { \chemmacros_set_keys:nn {particles} {##1} }%
          \use:c {__chemmacros_ \chemmacros_remove_backslash:N #1:}%
        \group_end:%
        \end{lateximage}%
```

```
      }
  }

\RenewChemNucleophile \Nuc {Nu}
\RenewChemNucleophile \ba  {ba}

}{}% \@ifchemmacrosmoduleloaded
}% AtBeginDocument
```

§ 163.9 Phases

```
\AtBeginDocument{
\@ifchemmacrosmoduleloaded{phases}{
\PackageInfo{lwarp}{Patching~chemmacros~module~phases}

\cs_undefine:N \chemmacros_phase:n
\cs_new_protected:Npn \chemmacros_phase:n #1
  {
    \chemmacros_leave_vmode:
    \bool_if:NTF \l__chemmacros_phases_sub_bool
      {
        \ifnumequal{\value{LWR@lateximagedepth}}{0}
        {
            \textsubscript{ (#1) }
        }
        {
            \chemformula_subscript:n { (#1) }
        }
      }
      {
        \skip_horizontal:N \l__chemmacros_phases_space_dim
        \chemmacros_text:n { (#1) }
      }
  }

}{}% \@ifchemmacrosmoduleloaded
}% AtBeginDocument
```

§ 163.10 Mechanisms

```
\AtBeginDocument{
\@ifchemmacrosmoduleloaded{mechanisms}{
\PackageInfo{lwarp}{Patching~chemmacros~module~mechanisms}

\chemmacros_define_keys:nn {textmechanisms}
  {
    type      .choice: ,
    type /    .code:n    =
      {
        \__chemmacros_set_mechanisms:nnn { S }
          {
              \textsubscript{N}
          }
          { }
      } ,
```

```
    type / 1  .code:n    =
      {
        \__chemmacros_set_mechanisms:nnn { S }
          {
              \textsubscript{N}
            1
          }
          { }
      } ,
    type / 2  .code:n    =
      {
        \__chemmacros_set_mechanisms:nnn { S }
          {
              \textsubscript{N}
            2
          }
          { }
      } ,
    type / se .code:n    =
      {
        \__chemmacros_set_mechanisms:nnn { S }
          {
              \textsubscript{E}
          }
          { }
      } ,
    type / 1e .code:n    =
      {
        \__chemmacros_set_mechanisms:nnn { S }
          {
              \textsubscript{E}
            1
          }
          { }
      } ,
    type / 2e .code:n    =
      {
        \__chemmacros_set_mechanisms:nnn { S }
          {
              \textsubscript{E}
            2
          }
          { }
      } ,
    type / ar .code:n    =
      {
        \__chemmacros_set_mechanisms:nnn { S }
          {
              \textsubscript{E}
          }
          { Ar - }
      } ,
    type / e  .code:n    =
      { \__chemmacros_set_mechanisms:nnn { E } { } { } } ,
    type / e1 .code:n    =
```

```
      { \__chemmacros_set_mechanisms:nnn { E } { 1 } { } } ,
    type / e2 .code:n    =
      { \__chemmacros_set_mechanisms:nnn { E } { 2 } { } } ,
    type / cb .code:n    =
      {
        \__chemmacros_set_mechanisms:nnn { E }
          {
            1
              \textsubscript{cb}
          }
          { }
      } ,
    type      .default:n =
  }

\cs_gset_protected:Npn \chemmacros_mechanisms:n #1
  {
    \tl_if_blank:nTF {#1}
      { \chemmacros_set_keys:nn {textmechanisms} { type } }
      { \chemmacros_set_keys:nn {textmechanisms} { type = #1 } }
    \mbox
      {
        \tl_use:N \l__chemmacros_mechanisms_ar_tl
        \tl_use:N \l__chemmacros_mechanisms_type_tl
        \tl_use:N \l__chemmacros_mechanisms_mol_tl
      }
  }

\appto\LWR@restoreorigformatting{%
\cs_set_protected:Npn \chemmacros_mechanisms:n #1%
  {%
    \tl_if_blank:nTF {#1}%
      { \chemmacros_set_keys:nn {mechanisms} { type } }%
      { \chemmacros_set_keys:nn {mechanisms} { type = #1 } }%
    \mbox%
      {%
        \tl_use:N \l__chemmacros_mechanisms_ar_tl%
        \tl_use:N \l__chemmacros_mechanisms_type_tl%
        \tl_use:N \l__chemmacros_mechanisms_mol_tl%
      }%
  }%
}

}{}% \@ifchemmacrosmoduleloaded
}% AtBeginDocument
```

§ 163.11 Newman

```
\AtBeginDocument{
\@ifchemmacrosmoduleloaded{newman}{
\PackageInfo{lwarp}{Patching~chemmacros~module~newman}

\RenewDocumentCommand \newman {od()m}%
  {
    \IfValueTF{#2}
```

```
    {\begin{lateximage}[\textbackslash{}newman(#2)\{#3\}]}
    {\begin{lateximage}[\textbackslash{}newman\{#3\}]}
    \group_begin:
      \IfNoValueF  {#1} { \chemmacros_set_keys:nn {newman} {#1} }
      \IfNoValueTF {#2}
        { \chemmacros_newman:nn {  } {#3} }
        { \chemmacros_newman:nn {#2} {#3} }
    \group_end:
    \end{lateximage}
  }%

}{}% \@ifchemmacrosmoduleloaded
}% AtBeginDocument
```

§ 163.12 Orbital

```
\AtBeginDocument{
\@ifchemmacrosmoduleloaded{orbital}{
\PackageInfo{lwarp}{Patching~chemmacros~module~orbital}

\RenewDocumentCommand \orbital {om}
  {
    \IfValueTF{#1}
    {
        \begin{lateximage}[%
            \textbackslash{}orbital{[}\LWR@HTMLsanitize{#1}{]}\{#2\}%
        ][][margin-left: 1em ; margin-right: 1em]
    }
    {
        \begin{lateximage}[%
            \textbackslash{}orbital\{#2\}%
        ][][margin-left: 1em ; margin-right: 1em]
    }
    \group_begin:
      \chemmacros_set_keys:nn {orbital/type} {#2}
      \IfNoValueTF {#1}
        { \chemmacros_orbital:n {  } }
        { \chemmacros_orbital:n {#1} }
    \group_end:
    \end{lateximage}
  }

}{}% \@ifchemmacrosmoduleloaded
}% AtBeginDocument
```

§ 163.13 Reactions

\chemmacros_declare_reaction_env {⟨*chem*⟩} {⟨*math*⟩} {⟨*args number*⟩} {⟨*argument list ({#2}{#3}...)*⟩}

```
\AtBeginDocument{
\@ifchemmacrosmoduleloaded{reactions}{
\PackageInfo{lwarp}{Patching~chemmacros~module~reactions}

\cs_gset_protected:Npn \chemmacros_declare_reaction_env:nnnn #1#2#3#4
  {
```

```
  \exp_args:Nnx \DeclareDocumentEnvironment {#1} { O{} \prg_replicate:nn {#3+0} {m} }
     {
       \boolfalse{mathjax}%    lwarp
       \chemmacros_add_reaction_description:n {##1}
       \__chemmacros_begin_reaction:
       \chemmacros_reaction_read:nnw {#2} {#4}
     }
     {
       \__chemmacros_end_reaction:
     }
  }
\cs_generate_variant:Nn \chemmacros_declare_reaction_env:nnnn {nnnV}

\RenewChemReaction {reaction}   {equation}
\RenewChemReaction {reaction*}  {equation*}
\RenewChemReaction {reactions}  {align}
\RenewChemReaction {reactions*} {align*}

}{}% \@ifchemmacrosmoduleloaded
}% AtBeginDocument
```

§ 163.14 Redox

```
\AtBeginDocument{
\@ifchemmacrosmoduleloaded{redox}{
\PackageInfo{lwarp}{Patching~chemmacros~module~redox}

\NewDocumentCommand \LWR@chemmacros@ox { s m >{\SplitArgument{1}{,}}m }
  {
    \IfBooleanTF {#1}
      { \chemmacros_ox:nnnn {#1} {#2} #3 }
      { \chemmacros_ox:nnnn {  } {#2} #3 }
  }

\RenewDocumentCommand \ox { s O{} m }
  {
    \begingroup
    \boolfalse{mathjax}
    \IfBooleanTF {#1}
      {
        \LWR@subsingledollar*{% yes hash
            \textbackslash{}ox*\{\LWR@HTMLsanitize{#3}\}% alt
        }{%
            star \protect\LWR@HTMLsanitize{\detokenize\expandafter{#2}}%
        }{%
            \LWR@chemmacros@ox* {#2} {#3}% contents
        }%
      }
      {
        \LWR@subsingledollar*{% yes hash
            \textbackslash{}ox*\{\LWR@HTMLsanitize{#3}\}% alt
        }{%
            \protect\LWR@HTMLsanitize{\detokenize\expandafter{#2}}%
        }{%
            \LWR@chemmacros@ox {#2} {#3}% contents
```

```
        }%
      }
    \endgroup
  }

}{}% \@ifchemmacrosmoduleloaded
}% AtBeginDocument
```

§ 163.15 Scheme

Fix for chemmacros as of v5.8b, when using newfloat and babel:

```
\AtBeginDocument{
\@ifchemmacrosmoduleloaded{scheme}{
\PackageInfo{lwarp}{Patching~chemmacros~module~scheme}

\ifdefstring{\schemename}{los}{
\SetupFloatingEnvironment{scheme}{
name = \chemmacros_translate:n {scheme-name}
}
}{}

}{}% \@ifchemmacrosmoduleloaded
}% AtBeginDocument
```

§ 163.16 Spectroscopy

```
\AtBeginDocument{
\@ifchemmacrosmoduleloaded{spectroscopy}{
\PackageInfo{lwarp}{Patching~chemmacros~module~spectroscopy}

\ChemCompatibilityTo{5.8}
\cs_gset_protected:Npn \__chemmacros_nmr_base:nn #1#2
  {
    \tl_if_blank:VF \g__chemmacros_nmr_element_coupled_tl
      {
        \tl_put_left:Nn \g__chemmacros_nmr_element_coupled_tl { \{ }
        \tl_put_right:Nn \g__chemmacros_nmr_element_coupled_tl { \} }
      }
    \tl_put_left:Nn \g__chemmacros_nmr_element_coupled_tl {#2}
%     \chemmacros_chemformula:n { ^{#1} }
    \textsuperscript{#1}
    \bool_if:NTF \l__chemmacros_nmr_parse_bool
      { \chemformula_ch:nV {} \g__chemmacros_nmr_element_coupled_tl }
      { \chemmacros_chemformula:V \g__chemmacros_nmr_element_coupled_tl }
    \tl_use:N \l__chemmacros_nmr_element_method_connector_tl
    \tl_use:N \l__chemmacros_nmr_method_tl
  }
\EndChemCompatibility
\ChemCompatibilityFrom{5.8}
\cs_gset_protected:Npn \__chemmacros_nmr_base:nn #1#2
  {
    \group_begin:
      \tl_use:N \l__chemmacros_nmr_base_format_tl
      \tl_if_blank:VF \g__chemmacros_nmr_element_coupled_tl
```

```
572         {
573           \tl_put_left:Nn \g__chemmacros_nmr_element_coupled_tl { \{ }
574           \tl_put_right:Nn \g__chemmacros_nmr_element_coupled_tl { \} }
575         }
576       \tl_put_left:Nn \g__chemmacros_nmr_element_coupled_tl {#2}
577 %       \chemmacros_chemformula:n { ^{#1} }
578       \textsuperscript{#1}
579       \tl_if_blank:VF \g__chemmacros_nmr_element_coupled_tl
580         {
581           \bool_if:NTF \l__chemmacros_nmr_parse_bool
582             { \chemformula_ch:nV {} \g__chemmacros_nmr_element_coupled_tl }
583             { \chemmacros_chemformula:V \g__chemmacros_nmr_element_coupled_tl }
584         }
585       \tl_use:N \l__chemmacros_nmr_element_method_connector_tl
586       \tl_use:N \l__chemmacros_nmr_method_tl
587     \group_end:
588   }
589 \EndChemCompatibility
590
591
592 \cs_gset_protected:Npn \chemmacros_nmr_position:n #1
593   {
594     \chemmacros_chemformula:x
595       {
596         \exp_not:V \g__chemmacros_nmr_element_tl
597         \bool_if:NF \l__chemmacros_nmr_position_side_bool
598           {
599             \tl_if_eq:NnTF \l__chemmacros_nmr_position_tl {^}% lwarp
600             { \textsuperscript{\exp_not:n { {#1} }} }% lwarp
601             { \textsubscript{\exp_not:n { {#1} }} }% lwarp
602 %             \exp_not:V \l__chemmacros_nmr_position_tl
603 %             \exp_not:n { {#1} }
604           }
605       }
606     \bool_if:NT \l__chemmacros_nmr_position_side_bool
607       {
608         \tl_use:N \l__chemmacros_nmr_position_tl
609         \__chemmacros_nmr_position:n {#1}
610       }
611   }
612
613 \cs_gset_protected:Npn \__chemmacros_nmr_coupling:w (#1;#2)
614   {
615     \tl_set:Nn \l__chemmacros_nmr_coupling_bonds_tl
616       {
617         \l__chemmacros_nmr_coupling_bonds_pre_tl
618         #1
619         \l__chemmacros_nmr_coupling_bonds_post_tl
620       }
621     \bool_if:NTF \l__chemmacros_nmr_coupling_nuclei_sub_bool
622       {
623         \tl_set:Nn \l__chemmacros_nmr_coupling_nuclei_tl
624           {
625 %             \c_math_subscript_token
626             \textsubscript% lwarp
```

```
              {
                \l__chemmacros_nmr_coupling_nuclei_pre_tl
                \chemmacros_chemformula:n {#2}
                \l__chemmacros_nmr_coupling_nuclei_post_tl
              }
          }
      }
      {
        \tl_set:Nn \l__chemmacros_nmr_coupling_nuclei_tl
          {
            \l__chemmacros_nmr_coupling_nuclei_pre_tl
            \chemmacros_chemformula:n {#2}
            \l__chemmacros_nmr_coupling_nuclei_post_tl
          }
      }
    \__chemmacros_nmr_coupling_aux_i:w
  }

\AfterEndPreamble{% After \AtBeginDocument
% \NMR{<num>,<elem>}(<num>,<unit>)[<solvent>] ALL arguments are optional
% \NMR* same but without ": $\delta$" at end
\cs_gset_protected:Npn \chemmacros_nmr:nnnn #1#2#3#4
    {
    \bool_if:NT \l__chemmacros_nmr_list_bool { \item \scan_stop: }
    \group_begin:
        \chemmacros_leave_vmode:
        \bool_set_false:N \l__chemmacros_nmr_frequency_bool
        \bool_set_false:N \l__chemmacros_nmr_solvent_bool
        \tl_if_empty:nF {#3}
        { \bool_set_true:N \l__chemmacros_nmr_frequency_bool }
        \tl_if_empty:nF {#4}
        { \bool_set_true:N \l__chemmacros_nmr_solvent_bool }
        \bool_if:nT
        {
            \l__chemmacros_nmr_frequency_bool
            ||
            \l__chemmacros_nmr_solvent_bool
        }
        { \bool_set_true:N \l__chemmacros_nmr_delimiters_bool }
        \bool_if:nT
        {
            \l__chemmacros_nmr_frequency_bool
            &&
            \l__chemmacros_nmr_solvent_bool
        }
        { \bool_set_true:N \l__chemmacros_nmr_comma_bool }
        \tl_if_empty:nTF {#2}
        {
            \__chemmacros_nmr_nucleus:VV
            \l__chemmacros_nmr_isotope_default_tl
            \l__chemmacros_nmr_element_default_tl
        }
        { \__chemmacros_nmr_nucleus:w #2 \q_stop }
        \mode_if_math:TF
        {
```

```
            \text
            {
                \group_begin:
                \tl_use:N \l__chemmacros_nmr_format_tl
\LWR@textcurrentcolor{\LWR@textcurrentfont{% lwarp
                \__chemmacros_nmr_base:VV
                    \g__chemmacros_nmr_isotope_tl
                    \g__chemmacros_nmr_element_tl
                \bool_if:NT \l__chemmacros_nmr_delimiters_bool
                    { ~ ( }
                \bool_if:NT \l__chemmacros_nmr_frequency_bool
                    { \__chemmacros_nmr_frequency:n {#3} }
                \bool_if:NT \l__chemmacros_nmr_comma_bool
                    { , ~ }
                \bool_if:NT \l__chemmacros_nmr_solvent_bool
                    { \chemmacros_chemformula:n {#4} }
                \bool_if:NT \l__chemmacros_nmr_delimiters_bool
                    { ) }
                \tl_if_blank:nT {#1} {:~}
}}% lwarp
                \group_end:
            }
            \tl_if_blank:nT {#1}
            {
                \delta
                \text { \l__chemmacros_nmr_delta_tl }
                \bool_if:NT \l__chemmacros_nmr_use_equal_bool {=}
            }
        }
        {
            \group_begin:
            \tl_use:N \l__chemmacros_nmr_format_tl
\LWR@textcurrentcolor{\LWR@textcurrentfont{% lwarp
            \__chemmacros_nmr_base:VV
                \g__chemmacros_nmr_isotope_tl
                \g__chemmacros_nmr_element_tl
            \bool_if:NT \l__chemmacros_nmr_delimiters_bool
                {~(}
            \bool_if:NT \l__chemmacros_nmr_frequency_bool
                { \__chemmacros_nmr_frequency:n {#3} }
            \bool_if:NT \l__chemmacros_nmr_comma_bool
                {,~}
            \bool_if:NT \l__chemmacros_nmr_solvent_bool
                {
                \bool_if:NTF \l__chemmacros_nmr_parse_bool

%                    { \chemformula_ch:nn { } {#4} }%  original
                    {\ch{#4}}%  lwarp
                    {#4}
                }
            \bool_if:NT \l__chemmacros_nmr_delimiters_bool
                {)}
}}% lwarp
            \tl_if_blank:nT {#1} {:}
            \group_end:
```

```
            \tl_if_blank:nT {#1}
            {
                \tl_use:N \c_space_tl
                \c_math_toggle_token
                \delta
                \c_math_toggle_token
                \l__chemmacros_nmr_delta_tl
                \bool_if:NT \l__chemmacros_nmr_use_equal_bool {~=}
            }
        }
    \group_end:
    }
}% AfterEndPremble


\RenewDocumentCommand \chemmacros_data:w { smo }
  {
    \bool_if:NT \l__chemmacros_nmr_list_bool { \item }
      {
%         \tl_use:N \l__chemmacros_nmr_format_tl #2
        \tl_use:N \l__chemmacros_nmr_format_tl
        \LWR@textcurrentcolor{\LWR@textcurrentfont{% lwarp
            #2
            \IfNoValueF {#3} { ~ ( #3 ) }
         \IfBooleanT {#1} { \bool_if:NT \l__chemmacros_nmr_use_equal_bool { : } }
        }}% lwarp
      }
    \IfBooleanF {#1} { \bool_if:NT \l__chemmacros_nmr_use_equal_bool { ~ = } }
  }

}{}% \@ifchemmacrosmoduleloaded
}% AtBeginDocument
```

§ 163.17 Thermodynamics

```
\AtBeginDocument{
\@ifchemmacrosmoduleloaded{thermodynamics}{
\PackageInfo{lwarp}{Patching~chemmacros~module~thermodynamics}

\cs_gset_protected:Npn \chemmacros_state:nn #1#2
  {
    \group_begin:
      \boolfalse{mathjax}
      \chemmacros_set_keys:nn {thermodynamics} {#1}
        \LWR@subsingledollar*{% yes hashing
            \textbackslash{}state\{\LWR@HTMLsanitize{#2}\}% alt
        }{%
            chemmacros_state% add'l hashing
            #1% options
            LSP \tl_use:N \l__chemmacros_state_sp_left_tl% super/subscripts
            LSB \tl_use:N \l__chemmacros_state_sb_left_tl
            RSP \tl_use:N \l__chemmacros_state_sp_right_tl
            RSB \tl_use:N \l__chemmacros_state_sb_right_tl
        }
        {
```

```
        \LWR@origensuredmath{
         \chemmacros_text:V \l__chemmacros_state_pre_tl
         \c_math_superscript_token
           { \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*.

```
         \tl_if_empty:NTF \l__chemmacros_state_sb_left_tl
         {}
         {
           \c_math_subscript_token
           { \chemmacros_text:V \l__chemmacros_state_sb_left_tl }
         }
         #2
         \c_math_superscript_token
           { \chemmacros_text:V \l__chemmacros_state_sp_right_tl }
         \tl_if_empty:NTF \l__chemmacros_state_sb_right_tl
         {}
         {
           \c_math_subscript_token
           { \chemmacros_text:V \l__chemmacros_state_sb_right_tl }
         }
         \chemmacros_text:V \l__chemmacros_state_post_tl
        }
       }
    \group_end:
  }
\cs_generate_variant:Nn \chemmacros_state:nn { nV }

\cs_gset_protected:Npn \chemmacros_declare_state:Nn #1#2
  {
    \chemmacros_define_keys:xn
      {thermodynamics/\chemmacros_remove_backslash:N #1}
      {
        pre              .meta:nn = {chemmacros/thermodynamics} { pre = ##1 } ,
        post             .meta:nn = {chemmacros/thermodynamics} { post = ##1 } ,
      superscript-left  .meta:nn = {chemmacros/thermodynamics} { superscript-left = ##1 } ,
      superscript-right .meta:nn = {chemmacros/thermodynamics} { superscript-right = ##1 } ,
        superscript       .meta:n  = { superscript-right = ##1 } ,
      subscript-left   .meta:nn = {chemmacros/thermodynamics} { subscript-left = ##1 } ,
      subscript-right  .meta:nn = {chemmacros/thermodynamics} { subscript-right = ##1 } ,
        subscript         .meta:n     = { subscript-left = ##1 } ,
        subscript-pos     .choices:nn =
          { left , right }
          { \tl_set_eq:NN \l__chemmacros_state_sb_pos_tl \l_keys_choice_tl } ,
        symbol            .tl_set:N = \l__chemmacros_state_symbol_tl ,
        unit              .tl_set:N = \l__chemmacros_state_unit_tl
      }
    \DeclareDocumentCommand #1 { sO{}D(){}m }
      {
        \group_begin:
          \chemmacros_set_keys:xn
            {thermodynamics/\chemmacros_remove_backslash:N #1}
            {#2}
```

```
          \tl_if_blank:nF {##3}
            {
              \chemmacros_set_keys:nx {thermodynamics}
               { subscript-\l__chemmacros_state_sb_pos_tl = \exp_not:n {##3} }
            }
              \chemmacros_state:nV {##2} \l__chemmacros_state_symbol_tl
             \chemmacros_set_keys_groups:nnn {thermodynamics} {variables} {##2}
           \IfBooleanF {##1} {  = ~ \SI {##4} { \l__chemmacros_state_unit_tl } }
        \group_end:
      }
  }
```

The pre-existing macros are redefined with the new definition:

```
\RenewChemState \enthalpy { symbol = H , unit = \kilo\joule\per\mole }
\RenewChemState \entropy  { symbol = S , unit = \joule\per\kelvin\per\mole , pre = }
\RenewChemState \gibbs    { symbol = G , unit = \kilo\joule\per\mole }

}{}% \@ifchemmacrosmoduleloaded
}% AtBeginDocument

\ExplSyntaxOff
```

File 67 **lwarp-chemnum.sty**

§ 164 Package

chemnum

(*Emulates or patches code by* Clemens Niederberger.)

Pkg chemnum chemnum is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{chemnum}[2016/04/14]

\ExplSyntaxOn

\cs_gset_protected:Npn \chemnum_compound_write:n #1
  {
    \chemnum_get_compound_property:nn {#1} {pre-main-label-code}
    \group_begin:
      \bool_if:NTF \l__chemnum_compound_local_bool
        { \l__chemnum_local_label_format_tl }
        { \chemnum_get_compound_property:nn {#1} {label-format} }
      {
        \LWR@textcurrentfont{
            \chemnum_get_compound_property:nn {#1} {counter-representation}
        }
      }
    \group_end:
    \chemnum_get_compound_property:nn {#1} {post-main-label-code}
  }

\cs_gset_protected:Npn \chemnum_subcompound_write:nn #1#2
  {
```

```
    \group_begin:
      \bool_if:NTF \l__chemnum_compound_local_bool
        { \l__chemnum_local_label_format_tl }
        { \chemnum_get_compound_property:nn {#1} {label-format} }
      {
        \LWR@textcurrentfont{
            \chemnum_get_subcompound_property:nnn {#1} {#2}
            {counter-representation}
        }
      }
    \group_end:
  }

\ExplSyntaxOff
```

File 68 **lwarp-chkfloat.sty**

§ 165 Package chkfloat

Pkg chkfloat chkfloat is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{chkfloat}[2012/08/19]
```

File 69 **lwarp-chngpage.sty**

§ 166 Package chngpage

(*Emulates or patches code by* Peter Wilson.)

Pkg chngpage chngpage is emulated.

for HTML output: Discard all options for lwarp-chngpage:

```
\LWR@ProvidesPackageDrop{chngpage}[2009/10/20]
\LWR@origRequirePackage{lwarp-changepage}
```

File 70 **lwarp-cite.sty**

§ 167 Package cite

(*Emulates or patches code by* Donald Arseneau.)

Pkg cite cite is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{cite}[2015/02/27]
```

For the [super] option, the \kern must be removed:

```
\def\LWRCT@biblabel#1{\@citess{#1}\kern-\labelsep\,}

\ifdefstrequal{\@biblabel}{\LWRCT@biblabel}
{
    \def\@biblabel#1{\@citess{#1}}
}{}
```

For the [super] option, `\textsuperscript` is used instead of math superscript:

```
\def\@citess#1{\textsuperscript{#1}}

\DeclareDocumentCommand\citepunct{}{,\,\relax}
```

File 71 **lwarp-CJK.sty**

§ 168 Package CJK

Pkg CJK CJK does not work with lwarp unless called from ctex.

for HTML output:

```
\@ifpackageloaded{xeCJK}{}{
    \LWR@loadnever{CJK}{ctex, xeCJK}
}

\LWR@ProvidesPackagePass{CJK}[2015/04/18]
```

File 72 **lwarp-CJKutf8.sty**

§ 169 Package CJKutf8

Pkg CJKutf8 CJKutf8 does not work with lwarp unless called from ctex.

for HTML output:

```
\@ifpackageloaded{xeCJK}{}{
    \LWR@loadnever{CJKutf8}{ctex, xeCJK}
}

\LWR@ProvidesPackagePass{CJKutf8}[2015/04/18]
```

File 73 **lwarp-clrdblpg.sty**

§ 170 Package clrdblpg

Pkg clrdblpg clrdblpg is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{clrdblpg}[2018/04/21]
```

File 74 **lwarp-cmdtrack.sty**

§ 171 Package cmdtrack

Pkg cmdtrack cmdtrack is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{cmdtrack}[2012/12/18]
\newcommand{\untrack}[1]{}
```

File 75 **lwarp-color.sty**

§ 172 Package color

Pkg color 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:

```
\LWR@ProvidesPackageDrop{color}[2016/07/10]
\RequirePackage{xcolor}
```

File 76 **lwarp-colortbl.sty**

§ 173 Package colortbl

Pkg colortbl 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:

```
\let\rowcolor\relax

\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.

```
\NewDocumentCommand{\LWR@HTML@columncolor}{O{named} m o o}{%
\convertcolorspec{#1}{#2}{HTML}\LWR@columnHTMLcolor%
\LWR@addtabularcellcolor%
}

\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*⟩]

```
\NewDocumentCommand{\LWR@HTML@rowcolor}{O{named} m o o}{%
\convertcolorspec{#1}{#2}{HTML}\LWR@rowHTMLcolor%
\LWR@getmynexttoken%
}

\AtBeginDocument{\LWR@expandableformatted{rowcolor}}
```

`\cellcolor` [⟨*model*⟩] {⟨*color*⟩} [⟨*left overhang*⟩] [⟨*right overhang*⟩]

```
\NewDocumentCommand{\LWR@HTML@cellcolor}{O{named} m o o}{%
\convertcolorspec{#1}{#2}{HTML}\LWR@cellHTMLcolor%
\LWR@addtabularcellcolor%
}

\AtBeginDocument{\LWR@formatted{cellcolor}}
```

`\arrayrulecolor` [⟨*model*⟩] {⟨*color*⟩}

The HTML version for use outside a `tabular`. Inside a `tabular`, `\LWR@HTML@arrayrulecolornexttoken` is used instead.

```
\newcommand{\LWR@HTML@arrayrulecolor}[2][named]{%
\convertcolorspec{#1}{#2}{HTML}\LWR@ruleHTMLcolor%
}

\AtBeginDocument{\LWR@expandableformatted{arrayrulecolor}}
```

[⟨*model*⟩] {⟨*color*⟩}

`\LWR@arrayrulecolornexttoken` The HTML version for use inside a `tabular`.

```
\newcommand{\LWR@HTML@arrayrulecolornexttoken}[2][named]{%
\convertcolorspec{#1}{#2}{HTML}\LWR@ruleHTMLcolor%
\LWR@getmynexttoken%
}

\AtBeginDocument{\LWR@expandableformatted{arrayrulecolornexttoken}}
```

`\doublerulesepcolor` [⟨*model*⟩] {⟨*color*⟩}

The version for use outside a `tabular`.

```
\newcommand{\LWR@HTML@doublerulesepcolor}[2][named]{}
```

```
34
35 \AtBeginDocument{\LWR@expandableformatted{doublerulesepcolor}}
```

[⟨*model*⟩] {⟨*color*⟩}

\LWR@doublerulesepcolornexttoken The version for use inside a tabular.

```
36 \newcommand{\LWR@HTML@doublerulesepcolornexttoken}[2][named]{\LWR@getmynexttoken}
37
38 \AtBeginDocument{\LWR@expandableformatted{doublerulesepcolornexttoken}}
```

File 77 **lwarp-continue.sty**

§ 174 Package

continue

Pkg continue 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 78 **lwarp-copyrightbox.sty**

§ 175 Package

copyrightbox

(*Emulates or patches code by* Thomas Fischer, Ives van der Flaas.)

Pkg copyrightbox copyrightbox is emulated for use by lwarp.

The entire copyright box is placed inside a of class copyrightbox.

The contents are placed inside a of class copyrightboxcontents.

The copyright notice is placed inside a 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
```

```
\end{BlockClass}
\begin{BlockClass}{copyrightboxnote}
#3
\end{BlockClass}
\end{BlockClass}
}

\newcommand{\CRB@setcopyrightfont}{}
\newcommand{\CRB@setcopyrightparagraphstyle}{}
```

File 79 **lwarp-crop.sty**

§ 176 Package crop

(*Emulates or patches code by* Melchior FRANZ.)

Pkg crop Emulated.

for HTML output: Discard all options for lwarp-crop:

```
\LWR@ProvidesPackageDrop{crop}[2003/05/20]

\newcommand*{\crop}[1][]{}
\newcommand*{\cropdef}[6][]{}
```

File 80 **lwarp-cuted.sty**

§ 177 Package cuted

(*Emulates or patches code by* Sigitas Tolušis.)

Pkg cuted cuted is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{cuted}[2012/10/04]

\newenvironment{strip}{}{}
\newskip\stripsep
\def\oldcolsbreak#1{}
```

File 81 **lwarp-cutwin.sty**

§ 178 Package cutwin

(*Emulates or patches code by* Peter Wilson and Alan Hoenig.)

Pkg cutwin Emulated.

for HTML output: Discard all options for lwarp-cutwin:

```
\LWR@ProvidesPackageDrop{cutwin}[2010/09/29]

\newcommand*{\opencutleft}{}
\newcommand*{\opencutright}{}
\newcommand*{\opencutcenter}{}
\newcommand*{\cutfuzz}{}

\newenvironment{cutout}[4]
{\marginpar{\windowpagestuff}}
{}

\newcommand*{\windowpagestuff}{}

\newcommand*{\pageinwindow}{%
% \begin{minipage}{.3\linewidth}
\windowpagestuff
% \end{minipage}
}

\newenvironment{shapedcutout}[3]
{\marginpar{\picinwindow}}
{}

\newcommand*{\putstuffinpic}{}

\newcommand*{\picinwindow}{%
\begin{picture}(0,0)
\putstuffinpic
\end{picture}}
```

File 82 **lwarp-dblfloatfix.sty**

§ 179 Package dblfloatfix

Pkg dblfloatfix dblfloatfix is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{dblfloatfix}[2012/12/31]
```

File 83 **lwarp-dblfnote.sty**

§ 180 Package dblfnote

(*Emulates or patches code by* Hiroshi Nakashima.)

Pkg dblfnote dblfnote is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{dblfnote}[1999/07/14]
```

```
\newcounter{DFNsloppiness}
\newdimen\DFNcolumnsep
\newdimen\DFNcolumnwidth
\def\DFNallowcbreak{}
\def\DFNinhibitcbreak{}
\def\DFNtrysingle{}
\def\DFNalwaysdouble{}
\def\DFNruleboth{}
\def\DFNruleleft{}
```

File 84 **lwarp-dcolumn.sty**

§ 181 Package dcolumn

Pkg dcolumn dcolumn is emulated by the lwarp core.

```
\LWR@ProvidesPackageDrop{dcolumn}[2014/10/28]
```

File 85 **lwarp-diagbox.sty**

§ 182 Package diagbox

(*Emulates or patches code by* Leo Liu.)

Pkg diagbox diagbox is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{diagbox}[2016/12/28]
```

To restore print-mode inside a `lateximage`:

```
\LetLtxMacro\LWR@origdiagbox@double\diagbox@double
\LetLtxMacro\LWR@origdiagbox@triple\diagbox@triple

\appto\LWR@restoreorigformatting{%
\LetLtxMacro\diagbox@double\LWR@origdiagbox@double%
\LetLtxMacro\diagbox@triple\LWR@origdiagbox@triple%
}
```

\LWR@diagbox@AB {⟨*E*/*W*⟩} {⟨*A*⟩} {⟨*E*/*W*⟩} {⟨*B*⟩}

```
\newcommand{\LWR@diagbox@AB}[4]{
\begingroup%
\LetLtxMacro\\\newline%
\BlockClassSingle{diagbox#1}{#2}%
\BlockClassSingle{diagbox#3}{#4}%
\endgroup%
\LWR@stoppars%
}
```

\LWR@diagboxNW {⟨A⟩} {⟨B⟩}

```
\newcommand{\LWR@diagboxNW}[2]{%
\LWR@diagbox@AB{E}{#2}{W}{#1}%
}
```

Likewise for NE, SW, SE:

```
\newcommand{\LWR@diagboxNE}[2]{%
\LWR@diagbox@AB{W}{#1}{E}{#2}%
}

\let\LWR@diagboxSW\LWR@diagboxNE
\let\LWR@diagboxSE\LWR@diagboxNW
```

\diagbox@double {⟨keys⟩} {⟨A⟩} {⟨B⟩}

```
\def\diagbox@double#1#2#3{%
\setkeys{diagbox}{dir=NW,#1}%
\@nameuse{LWR@diagbox\diagbox@dir}{#2}{#3}%
}
```

\LWR@diagboxTNW {⟨title⟩} {⟨A⟩} {⟨B⟩}

```
\newcommand{\LWR@diagboxTNW}[3]{%
\BlockClassSingle{diagboxtitleN}{#1}
\LWR@diagboxNW{#2}{#3}
}
```

Likewise for NE, SW, SE:

```
\newcommand{\LWR@diagboxTNE}[3]{%
\BlockClassSingle{diagboxtitleN}{#1}
\LWR@diagboxNE{#2}{#3}
}

\newcommand{\LWR@diagboxTSW}[3]{%
\LWR@diagboxSW{#2}{#3}
\BlockClassSingle{diagboxtitleS}{#1}
\LWR@stoppars%
}

\newcommand{\LWR@diagboxTSE}[3]{%
\LWR@diagboxSE{#2}{#3}
\BlockClassSingle{diagboxtitleS}{#1}
\LWR@stoppars%
}
```

\diagbox@triple {⟨keys⟩} {⟨A⟩} {⟨T⟩} {⟨B⟩}

```
\def\diagbox@triple#1#2#3#4{%
\setkeys{diagbox}{dir=NW,#1}%
\@nameuse{LWR@diagboxT\diagbox@dir}{#3}{#2}{#4}%
}
```

File 86 **lwarp-dingbat.sty**

§ 183 Package

dingbat

(*Emulates or patches code by* Scott Pakin.)

Pkg dingbat dingbat is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{dingbat}[2001/04/27]

\newcommand*{\LWR@dingbatsymbol}[1]{\HTMLunicode{#1}}

\newcommand{\LWR@HTML@rightpointright}{\LWR@dingbatsymbol{261E}}
\newcommand{\LWR@HTML@leftpointright}{\LWR@dingbatsymbol{261E}}
\newcommand{\LWR@HTML@leftthumbsdown}{\LWR@dingbatsymbol{1F44E}}
\newcommand{\LWR@HTML@leftthumbsup}{\LWR@dingbatsymbol{1F44D}}
\newcommand{\LWR@HTML@rightpointleft}{\LWR@dingbatsymbol{261C}}
\newcommand{\LWR@HTML@rightthumbsdown}{\LWR@dingbatsymbol{1F44E}}
\newcommand{\LWR@HTML@rightthumbsup}{\LWR@dingbatsymbol{1F44D}}
\newcommand{\LWR@HTML@squarewithdots}{\LWR@dingbatsymbol{25C7}}
\newcommand{\LWR@HTML@filledsquarewithdots}{\LWR@dingbatsymbol{25C6}}
\newcommand{\LWR@HTML@Sborder}{\LWR@dingbatsymbol{271A}}
\newcommand{\LWR@HTML@Zborder}{\LWR@dingbatsymbol{274B}}
\newcommand{\LWR@HTML@largepencil}{\LWR@dingbatsymbol{270E}}
\newcommand{\LWR@HTML@anchor}{\LWR@dingbatsymbol{2693}}
\newcommand{\LWR@HTML@carriagereturn}{\LWR@dingbatsymbol{23CE}}
\newcommand{\LWR@HTML@checkmark}{\LWR@dingbatsymbol{2713}}
\newcommand{\LWR@HTML@eye}{\LWR@dingbatsymbol{1F441}}
\newcommand{\LWR@HTML@satellitedish}{\LWR@dingbatsymbol{1F4E1}}
\newcommand{\LWR@HTML@smallpencil}{\LWR@dingbatsymbol{270E}}

\LWR@formatted{rightpointright}
\LWR@formatted{leftpointright}
\LWR@formatted{leftthumbsdown}
\LWR@formatted{leftthumbsup}
\LWR@formatted{rightpointleft}
\LWR@formatted{rightthumbsdown}
\LWR@formatted{rightthumbsup}
\LWR@formatted{squarewithdots}
\LWR@formatted{filledsquarewithdots}
\LWR@formatted{Sborder}
\LWR@formatted{Zborder}
\LWR@formatted{largepencil}
\LWR@formatted{anchor}
\LWR@formatted{carriagereturn}
\LWR@formatted{checkmark}
\LWR@formatted{eye}
\LWR@formatted{satellitedish}
\LWR@formatted{smallpencil}
```

File 87 **lwarp-dprogress.sty**

§ 184 Package dprogress

Pkg dprogress dprogress is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{dprogress}[2008/02/21]
```

File 88 **lwarp-draftcopy.sty**

§ 185 Package draftcopy

Pkg draftcopy draftcopy is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{draftcopy}[2002/02/25]

\newcommand{\draftcopyVersion}[1]{}
\newcommand{\draftcopySetGrey}[1]{}
\newcommand{\draftcopySetScale}[1]{}
\newcommand{\draftcopySetScaleFactor}[1]{}
\newcommand{\draftcopyFirstPage}[1]{}
\newcommand{\draftcopyLastPage}[1]{}
\newcommand{\draftcopyName}[2]{}
\newcommand{\draftcopyPageTransform}[1]{}
\newcommand{\draftcopyBottomTransform}[1]{}
\newcommand{\draftcopyPageX}[1]{}
\newcommand{\draftcopyPageY}[1]{}
\newcommand{\draftcopyBottomX}[1]{}
\newcommand{\draftcopyBottomY}[1]{}
```

File 89 **lwarp-draftfigure.sty**

§ 186 Package draftfigure

Pkg draftfigure draftfigure is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{draftfigure}[2017/07/19]
\RequirePackage{xkeyval}

\define@key{draftfigure}{code}{}
\define@key{draftfigure}{noframe}[true]{}
\define@key{draftfigure}{filename}[true]{}
\define@key{draftfigure}{content}[]{}
\define@key{draftfigure}{style}[normal]{}
\define@key{draftfigure}{position}[left]{}
```

```
\define@key{draftfigure}{size}[normal]{}
\newcommand\setdf[1]{\setkeys{draftfigure}{#1}}
```

File 90 **lwarp-draftwatermark.sty**

§ 187 Package draftwatermark

(*Emulates or patches code by* Sergio Callegari.)

Pkg draftwatermark draftwatermark is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{draftwatermark}[2015/02/19]

\newcommand{\SetWatermarkAngle}[1]{}
\newcommand{\SetWatermarkColor}[1]{}
\newcommand{\SetWatermarkLightness}[1]{}
\newcommand{\SetWatermarkFontSize}[1]{}
\newcommand{\SetWatermarkScale}[1]{}
\newcommand{\SetWatermarkHorCenter}[1]{}
\newcommand{\SetWatermarkVertCenter}[1]{}
\newcommand{\SetWatermarkText}[1]{}
```

File 91 **lwarp-easy-todo.sty**

§ 188 Package easy-todo

(*Emulates or patches code by* Juan Rada-Vilela.)

Pkg easy-todo easy-todo is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{easy-todo}[2014/01/01]
```

\listoftodos Modified to correct buggy use of \flushright.

```
\let\LWR@origlistoftodos\listoftodos

\renewcommand{\listoftodos}{%
\begingroup
\renewcommand{\flushright}{}
\LWR@origlistoftodos
\endgroup
}
```

\todoii Modified to use \textcolor instead of \color.

```
\renewcommand{\todoii}[2]{%
\ifthenelse{\equal{\@todoobeyfinal}{true}}%
    {%
        \ifoptionfinal{\todoenable{false}}{\todoenable{true}}%
```

```
    }%
    {}%
\ifthenelse{\equal{\@todoenable}{true}}%
    {%
        \refstepcounter{todos}%
        \noindent{%
            \todocolor%
            \LWR@textcurrentcolor{%
                \normalfont\scriptsize{\bfseries{\thetodos.#1}}%
            }%
        }%
        \addcontentsline{lod}{todos}{\protect{\thetodos. }\LWR@isolate{#2}}%
    }%
    {}%
}
```

File 92 **lwarp-ebook.sty**

§ 189 Package ebook

(*Emulates or patches code by* Jørgen Steensgaard.)

Pkg ebook ebook is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{ebook}

\setcounter{secnumdepth}{0}
\setcounter{tocdepth}{2}

\providecommand{\pagefill}[1][0.001mm]{\noindent}

\providecommand{\ebook}{
\setcounter{secnumdepth}{0}
\setcounter{tocdepth}{2}
}
```

File 93 **lwarp-ed.sty**

§ 190 Package ed

(*Emulates or patches code by* Michael Kohlhase.)

Pkg ed ed is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{ed}[2012/01/29]
```

Bugs:

1. `todolist` fails with the `hide` option, as does `\edexplanation`.

2. \edstubURI is actually \edstuURI.

```
\RequirePackage{xcolor}

\renewenvironment{edstub}[2][The following blue text]
{%
    \def\@test{#1}%
    \begin{center}%
        \huge%
        \textcolor{red}{%
            #1 is only a provisional stub\\\Large
            the Office document
            \ifx\ed@stubURI\@empty{#2}\else\href{\ed@stubURI}{#2}\fi\
            contains more text\\which will be merged for the final document%
        }%
    \end{center}%
    \BlockClass[color:blue]{edstub}%
}
{\endBlockClass}
```

File 94 **lwarp-ellipsis.sty**

§ 191 Package ellipsis

(*Emulates or patches code by* Peter J. Heslin.)

Pkg ellipsis ellipsis is emulated.

```
\LWR@ProvidesPackageDrop{ellipsis}[2004/09/28]

\newcommand{\ellipsisgap}{0.1em}

\newcommand*{\midwordellipsis}{\,\textellipsis\,}
```

File 95 **lwarp-embrac.sty**

§ 192 Package embrac

(*Emulates or patches code by* Clemens Niederberger.)

Pkg embrac embrac is nullfied for HTML and used as-is for print.

for HTML output:

```
\LWR@ProvidesPackagePass{embrac}[2017/07/04]

\LetLtxMacro\LWR@orig@HTML@emph\LWR@HTML@emph
\RenewDocumentCommand{\LWR@HTML@emph}{s m}{\LWR@orig@HTML@emph{#2}}

\LetLtxMacro\LWR@orig@HTML@textit\LWR@HTML@textit
\RenewDocumentCommand{\LWR@HTML@textit}{s m}{\LWR@orig@HTML@textit{#2}}
```

```
\LetLtxMacro\LWR@orig@HTML@textsl\LWR@HTML@textsl
\RenewDocumentCommand{\LWR@HTML@textsl}{s m}{\LWR@orig@HTML@textsl{#2}}

\ifxetexorluatex
    \LetLtxMacro\LWR@orig@HTML@textsi\LWR@HTML@textsi
    \RenewDocumentCommand{\LWR@HTML@textsi}{s m}{%
    \LWR@orig@HTML@textsi{#2}}
\fi

\AtBeginDocument{
    \LWR@formatted{emph}
    \LWR@formatted{textit}
    \LWR@formatted{textsl}
    \ifxetexorluatex
        \LWR@formatted{textsi}
    \fi
}

\newcommand{\LWR@HTML@EmbracOff}{}
\LWR@formatted{EmbracOff}

\newcommand{\LWR@HTML@EmbracOn}{}
\LWR@formatted{EmbracOn}
```

File 96 **lwarp-emptypage.sty**

§ 193 Package emptypage

Pkg emptypage emptypage is ignored.

for HTML output: Discard all options for lwarp-emptypage:

```
\LWR@ProvidesPackageDrop{emptypage}[2010/05/30]
```

File 97 **lwarp-endfloat.sty**

§ 194 Package endfloat

Pkg endfloat endfloat is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{endfloat}[2018/03/24]

\newcommand\figureplace{}
\newcommand\tableplace{}
\newcommand\floatplace[1]{}
\newcounter{posttable}
\newcounter{postfigure}
\newcommand*{\theposttbl}{}
\newcommand*{\thepostfig}{}
```

```
\newcommand{\AtBeginFigures}[1]{}
\newcommand{\AtBeginTables}[1]{}
\newcommand{\AtBeginDelayedFloats}[1]{}
\newcommand*{\processdelayedfloats}{}
\newcommand*{\efloatseparator}{}
\def\efloattype{}
\providecommand\efloatheading[1]{}
\providecommand\efloatpreamble{}
\providecommand\efloatpostamble{}
```

File 98 **lwarp-endheads.sty**

§ 195 Package

endheads

Pkg endheads endheads is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{endheads}[2017/04/06]

\newcommand{\changesinglepageabbrev}[1]{}
\newcommand{\changemultiplepageabbrev}[1]{}
\newcommand{\changenotesname}[1]{}
\newcommand{\changenotesheader}[1]{}
\newcommand{\changenotescontentsname}[1]{}
\newcommand{\changechapternotesline}[1]{}
\newcommand{\checknoteheaders}{}
\newif\ifnotesincontentson \notesincontentsonfalse
\newcommand{\notesincontents}{\notesincontentsontrue}
\newif\ifendnoteheaderson \endnoteheadersonfalse
\newcommand{\setupendnoteheaders}{%
    \endnoteheadersontrue%
}
\newif\iftitleinnotes \titleinnotestrue
\newcommand{\styleforchapternotebegin}{}
\newcommand{\styleforchapternoteend}{}
\newcommand{\setstyleforchapternotebegin}[1]{%
    \renewcommand{\styleforchapternotebegin}{#1}%
}
\newcommand{\setstyleforchapternoteend}[1]{%
    \renewcommand{\styleforchapternoteend}{#1}%
}
\newcommand{\resetendnotes}{}
\newif\ifnotesbychapteron \notesbychapteronfalse
\newcommand{\notesbychapter}{\notesbychapterontrue}
```

File 99 **lwarp-endnotes.sty**

§ 196 Package

endnotes

(*Emulates or patches code by* John Lavagnino.)

Pkg endnotes Used as-is.

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
```

for HTML output:

```
\LWR@ProvidesPackagePass{endnotes}

\def\enoteformat{%
% \rightskip\z@ \leftskip\z@ \parindent=1.8em
\leavevmode
% \llap{
\makeenmark
% }
}

\def\@makeenmark{\hbox{\LWR@htmlspan{sup}{\normalfont\theenmark}}}
\def\makeenmark{\@makeenmark}
```

File 100 **lwarp-enumerate.sty**

§ 197 Package enumerate

Pkg enumerate 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:

```
\LWR@ProvidesPackagePass{enumerate}[2015/07/23]
```

File 101 **lwarp-enumitem.sty**

§ 198 Package enumitem

(*Emulates or patches code by* JAVIER BEZOS.)

Pkg enumitem enumitem is supported with minor adjustments.

for HTML output:

```
\LWR@ProvidesPackagePass{enumitem}[2018/11/30]
```

for HTML output:

```
\begin{warpHTML}
```

\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.

```
\let\LWR@enumitem@orignewlist\newlist

\renewcommand*{\newlist}[3]{%
\LWR@enumitem@orignewlist{#1}{#2}{#3}%
\AtBeginEnvironment{#1}{\@nameuse{LWR@#2start}}%
\AtEndEnvironment{#1}{\@nameuse{LWR@#2end}}%
}

\def\DrawEnumitemLabel{}


\end{warpHTML}
```

File 102 **lwarp-epigraph.sty**

§ 199 Package

epigraph

(*Emulates or patches code by* Peter Wilson.)

Pkg epigraph epigraph is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{epigraph}[2009/09/02]

\DeclareDocumentCommand{\qitem}{m m}
{
\begin{BlockClass}{qitem}
#1
\ifbool{FormatWP}
{\begin{BlockClass}[border-top:1px solid gray]{epigraphsource}}
{\begin{BlockClass}{epigraphsource}}
#2
\end{BlockClass}
\end{BlockClass}
}


\DeclareDocumentCommand{\epigraph}{m m}
{
\begin{LWR@BlockClassWP}{\LWR@print@mbox{text-align:right}}{}{epigraph}
\qitem{#1}{#2}
\end{LWR@BlockClassWP}
}

\DeclareDocumentEnvironment{epigraphs}{}
{\LWR@BlockClassWP{\LWR@print@mbox{text-align:right}}{}{epigraph}}
{\endLWR@BlockClassWP}
```

Use css to format epigraphs.

The following are null commands for source compatibility:

```
\newenvironment*{flushepinormal}{}{}


\@ifclassloaded{memoir}{
\setlength{\epigraphwidth}{.5\linewidth}
\renewcommand{\textflush}{flushepinormal}
\renewcommand{\epigraphhead}[2][0]{#2}
\renewcommand{\dropchapter}[1]{}
\renewcommand*{\undodrop}{}
}{% not memoir
\newlength{\epigraphwidth}
\setlength{\epigraphwidth}{.5\linewidth}
\newcommand{\textflush}{flushepinormal}
\newcommand{\epigraphflush}{flushright}
\newcommand{\sourceflush}{flushright}
\newcommand*{\epigraphsize}{\small}
\newlength{\epigraphrule}
\newlength{\beforeepigraphskip}
\newlength{\afterepigraphskip}
\newcommand{\epigraphhead}[2][0]{#2}
\newcommand{\dropchapter}[1]{}
\newcommand*{\undodrop}{}
}% not memoir

\let\cleartoevenpage\relax% also in nextpage
\newcommand{\cleartoevenpage}[1][]{}
```

File 103 **lwarp-epsfig.sty**

§ 200 Package

epsfig

Pkg epsfig epsfig is emulated for use by lwarp.

⚠ Only the LaTeX2e syntax is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{epsfig}[2017/06/25]
```

A few additional keys to capture the filename:

```
\RequirePackage{graphics}

\define@key{igraph}{file}{%
    \xdef\LWR@epsfig@filename{#1}%
}

\define@key{igraph}{figure}{%
    \xdef\LWR@epsfig@filename{#1}%
}

```

```
\define@key{igraph}{prolog}{}

\define@key{igraph}{silent}[]{}
```

The captured filename is used as the argument to `\includegraphics`:

```
\newcommand{\epsfig}[1]{\includegraphics[#1]{\LWR@epsfig@filename}}

\newcommand{\psfig}[1]{\includegraphics[#1]{\LWR@epsfig@filename}}
```

File 104 **lwarp-epstopdf.sty**

§ 201 Package epstopdf

Pkg epstopdf

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:

```
\LWR@ProvidesPackagePass{epstopdf}[2016/05/15]
```

File 105 **lwarp-epstopdf-base.sty**

§ 202 Package epstopdf-base

Pkg epstopdf-base

⚠ 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:

```
\LWR@ProvidesPackagePass{epstopdf-base}[2016/05/15]
```

Redefine to remember the image filename, replacing .pdf with .svg. Use the epstopdf print version inside a `latexmage`.

```
\newcommand*{\LWR@HTML@ETE@OrgGin@setfile}[3]{%
    \edef\LWR@tempone{#3}%
    \StrSubstitute{\LWR@tempone}{.pdf}{.svg}[\LWR@tempone]%
    \StrSubstitute{\LWR@tempone}{.PDF}{.SVG}[\LWR@tempone]%
    \xdef\LWR@parsedfilename{\LWR@tempone}%
}

\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:

```
\renewcommand*{\LWR@HTML@Gin@setfile}[3]{%
    \ETE@Gin@setfile{#1}{#2}{#3}%
}
```

Allow .eps images to be found if a suffix is not provided:

```
\AtBeginDocument{
\DeclareGraphicsExtensions{%
    .eps,.EPS,.svg,.SVG,.gif,.GIF,.png,.PNG,.jpg,.JPG,.jpeg,.JPEG%
}
\DeclareGraphicsRule{.svg}{svg}{.svg}{}
\DeclareGraphicsRule{.SVG}{svg}{.SVG}{}
}
```

Likewise when inside a lateximage:

```
\appto\LWR@restoreorigformatting{%
\DeclareGraphicsExtensions{%
    .eps,.EPS,.pdf,.PDF,.gif,.GIF,.png,.PNG,.jpg,.JPG,.jpeg,.JPEG%
}%
}
```

File 106 **lwarp-errata.sty**

§ 203 Package

errata

(*Emulates or patches code by* Michael Kohlhase.)

Pkg errata 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:

```
\StartDefiningMath
```

Now load the package:

```
\LWR@ProvidesPackagePass{errata}[2006/11/12]
```

Patches for dynamic inline math:

```
\xpatchcmd{\erratumAdd}
    {$_a^{\arabic{erratum}}$}
%     {\inlinemathother$_a^{\arabic{erratum}}$\inlinemathnormal}
    {\textsubscript{a}\textsuperscript{\arabic{erratum}}}
    {}
```

```
    {\LWR@patcherror{erratum}{erratumAdd}}

\xpatchcmd{\erratumDelete}
    {$_d^{\arabic{erratum}}$}
%     {\inlinemathother$_d^{\arabic{erratum}}$\inlinemathnormal}
    {\textsubscript{d}\textsuperscript{\arabic{erratum}}}
    {}
    {\LWR@patcherror{erratum}{erratumDelete}}

\xpatchcmd{\erratumReplace}
    {$_r^{\arabic{erratum}}$}
%     {\inlinemathother$_r^{\arabic{erratum}}$\inlinemathnormal}
    {\textsubscript{r}\textsuperscript{\arabic{erratum}}}
    {}
    {\LWR@patcherror{erratum}{erratumReplace}}

\xpatchcmd{\erratum}
    {$_a$}
%     {\inlinemathother$_a$\inlinemathnormal}
    {\textsubscript{a}}
    {}
    {\LWR@patcherror{erratum}{erratumDelete}}

\xpatchcmd{\erratum}
    {$_d^{\@thefnmark}$}
%     {\inlinemathother$_d^{\@thefnmark}$\inlinemathnormal}
    {\textsubscript{d}\textsuperscript{\@thefnmark}}
    {}
    {\LWR@patcherror{erratum}{eDelete}}

\xpatchcmd{\erratum}
    {$_r^{\@thefnmark}$}
%     {\inlinemathother$_r^{\@thefnmark}$\inlinemathnormal}
    {\textsubscript{r}\textsuperscript{\@thefnmark}}
    {}
    {\LWR@patcherror{erratum}{eReplace}}
```

Finish the current page's errata before closing and reloading the list:

```
\preto\PrintErrata{\LWR@orignewpage}
```

No longer defining math macros with the HTML $:

```
\StopDefiningMath
```

File 107 **lwarp-eso-pic.sty**

§ 204 Package eso-pic

(*Emulates or patches code by* ROLF NIEPRASCHK.)

Pkg eso-pic eso-pic is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{eso-pic}[2018/04/12]

\newcommand*{\LenToUnit}{}
\newcommand{\AtPageUpperLeft}[1]{}
\newcommand{\AtPageLowerLeft}[1]{}
\newcommand{\AtPageCenter}[1]{}
\newcommand{\AtStockLowerLeft}[1]{}
\newcommand{\AtStockUpperLeft}[1]{}
\newcommand{\AtStockCenter}[1]{}
\newcommand{\AtTextUpperLeft}[1]{}
\newcommand{\AtTextLowerLeft}[1]{}
\newcommand{\AtTextCenter}[1]{}
\NewDocumentCommand{\AddToShipoutPictureBG}{s +m}{}

\newcommand{\AddToShipoutPicture}{\AddToShipoutPictureBG}
\NewDocumentCommand{\AddToShipoutPictureFG}{s +m}{}
\newcommand*{\ClearShipoutPictureBG}{}
\newcommand*{\ClearShipoutPicture}{}
\newcommand*{\ClearShipoutPictureFG}{}
\newcommand{\gridSetup}[6][]{}
```

File 108 **lwarp-eurosym.sty**

§ 205 Package eurosym

(*Emulates or patches code by* Henrik Theiling.)

Pkg eurosym eurosym is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{eurosym}[1998/08/06]

\renewrobustcmd\officialeuro{\HTMLentity{euro}}
\let\geneuro\officialeuro
\let\geneuronarrow\officialeuro
\let\geneurowide\officialeuro
\let\euro\officialeuro
\renewrobustcmd\eurobars{}
\renewrobustcmd\eurobarsnarrow{}
\renewrobustcmd\eurobarswide{}
```

File 109 **lwarp-everypage.sty**

§ 206 Package everypage

(*Emulates or patches code by* Sergio Callegari.)

Pkg everypage everypage is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{everypage}[2007/06/20]
```

```
\newcommand*{\AddEverypageHook}[1]{}
\newcommand*{\AddThispageHook}[1]{}
```

File 110 **lwarp-everyshi.sty**

§ 207 Package everyshi

(*Emulates or patches code by* Martin Schröder.)

Pkg everyshi Emulated.

for HTML output: Discard all options for lwarp-everyshi:

```
\LWR@ProvidesPackageDrop{everyshi}[2001/05/15]

\newcommand*{\EveryShipout}[1]{}
\newcommand*{\AtNextShipout}[1]{}
```

File 111 **lwarp-extramarks.sty**

§ 208 Package extramarks

(*Emulates or patches code by* Piet van Oostrum.)

Pkg extramarks extramarks is emulated.

for HTML output: Discard all options for lwarp-extramarks:

```
\LWR@ProvidesPackageDrop{extramarks}[2019/01/31]

\newcommand*{\extramarks}[2]{}
\newcommand*{\firstleftxmark}{}
\newcommand*{\lastleftxmark}{}
\newcommand*{\firstrightxmark}{}
\newcommand*{\lastrightxmark}{}
\newcommand*{\firstxmark}{}
\newcommand*{\lastxmark}{}
\newcommand*{\topxmark}{}
\newcommand*{\topleftxmark}{}
\newcommand*{\toprightxmark}{}
\newcommand*{\firstleftmark}{}
\newcommand*{\lastrightmark}{}
\newcommand*{\firstrightmark}{}
\newcommand*{\lastleftmark}{}
```

File 112 **lwarp-fancybox.sty**

§ 209 Package

fancybox

(*Emulates or patches code by* Timothy Van Zandt.)

Pkg fancybox

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 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 loses 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.

⚠ \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.

```
\LWR@ProvidesPackagePass{fancybox}[2010/05/15]
```

After the preamble is loaded, after any patches to Verbatim:

```
\AfterEndPreamble{
\LWR@traceinfo{Patching fancybox.}
```

\VerbatimFootnotes

Patched to use the new version.

```
\def\VerbatimFootnotes{%
\let\@footnotetext\V@footnotetext%
\let\LWR@footnotetext\V@footnotetext% lwarp
}
```

\V@@footnotetext

Patches in a subset of lwarp's \LWR@footnotetext to the fancyvrb version of \V@@footnotetext.

```
\def\V@@footnotetext{%
\LWR@traceinfo{V@footnotetext}%
\global\setbox\LWR@footnotes=\vbox\bgroup%
```

Add to any current footnotes:

```
    \unvbox\LWR@footnotes%
```

Remember the footnote number for \ref:

```
    \protected@edef\@currentlabel{%
        \csname p@footnote\endcsname\@thefnmark%
    }% @currentlabel
```

Use HTML superscripts in the footnote even inside a `lateximage`:

```
    \renewrobustcmd{\textsuperscript}[1]{\LWR@htmlspan{sup}{##1}}%
```

Use paragraph tags if in a tabular data cell or a `lateximage`:

```
    \ifthenelse{%
        \boolean{LWR@doingstartpars} \AND%
        \cnttest{\value{LWR@lateximagedepth}}{=}{0}%
    }%
        {}%
        {\LWR@htmltagc{\LWR@tagregularparagraph}\LWR@orignewline}%
```

Append the footnote to the list:

```
    \@makefntext{}%
```

```
  \bgroup%
  \aftergroup{\V@@@footnotetext}%
  \ignorespaces%
}%
```

```
}% AfterEndPreamble
```

```
\renewcommand*{\@shadowbox}[1]{%
\ifbool{FormatWP}%
{\InlineClass[border:1px solid black]{shadowbox}{#1}}%
{\InlineClass{shadowbox}{#1}}%
}

\renewcommand*{\@doublebox}[1]{%
\ifbool{FormatWP}%
{\InlineClass[border:1px double black]{doublebox}{#1}}%
{\InlineClass{doublebox}{#1}}%
}

\renewcommand*{\@ovalbox}[2]{%
\ifbool{FormatWP}%
{\InlineClass[border:1px solid black; border-radius:1ex]{ovalbox}{#2}}%
{%
    \ifthenelse{\isequivalentto{#1}{\thinlines}}%
        {\InlineClass{ovalbox}{#2}}%
        {\InlineClass{Ovalbox}{#2}}%
}%
}
```

Convert minipages, parboxes, and lists into linear text using the `LWR@nestspan` environment:

```
\let\LWR@origSbox\Sbox

\def\Sbox{\LWR@origSbox\LWR@nestspan}

```

```
53
54 \let\LWR@origendSbox\endSbox
55
56 \def\endSbox{\endLWR@nestspan\LWR@origendSbox}
```

`Beqnarray` is adapted for MathJax or enclosed inside a `lateximage`:

```
57 \RenewEnviron{Beqnarray}
58 {\LWR@eqnarrayfactor}
59
60 \csgpreto{Beqnarray*}{\boolfalse{LWR@numbereqnarray}}
```

`\GenericCaption` is enclosed in an HTML block:

```
61 \renewcommand{\GenericCaption}[1]{%
62 \LWR@figcaption%
63 \LWR@isolate{#1}%
64 \endLWR@figcaption%
65 }
```

`Btrivlist` is enclosed in an HTML block. This is a tabular, and does not use `\item`.

`\trivlist` {⟨*l/c/r*⟩} [⟨*t/c/b*⟩]

```
66 \RenewDocumentEnvironment{Btrivlist}{m o}
67 {%
68     \begin{BlockClass}{Btrivlist}%
69     \tabular{#1}%
70 }
71 {%
72     \endtabular%
73     \end{BlockClass}%
74 }
```

`Btrivlist` is also neutralized when used inside a span:

```
75 \AtBeginEnvironment{LWR@nestspan}{%
76 \RenewDocumentEnvironment{Btrivlist}{m o}{}{}%
77 }
```

lwarp's handling of `\item` is patched to accept fancybox's optional arguments:

```
78 \let\LWRFB@origitemizeitem\LWR@itemizeitem
79 \let\LWRFB@origdescitem\LWR@descitem
80
81 \RenewDocumentCommand{\LWR@itemizeitem}{d()o}{%
82     \IfValueTF{#2}{%
83         \LWRFB@origitemizeitem[#2]%
84     }{%
85         \LWRFB@origitemizeitem%
86     }%
87 }
88
```

```
\RenewDocumentCommand{\LWR@descitem}{d()o}{%
    \IfValueTF{#2}{%
        \LWRFB@origdescitem[#2]~%
    }{%
        \LWRFB@origdescitem%
    }%
}

\RenewDocumentCommand{\LWR@nestspanitem}{d()}{%
\if@newlist\else{\LWR@htmltagc{br /}}\fi%
\LWR@origitem%
}
```

The various boxed lists become regular lists:

```
\renewenvironment{Bitemize}[1][]{\begin{itemize}}{\end{itemize}}
\renewenvironment{Benumerate}[1][]{\begin{enumerate}}{\end{enumerate}}
\renewenvironment{Bdescription}[1][]{\begin{description}}{\end{description}}
```

\boxput simply prints one then the other argument, side-by-side instead of above and behind:

```
\RenewDocumentCommand{\boxput}{s d() m m}{%
\IfBooleanTF{#1}{#3\quad#4}{#4\quad#3}%
}
```

Neutralized commands:

```
\RenewDocumentCommand{\fancyput}{s d() m}{}
\RenewDocumentCommand{\thisfancyput}{s d() m}{}

\RenewDocumentCommand{\fancypage}{m m}{}
\RenewDocumentCommand{\thisfancypage}{m m}{}

\def\LandScape#1{}
\def\endLandScape{}
\def\@Landscape#1#2#3{}
\def\endLandscape{}
```

Low-level patches for UseVerbatim and friends:

```
\let\LWRFB@UseVerbatim\UseVerbatim
\renewcommand*{\UseVerbatim}[1]{%
\LWR@atbeginverbatim{3}{Verbatim}%
\LWRFB@UseVerbatim{#1}%
\LWR@afterendverbatim{.5}%
}

\let\LWRFB@LUseVerbatim\LUseVerbatim

\renewcommand*{\LUseVerbatim}[1]{%
\LWR@atbeginverbatim{3}{LVerbatim}%
```

```
\noindent%
\LWRFB@LUseVerbatim{#1}%
\LWR@afterendverbatim{.5}%
}

\def\@BUseVerbatim[#1]#2{%
\LWR@atbeginverbatim{3}{BVerbatim}%
\LWRFB@UseVerbatim{#2}%
\LWR@afterendverbatim{.5}%
}
```

File 113 **lwarp-fancyhdr.sty**

§ 210 Package

fancyhdr

(*Emulates or patches code by* Piet van Oostrum.)

Pkg fancyhdr fancyhdr is nullified.

for HTML output: Discard all options for lwarp-fancyhdr:

```
\LWR@ProvidesPackageDrop{fancyhdr}[2019/01/31]

\newcommand*{\fancyhead}[2][]{}
\newcommand*{\fancyfoot}[2][]{}
\newcommand*{\fancyhf}[2][]{}
\newcommand*{\fancypagestyle}[2]{}
\newcommand*{\lhead}[2][]{}
\newcommand*{\chead}[2][]{}
\newcommand*{\rhead}[2][]{}
\newcommand*{\lfoot}[2][]{}
\newcommand*{\cfoot}[2][]{}
\newcommand*{\rfoot}[2][]{}
\newcommand*{\headrulewidth}{}
\newcommand*{\footrulewidth}{}
\newcommand*{\headrule}{}
\newcommand*{\footrule}{}
\newlength{\headwidth}
\newcommand*{\fancyheadoffset}[2][]{}
\newcommand*{\fancyfootoffset}[2][]{}
\newcommand*{\fancyhfoffset}[2][]{}
\newcommand*{\iffloatpage}[2]{#2}
\newcommand*{\ifftopfloat}[2]{#2}
\newcommand*{\iffbotfloat}[2]{#2}
\newcommand*{\iffootnote}[2]{#2}
```

File 114 **lwarp-fancyref.sty**

§ 211 Package

fancyref

Pkg fancyref fancyref is emulated.

for HTML output:

```
\LWR@ProvidesPackagePass{fancyref}[1999/02/03]
```

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.)

```
\newcommand*{\LWRfref@parenfancyrefhook}[1]{(#1)}

\ifdefstrequal{\fancyrefhook}{\LWRfref@parenfancyrefhook}
{}{
    \renewcommand*{\fancyrefhook}[1]{#1}%
}
```

Modified to ignore the page number and varioref.

```
\renewcommand*{\@f@ref}[4]{%
  \@ifundefined{#1r@#2@#3}{%
    \PackageError{fancyref}{%
      \backslashchar#1ref\space format ``#2''
      undefined\MessageBreak
      for label type ``#3''%
    }{%
      The format ``#2'' was not defined for the label type
      ``#3''\MessageBreak
      and the \backslashchar#1ref\space command.  Perhaps
      you have only misspelled its name.\MessageBreak
      Otherwise you will have to define it with
      \protect\new#1refformat\MessageBreak
      prior to using it.%
    }%
  }{%
    \fancyrefhook{%
      \@nameuse{#1r@#2@#3}%
        {\ref{#3\fancyrefargdelim#4}}%
%         {\pageref{#3\fancyrefargdelim#4}}% original
%         {\@fancyref@page@ref{#3\fancyrefargdelim#4}}% original
        {}% lwarp
        {}% lwarp
    }%
  }%
}%
```

File 115 **lwarp-fancytabs.sty**

§ 212 Package

fancytabs

Pkg fancytabs fancytabs is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{fancytabs}[2016/03/29]

\newcommand{\fancytab}[3][RIGHT]{}
\newcommand{\fancytabsStyle}[1]{}
\newcommand{\fancytabsHeight}[1]{}
\newcommand{\fancytabsWidth}[1]{}
\newcommand{\fancytabsCount}[1]{}
\newcommand{\fancytabsLeftColor}[1]{}
\newcommand{\fancytabsRightColor}[1]{}
\newcommand{\fancytabsTop}[1]{}
\newcommand{\fancytabsTextVPos}[1]{}
\newcommand{\fancytabsTextHPos}[1]{}
\newcommand{\fancytabsGap}[1]{}
\newcommand{\fancytabsFloor}[1]{}
\newcommand{\fancytabsRotate}[1]{}
```

File 116 **lwarp-fancyvrb.sty**

§ 213 Package

fancyvrb

(*Emulates or patches code by* Timothy Van Zandt.)

Pkg fancyvrb fancyvrb is supported with some patches.

⚠ \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.

```
\RequirePackage{xcolor}% for \convertcolorspec

\LWR@ProvidesPackagePass{fancyvrb}[2008/02/07]
```

Initial default patch for fancyvrb:

```
\fvset{frame=none}%
```

After the preamble is loaded, after any patches to `Verbatim`:

```
\AfterEndPreamble{
\LWR@traceinfo{Patching fancyvrb.}
```

`\VerbatimFootnotes` Patched to use the new version.

```
\def\VerbatimFootnotes{%
\let\@footnotetext\V@footnotetext%
\let\footnote\V@footnote%
\let\LWR@footnotetext\V@footnotetext% lwarp
}
```

`\V@@footnotetext` Patches in a subset of lwarp's `\LWR@footnotetext` to the fancyvrb version of `\V@@footnotetext`.

```
\def\V@@footnotetext{%
\LWR@traceinfo{V@footnotetext}%
\global\setbox\LWR@footnotes=\vbox\bgroup%
```

Add to any current footnotes:

```
    \unvbox\LWR@footnotes%
```

Remember the footnote number for `\ref`:

```
    \protected@edef\@currentlabel{%
        \csname p@footnote\endcsname\@thefnmark%
    }% @currentlabel
```

Use HTML superscripts in the footnote even inside a `lateximage`:

```
    \renewrobustcmd{\textsuperscript}[1]{\LWR@htmlspan{sup}{##1}}%
```

Use paragraph tags if in a tabular data cell or a `lateximage`:

```
    \ifthenelse{%
        \boolean{LWR@doingstartpars} \AND%
        \cnttest{\value{LWR@lateximagedepth}}{=}{0}%
    }%
        {}%
        {\LWR@htmltagc{\LWR@tagregularparagraph}\LWR@orignewline}%
```

Append the footnote to the list:

```
    \@makefntext{}%
```

```
  \bgroup%
  \aftergroup{\V@@@footnotetext}%
  \ignorespaces%
}%
```

```
\preto\FVB@Verbatim{\LWR@forcenewpage}
\preto\FVB@LVerbatim{\LWR@forcenewpage}
% \preto\FVB@BVerbatim{\LWR@forcenewpage}% Fails, so done below.
```

Simplified to remove PDF formatting:

```
\def\FV@BeginListFrame@Single{%
  \FV@SingleFrameLine{\z@}%
}

\def\FV@EndListFrame@Single{%
  \FV@SingleFrameLine{\@ne}%
}

\def\FV@BeginListFrame@Lines{%
  \FV@SingleFrameLine{\z@}%
}

\def\FV@EndListFrame@Lines{%
    \FV@SingleFrameLine{\@ne}%
}

\renewcommand*{\FV@SingleFrameSep}{}
```

Adds HTML formatting:

```
\def\FV@BUseVerbatim#1{%
    \LWR@atbeginverbatim[\LWR@FVstyle]{0}{verbatim}%
    \FV@BVerbatimBegin#1\FV@BVerbatimEnd%
    \LWR@afterendverbatim{0}%
}
```

`\LWR@FVstyle` Holds the style of the verbatim.

```
\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.

```
\newcommand*{\LWR@fvstartnone}{%
\LWR@traceinfo{fvstartnone}%
% \hbox to\z@{
\LWR@atbeginverbatim[\LWR@FVstyle]{0}{verbatim}%
% }%
}

\newcommand*{\LWR@fvendnone}{%
\LWR@traceinfo{fvendnone}%
% \hbox to\z@{
\LWR@afterendverbatim{0}%
% }%
}

\newcommand*{\LWR@fvstartsingle}{%
\LWR@traceinfo{fvstartsingle}%
\LWR@fvstartnone%
\FV@BeginListFrame@Single%
}
```

```
\newcommand*{\LWR@fvendsingle}{%
\LWR@traceinfo{fvendsingle}%
\FV@EndListFrame@Single%
\LWR@fvendnone%
}

\newcommand*{\LWR@fvstartline}{%
\LWR@traceinfo{fvstartline}%
\LWR@fvstartnone%
% \setlength{\LWR@templengthone}{\baselineskip}%
\FV@BeginListFrame@Lines%
% \setlength{\baselineskip}{\LWR@templengthone}%
% \setlength{\baselineskip}{5pt}%
}

\newcommand*{\LWR@fvendline}{%
\LWR@traceinfo{fvendline}%
\FV@EndListFrame@Lines%
\LWR@fvendnone%
}
```

The following patches select the start/left/right/end behaviors depending on `frame`. Original code is from the fancyvrb package.

```
\newcommand*{\LWR@FVfindbordercolor}{%
\FancyVerbRuleColor%
\LWR@findcurrenttextcolor%
\color{black}%
}

% border width of \FV@FrameRule
\newcommand*{\LWR@FVborderstyle}[1]{%
padding#1: \strip@pt\dimexpr \FV@FrameSep\relax\relax pt ; %
\LWR@FVfindbordercolor %
border#1: \strip@pt\dimexpr \FV@FrameRule\relax\relax pt %
solid \LWR@origpound\LWR@tempcolor ; %
}

\def\FV@Frame@none{%
\renewcommand*{\LWR@FVstyle}{\LWR@currenttextcolorstyle}%
\let\FV@BeginListFrame\LWR@fvstartnone%
\let\FV@LeftListFrame\relax%
\let\FV@RightListFrame\relax%
\let\FV@EndListFrame\LWR@fvendnone}

\FV@Frame@none% default values

\def\FV@Frame@single{%
\renewcommand*{\LWR@FVstyle}{\LWR@currenttextcolorstyle\LWR@FVborderstyle{}}%
\let\FV@BeginListFrame\LWR@fvstartsingle%
\let\FV@LeftListFrame\FV@LeftListFrame@Single%
\let\FV@RightListFrame\FV@RightListFrame@Single%
\let\FV@EndListFrame\LWR@fvendsingle}

```

```
\def\FV@Frame@lines{%
\renewcommand*{\LWR@FVstyle}{%
  \LWR@currenttextcolorstyle\LWR@FVborderstyle{-top}\LWR@FVborderstyle{-bottom}%
}%
\let\FV@BeginListFrame\LWR@fvstartline%
\let\FV@LeftListFrame\relax%
\let\FV@RightListFrame\relax%
\let\FV@EndListFrame\LWR@fvendline}

\def\FV@Frame@topline{%
\renewcommand*{\LWR@FVstyle}{\LWR@currenttextcolorstyle\LWR@FVborderstyle{-top}}%
\let\FV@BeginListFrame\LWR@fvstartline%
\let\FV@LeftListFrame\relax%
\let\FV@RightListFrame\relax%
\let\FV@EndListFrame\LWR@fvendnone}

\def\FV@Frame@bottomline{%
\renewcommand*{\LWR@FVstyle}{\LWR@currenttextcolorstyle\LWR@FVborderstyle{-bottom}}%
\let\FV@BeginListFrame\LWR@fvstartnone%
\let\FV@LeftListFrame\relax%
\let\FV@RightListFrame\relax%
\let\FV@EndListFrame\LWR@fvendline}

\def\FV@Frame@leftline{%
\renewcommand*{\LWR@FVstyle}{\LWR@currenttextcolorstyle\LWR@FVborderstyle{-left}}%
% To define the \FV@FrameFillLine macro (from \FV@BeginListFrame)
\ifx\FancyVerbFillColor\relax%
\let\FV@FrameFillLine\relax%
\else%
\@tempdima\FV@FrameRule\relax%
\multiply\@tempdima-\tw@%
\edef\FV@FrameFillLine{%
{\noexpand\FancyVerbFillColor{\vrule\@width\number\@tempdima sp}%
\kern-\number\@tempdima sp}}%
\fi%
\let\FV@BeginListFrame\LWR@fvstartnone%
\let\FV@LeftListFrame\FV@LeftListFrame@Single%
\let\FV@RightListFrame\relax%
\let\FV@EndListFrame\LWR@fvendnone}
```

Adds the optional label to the top and bottom edges. Original code is from the fancyvrb package.

```
\def\FV@SingleFrameLine#1{%
%   \hbox to\z@{%
%     \kern\leftmargin
    \ifnum#1=\z@\relax
      \let\FV@Label\FV@LabelBegin
    \else
      \let\FV@Label\FV@LabelEnd
    \fi
    \ifx\FV@Label\relax
%       \FancyVerbRuleColor{\vrule \@width\linewidth \@height\FV@FrameRule}%
    \else
      \ifnum#1=\z@
```

```
%         \setbox\z@\hbox{\strut\enspace\FV@LabelBegin\enspace\strut}%
        \ifx\FV@LabelPositionTopLine\relax
        \else
        \LWR@FVfindbordercolor
        \LWR@htmltagc{%
            div class="fancyvrblabel" % extra space
                style="color: \LWR@origpound\LWR@tempcolor"%
        }
        \LWR@print@textrm{\FV@LabelBegin}% \textrm preserves emdash
        \LWR@htmltagc{/div}
        \fi
      \else
%         \setbox\z@\hbox{\strut\enspace\FV@LabelEnd\enspace\strut}%
        \ifx\FV@LabelPositionBottomLine\relax
        \else
        \LWR@FVfindbordercolor

        \LWR@htmltagc{%
            div class="fancyvrblabel" % extra space
                style="color: \LWR@origpound\LWR@tempcolor"%
        }
        \LWR@print@textrm{\FV@LabelEnd}
        \LWR@htmltagc{/div}
        \fi
      \fi
    \fi
%     \hss
%     }
}
```

Processes each line, adding optional line numbers. Original code is from the fancyvrb package.

```
\def\FV@ListProcessLine#1{%
   \hbox to \hsize{%
%      \kern\leftmargin
      \hbox to \VerbatimHTMLWidth {%
       \ifcsvoid{FV@LeftListNumber}{}{\kern 2.5em}%
        \FV@LeftListNumber%
%       \FV@LeftListFrame
      \FancyVerbFormatLine{#1}%
      \hss%
%       \FV@RightListFrame
      \FV@RightListNumber%
    }%
      \hss% required to avoid underfull hboxes
}
}
```

Env `BVerbatim`

```
\AtBeginEnvironment{BVerbatim}
{%
\LWR@forcenewpage% instead of \preto
\LWR@atbeginverbatim{0}{bverbatim}%
```

```
}

\AfterEndEnvironment{BVerbatim}
{%
\LWR@afterendverbatim{0}%
}
```

End of the modifications to make at the end of the preamble:

```
} % \AfterEndPreamble
```

File 117 **lwarp-figcaps.sty**

§ 214 Package figcaps

(*Emulates or patches code by* Patrick W. Daly.)

Pkg figcaps Emulated.

for HTML output: Discard all options for lwarp-figcaps:

```
\LWR@ProvidesPackageDrop{figcaps}[1999/02/23]

\newcommand*{\figcapson}{}
\newcommand*{\figcapsoff}{}
\newcommand*{\printfigures}{}
\newcommand*{\figmarkon}{}
\newcommand*{\figmarkoff}{}
\def\figurecapname{Figure Captions}
\def\tablepagename{Tables}
\def\figurepagename{Figures}
```

File 118 **lwarp-figsize.sty**

§ 215 Package figsize

(*Emulates or patches code by* Anthony A. Tanbakuchi.)

Pkg figsize figsize is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{figsize}[2002/03/18]
```

Emulates a virtual 6×9 inch textsize.

```
\newlength{\figwidth}
\newlength{\figheight}

\newcommand{\SetFigLayout}[3][0]{%
\setlength{\figheight}{8in}%
```

```
\setlength{\figheight}{\figheight / #2}%
%
\setlength{\figwidth}{5.5in}%
\setlength{\figwidth}{\figwidth / #3}%
}
```

File 119 **lwarp-fitbox.sty**

§ 216 Package

fitbox

Pkg fitbox fitbox is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{fitbox}[2019/02/20]

\NewDocumentCommand{\fitbox}{s o m}{%
    \begin{BlockClass}{fitbox}
    #3
    \end{BlockClass}
}

\newcommand*{\fitboxset}[1]{}

\newdimen\fitboxnatheight
\newdimen\fitboxnatwidth

\newcommand\SetFitboxLayout[3][]{}
```

File 120 **lwarp-fix2col.sty**

§ 217 Package

fix2col

Pkg fix2col fix2col is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{fix2col}[2015/11/13]
```

File 121 **lwarp-fixme.sty**

§ 218 Package

fixme

(*Emulates or patches code by* Didier Verna.)

Pkg fixme fixme is patched for use by lwarp.

⚠ external layouts External layouts (`\fxloadlayouts`) are not supported.

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:

```
\LWR@ProvidesPackagePass{fixme}[2017/03/05]
```

Restore lwarp's version of `\@wrindex`, ignoring the fixme package's `target` option:

```
\let\@wrindex\LWR@wrindex
```

Float-related macros required by lwarp:

```
\newcommand{\ext@fixme}{lox}

\renewcommand{\l@fixme}[2]{\hypertocfloat{1}{fixme}{lox}{#1}{#2}}
```

Other modifications:

```
\def\FXFaceInlineHTMLStyle{font-weight:bold}

\renewcommand*\FXLayoutInline[3]{ %
\InlineClass[\FXFaceInlineHTMLStyle]{fixmeinline}%
    {\@fxtextstd{#1}{#2}{#3}}%
}

\def\FXFaceEnvHTMLStyle{font-weight:bold}

\renewcommand*\FXEnvLayoutPlainBegin[2]{%
\BlockClass[\FXFaceEnvHTMLStyle]{fixmebold}
\ignorespaces#2 \fxnotename{#1}: \ignorespaces}

\renewcommand*\FXEnvLayoutPlainEnd[2]{\endBlockClass}

\renewcommand*\FXEnvLayoutSignatureBegin[2]{%
\BlockClass[\FXFaceEnvHTMLStyle]{fixmebold}
\fxnotename{#1}: \ignorespaces}

\renewcommand*\FXEnvLayoutSignatureEnd[2]{\@fxsignature{#2}\endBlockClass}

\def\FXFaceSignatureHTMLStyle{font-style:italic}

\DeclareRobustCommand*\@fxsignature[1]{%
\ifthenelse{\equal{#1}{}}%
    {}%
    { -- {\InlineClass[\FXFaceSignatureHTMLStyle]{fixmesignature}{#1}}}%
}


\def\FXFaceTargetHTMLStyle{font-style:italic}

\renewcommand\FXTargetLayoutPlain[2]{%
    \InlineClass[\FXFaceTargetHTMLStyle]{fixmetarget}{#2}%
}
```

File 122 **lwarp-fixmetodonotes.sty**

§ 219 Package

fixmetodonotes

(*Emulates or patches code by* Gioele Barabucci.)

Pkg fixmetodonotes fixmetodonotes is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{fixmetodonotes}[2013/04/28]

\renewcommand{\NOTES@addtolist}[2]{%
    \refstepcounter{NOTES@note}%
%   \phantomsection% REMOVED
    \addcontentsline{notes}{NOTES@note}{%
        \protect\numberline{\theNOTES@note}{{#1}: {#2}}%
    }%
}

\renewcommand{\NOTES@marker}[2]{\fbox{%
    \textcolor{#2}{% WAS \color
        \textbf{#1}}%
    }}

\renewcommand{\NOTES@colorline}[2]{%
  \bgroup%
    \ULon{\LWR@backgroundcolor{#1}{#2}}%
}
```

File 123 **lwarp-flafter.sty**

§ 220 Package

flafter

Pkg flafter flafter is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{flafter}[2018/01/08]
\providecommand\fl@trace[1]{}
```

File 124 **lwarp-flippdf.sty**

§ 221 Package

flippdf

Pkg flippdf flippdf is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{flippdf}[2006/06/30]
```

```
\newcommand\FlipPDF{}
\newcommand\UnFlipPDF{}
```

File 125 **lwarp-float.sty**

§ 222 Package

float

(*Emulates or patches code by* Anselm Lingnau.)

Pkg float 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.

for HTML output:

```
\LWR@ProvidesPackageDrop{float}[2001/11/08]
```

\listof See section 74.2 for the \listof command.

\LWR@floatstyle The default float style:

```
\newcommand*{\LWR@floatstyle}{plain}
```

\newfloat {⟨*1: type*⟩} {⟨*2: placement*⟩} {⟨*3: ext*⟩} [⟨*4: within*⟩]

Emulates the \newfloat command from the float package.

"placement" is ignored.

```
\NewDocumentCommand{\newfloat}{m m m o}{%
\IfValueTF{#4}%
{\DeclareFloatingEnvironment[fileext=#3,within=#4]{#1}}%
{\DeclareFloatingEnvironment[fileext=#3]{#1}}%
```

Remember the float style:

```
\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.

```
\cslet{listof#1s}\relax%
\cslet{listof#1es}\relax%
```

Likesize, newfloat also creates \l@<type>, but float does not, so remove it here:

```
\cslet{l@#1}\relax%
}
```

\floatname {⟨*type*⟩} {⟨*name*⟩}

Sets the text name of the float, such as “Figure”.

```
\NewDocumentCommand{\floatname}{m +m}{%
    \SetupFloatingEnvironment{#1}{name=#2}%
}
```

\floatplacement {⟨*type*⟩} {⟨*placement*⟩}

Float placement is ignored.

```
\newcommand*{\floatplacement}[2]{%
    \SetupFloatingEnvironment{#1}{placement=#2}%
}
```

\floatstyle {⟨*style*⟩}

Remember the style for future floats:

```
\newcommand{\floatstyle}[1]{%
    \def\LWR@floatstyle{#1}%
}
```

\restylefloat * {⟨*type*⟩}

Remember the style for this float:

```
\NewDocumentCommand{\restylefloat}{s m}{%
    \csedef{LWR@floatstyle@#2}{\LWR@floatstyle}%
}
```

File 126 **lwarp-floatflt.sty**

§ 223 Package

floatflt

(*Emulates or patches code by* Mats Dahlgren.)

Pkg floatflt Emulated.

for HTML output: Discard all options for lwarp-floatflt:

```
\LWR@ProvidesPackageDrop{floatflt}[1997/07/16]
```

Env [⟨⟩] offset {⟨*type*⟩} {⟨*width*⟩} Borrowed from the lwarp version of keyfloat:

```
\NewDocumentEnvironment{KFLTfloatflt@marginfloat}{O{-1.2ex} m m}
{%
\setlength{\LWR@templengthone}{#3}%
\LWR@BlockClassWP{%
    float:right; %
    width:\LWR@printlength{\LWR@templengthone};  % extra space
    margin:10pt%
}{%
    width:\LWR@printlength{\LWR@templengthone}%
}%
```

```
{marginblock}%
\captionsetup{type=#2}%
}
{%
\endLWR@BlockClassWP%
}
```

Env floatingfigure [⟨*placement*⟩] {⟨*width*⟩}

```
\DeclareDocumentEnvironment{floatingfigure}{o m}
  {\begin{KFLTfloatflt@marginfloat}{figure}{#2}}
  {\end{KFLTfloatflt@marginfloat}}
```

Env floatingtable [⟨*placement*⟩]

```
\DeclareDocumentEnvironment{floatingtable}{o}
  {\begin{KFLTfloatflt@marginfloat}{table}{1.5in}}
  {\end{KFLTfloatflt@marginfloat}}
```

File 127 **lwarp-floatpag.sty**

§ 224 Package

floatpag

(*Emulates or patches code by* Vytas Statulevičius and Sigitas Tolušis.)

Pkg floatpag Emulated.

for HTML output: Discard all options for lwarp-floatpag:

```
\LWR@ProvidesPackageDrop{floatpag}[2012/05/29]

\newcommand*{\floatpagestyle}[1]{}
\newcommand*{\rotfloatpagestyle}[1]{}
\newcommand*{\thisfloatpagestyle}[1]{}
```

File 128 **lwarp-floatrow.sty**

§ 225 Package

floatrow

(*Emulates or patches code by* Olga Lapko.)

Pkg floatrow floatrow is emulated.

for HTML output:

```
\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 9.10.

⚠ 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:

```
\newbool{LWR@subcaptionloaded}

\AtBeginDocument{
\@ifpackageloaded{subcaption}
{\booltrue{LWR@subcaptionloaded}}
{\boolfalse{LWR@subcaptionloaded}}
}
```

\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.

```
\NewDocumentCommand{\floatbox}{o m o o o +m +m}{%
\ifbool{LWR@subcaptionloaded}%
{% subcaption
```

For subcaption:

```
    \ifbool{LWR@insubfloatrow}%
    {% subcaption in a subfloatrow
```

subfigure and subtable environments take width as an argument.

```
        \IfValueTF{#3}%
        {\@nameuse{sub#2}{#3}}%
        {\@nameuse{sub#2}{\linewidth}}%
    }% subcaption in a subfloatrow
    {% subcaption not in subfloatrow
```

figure and table environments do not take a width argument.

```
        \@nameuse{#2}%
    }% subcaption not in subfloatrow
    #6

    #7
```

End the environments:

```
    \ifbool{LWR@insubfloatrow}%
    {\@nameuse{endsub#2}}%
    {\@nameuse{end#2}}%
}% subcaption
{% assume subfig
```

For subfig:

```
\ifbool{LWR@insubfloatrow}%
{% 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.

```
    \begingroup
    \let\caption\@firstofone
    \subfloat[#6]{#7}
    \endgroup
}% subfig in a subfloatrow
{% subfig package, but not a subfig
```

`figure` and `table` are environments:

```
\@nameuse{#2}
#6

#7
\@nameuse{end#2}
}% subfig package, but not a subfig
}% assume subfig
}
```

Not used:

```
\newcommand*{\nocapbeside}{}
\newcommand*{\capbeside}{}
\newcommand*{\captop}{}
\newlength{\FBwidth}
\setlength{\FBwidth}{.3\linewidth}
\newlength{\FBheight}
\setlength{\FBheight}{2in}
\newcommand*{\useFCwidth}{}
\newcommand{\floatsetup}[2][]{}
\newcommand{\thisfloatsetup}[1]{}
\newcommand{\clearfloatsetup}[1]{}
\newcommand*{\killfloatstyle}{}
```

`\newfloatcommand` {⟨*1 command*⟩} {⟨*2 captype*⟩} [⟨*3 preamble*⟩] [⟨*4 default width*⟩]

Preamble and default width are ignored.

```
\NewDocumentCommand{\newfloatcommand}{m m o o}{%
\@namedef{#1}{
```

```
\floatbox{#2}
}
}
```

\renewfloatcommand {⟨*1 command*⟩} {⟨*2 captype*⟩} [⟨*3 preamble*⟩] [⟨*4 default width*⟩]

Preamble and default width are ignored.

```
\NewDocumentCommand{\renewfloatcommand}{m m o o}{%
\@namedef{#1}{%
\floatbox{#2}
}
}
```

\ffigbox [⟨*width*⟩] [⟨*height*⟩] [⟨*vposn*⟩] {⟨*caption commands*⟩} {⟨*contents*⟩}

```
\newfloatcommand{ffigbox}{figure}[\nocapbeside][]
```

\ttabbox [⟨*width*⟩] [⟨*height*⟩] [⟨*vposn*⟩] {⟨*caption commands*⟩} {⟨*contents*⟩}

```
\newfloatcommand{ttabbox}{table}[\captop][\FBwidth]
```

\fcapside [⟨*width*⟩] [⟨*height*⟩] [⟨*vposn*⟩] {⟨*caption commands*⟩} {⟨*contents*⟩}

```
\newfloatcommand{fcapside}{figure}[\capbeside][]
```

Env floatrow [⟨*numfloats*⟩]

The row of floats is placed into a of class floatrow.

```
\newenvironment*{floatrow}[1][2]
{
\LWR@forcenewpage
\BlockClass{floatrow}
```

While inside the floatrow, divide the \linewidth by the number of floats.

```
\booltrue{LWR@infloatrow}
\setlength{\linewidth}{6in/#1}
}
{
\boolfalse{LWR@infloatrow}
\endBlockClass
}
```

Keys for \DeclareNewFloatType:

```
\newcommand*{\LWR@frowkeyplacement}{}
\newcommand*{\LWR@frowkeyname}{}
\newcommand*{\LWR@frowkeyfileext}{}
\newcommand*{\LWR@frowkeywithin}{}
\newcommand*{\LWR@frowkeycapstyle}{}

\define@key{frowkeys}{placement}{}%
\define@key{frowkeys}{name}{\renewcommand{\LWR@frowkeyname}{#1}}%
\define@key{frowkeys}{fileext}{\renewcommand{\LWR@frowkeyfileext}{#1}}%
```

```
\define@key{frowkeys}{within}{\renewcommand{\LWR@frowkeywithin}{#1}}%
\define@key{frowkeys}{relatedcapstyle}{}%
```

\DeclareNewFloatType {⟨*type*⟩} {⟨*options*⟩}

Use `\listof{type}{Title}` to print a list of the floats.

```
\newcommand*{\DeclareNewFloatType}[2]{%
```

Reset key values:

```
\renewcommand*{\LWR@frowkeyplacement}{}%
\renewcommand*{\LWR@frowkeyname}{}%
\renewcommand*{\LWR@frowkeyfileext}{}%
\renewcommand*{\LWR@frowkeywithin}{}%
\renewcommand*{\LWR@frowkeycapstyle}{}%
```

Read new key values:

```
\LWR@traceinfo{about to setkeys frowkeys}%
\setkeys{frowkeys}{#2}%
\LWR@traceinfo{finished setkeys frowkeys}%
```

Create a new float with optional [within]:

```
\ifthenelse{\equal{\LWR@frowkeywithin}{}}%
{%
    \LWR@traceinfo{about to newfloat #1 \LWR@frowkeyplacement\ %
        \LWR@frowkeyfileext}%
    \newfloat{#1}{\LWR@frowkeyplacement}{\LWR@frowkeyfileext}%
}%
{%
    \LWR@traceinfo{about to newfloat #1\ \LWR@frowkeyplacement\ %
        \LWR@frowkeyfileext\ \LWR@frowkeywithin}%
    \newfloat{#1}{\LWR@frowkeyplacement}%
    {\LWR@frowkeyfileext}[\LWR@frowkeywithin]%
    \LWR@traceinfo{finished newfloat #1}%
}%
```

Rename the float if a name was given:

```
\ifthenelse{\equal{\LWR@frowkeyname}{}}%
    {}%
    {\floatname{#1}{\LWR@frowkeyname}}%
}
```

Not used:

```
\newcommand{\buildFBBOX}[2]{}
\newcommand*{\CenterFloatBoxes}{}
\newcommand*{\TopFloatBoxes}{}
\newcommand*{\BottomFloatBoxes}{}
\newcommand*{\PlainFloatBoxes}{}

\newcommand{\capsubrowsettings}{}

\NewDocumentCommand{\RawFloats}{o o}{}
```

\RawCaption {⟨*text*⟩}

To be used inside a minipage or parbox.

```
\newcommand{\RawCaption}[1]{#1}
```

\floatfoot {⟨*text*⟩}

Places additional text inside a float, inside a css of class floatfoot.

```
\NewDocumentCommand{\floatfoot}{s +m}{%
\begin{BlockClass}{floatfoot}
#2
\end{BlockClass}
}
```

Used to compute \linewidth.

```
\newbool{LWR@insubfloatrow}
\boolfalse{LWR@insubfloatrow}
```

Env subfloatrow [⟨*num_floats*⟩]

```
\newenvironment*{subfloatrow}[1][2]
{
```

The row of floats is placed into a of class floatrow:

```
\LWR@forcenewpage
\BlockClass{floatrow}
```

While inside the floatrow, LWR@insubfloatrow is set true, which tells \floatbox to use \subfigure or \subtable.

```
\begingroup
\booltrue{LWR@insubfloatrow}
}
{
\endgroup
\endBlockClass
\boolfalse{LWR@insubfloatrow}
}
```

File 129 **lwarp-fltrace.sty**

§ 226 Package

fltrace

Pkg fltrace fltrace is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{fltrace}[2018/01/08]
```

```
\def\tracefloats{}
\def\tracefloatsoff{}
\def\tracefloatvals{}
```

File 130 **lwarp-flushend.sty**

§ 227 Package flushend

(*Emulates or patches code by* Sigitas Tolušis.)

Pkg flushend Emulated.

for HTML output: Discard all options for lwarp-flushend:

```
\LWR@ProvidesPackageDrop{flushend}[2017/03/27]

\newcommand*{\flushend}{}
\newcommand*{\raggedend}{}
\newcommand*{\flushcolsend}{}
\newcommand*{\raggedcolsend}{}
\newcommand*{\atColsBreak}[1]{}
\newcommand*{\atColsEnd}[1]{}
\newcommand*{\showcolsendrule}{}
```

File 131 **lwarp-fnbreak.sty**

§ 228 Package fnbreak

Pkg fnbreak fnbreak is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{fnbreak}[2012/01/01]

\newcommand*{\fnbreakverbose}{}
\newcommand*{\fnbreaknonverbose}{}
\newcommand*{\fnbreaklabel}{}
\newcommand*{\fnbreaknolabel}{}
```

File 132 **lwarp-fncychap.sty**

§ 229 Package fncychap

(*Emulates or patches code by* Ulf A. Lindgren.)

Pkg fncychap fncychap is emulated.

for HTML output: Discard all options for lwarp-fncychap:

```
\LWR@ProvidesPackageDrop{fncychap}[2007/07/30]
```

```
\def\mghrulefill#1{}
\def\ChNameLowerCase{}
\def\ChNameUpperCase{}
\def\ChNameAsIs{}
\def\ChTitleLowerCase{}
\def\ChTitleUpperCase{}
\def\ChTitleAsIs{}
\newcommand{\ChRuleWidth}[1]{}
\newcommand{\ChNameVar}[1]{}
\newcommand{\ChNumVar}[1]{}
\newcommand{\ChTitleVar}[1]{}
\newcommand{\TheAlphaChapter}{}
\newcommand{\DOCH}{}
\newcommand{\DOTI}[1]{}
\newcommand{\DOTIS}[1]{}
\newlength{\mylen}
\newlength{\myhi}
\newlength{\px}
\newlength{\py}
\newlength{\pyy}
\newlength{\pxx}
\newlength{\RW}
\newcommand{\FmN}[1]{#1}
\newcommand{\FmTi}[1]{#1}
```

File 133 **lwarp-fnlineno.sty**

§ 230 Package fnlineno

Pkg fnlineno fnlineno is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{fnlineno}[2011/01/07]
```

File 134 **lwarp-fnpos.sty**

§ 231 Package fnpos

(*Emulates or patches code by* Hiroshi Nakashima.)

Pkg fnpos fnpos is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{fnpos}[1999/07/14]
```

```
\newcommand*{\makeFNbottom}{}
\newcommand*{\makeFNmid}{}
\newcommand*{\makeFNbelow}{}
\newcommand*{\makeFNabove}{}
```

File 135 **lwarp-fontawesome.sty**

§ 232 Package

fontawesome

(*Emulates or patches code by* Xavier Danaux.)

Pkg fontawesome 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 pdfLaTeX, *poppler* may issue a syntax warning regarding parsing a ligature component. XeLaTeX 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 pdfLaTeX, the `alt` tag includes the icon (symbol) number. For XeLaTeX and LuaLaTeX, the `alt` tag is generic.

for HTML output:

```
\LWR@ProvidesPackagePass{fontawesome}[2016/05/15]

\LetLtxMacro\LWR@orig@symbol\symbol

\ifxetexorluatex

\newfontfamily{\LWR@orig@FA}{FontAwesome}

\newcommand*{\LWR@fontawesome@xelatex@symbol}[1]{%
\begin{lateximage}*[(icon)][fontawesomexetex#1]%
\begingroup%
\LWR@orig@FA%
\LWR@orig@symbol{#1}%
\endgroup%
\end{lateximage}%
}

\RenewDocumentCommand{\FA}{}{%
    \LetLtxMacro\symbol\LWR@fontawesome@xelatex@symbol%
}

\else

\newcommand*{\LWR@fontawesome@symbolone}[1]{%
\begin{lateximage}*[(icon #1)][fontawesomeone#1]%
\begingroup%
\fontencoding{U}\fontfamily{fontawesomeone}\selectfont%
\LWR@orig@symbol{#1}%
\endgroup%
\end{lateximage}%
}

```

```
\newcommand*{\LWR@fontawesome@symboltwo}[1]{%
\begin{lateximage}*[(icon #1)][fontawesometwo#1]%
\begingroup%
\fontencoding{U}\fontfamily{fontawesometwo}\selectfont%
\LWR@orig@symbol{#1}%
\endgroup%
\end{lateximage}%
}

\newcommand*{\LWR@fontawesome@symbolthree}[1]{%
\begin{lateximage}*[(icon #1)][fontawesomethree#1]%
\begingroup%
\fontencoding{U}\fontfamily{fontawesomeonethree}\selectfont%
\LWR@orig@symbol{#1}%
\endgroup%
\end{lateximage}%
}

\renewrobustcmd\FAone{%
    \LetLtxMacro\symbol\LWR@fontawesome@symbolone%
}

\renewrobustcmd\FAtwo{%
    \LetLtxMacro\symbol\LWR@fontawesome@symboltwo%
}

\renewrobustcmd\FAthree{%
    \LetLtxMacro\symbol\LWR@fontawesome@symbolthree%
}
\fi
```

File 136 **lwarp-fontawesome5.sty**

§ 233 Package fontawesome5

(*Emulates or patches code by* Marcel Krüger.)

Pkg fontawesome5 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:

```
\LWR@ProvidesPackagePass{fontawesome5}[2018/07/27]

\ExplSyntaxOn
\cs_set:Nn\fontawesome_use_icon:nn{
  \cs_if_exist:cTF{c__fontawesome_slot_#2_tl}{
    \begin{lateximage}*[(#2)][#1]
    \exp_last_unbraced:Nv
      \__fontawesome_icon_at:nnnn
      {c__fontawesome_slot_#2_tl}
        {#1}{#2}
```

```
    \end{lateximage}
  }{
    \msg_error:nnxx{fontawesome5}{icon-not-found}{#2}{#1}
  }
}
\ExplSyntaxOff
```

File 137 **lwarp-fontenc.sty**

§ 234 Package fontenc

Pkg fontenc

If using pdfLaTeX, 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:

```
\LWR@ProvidesPackagePass{fontenc}[2017/04/05]
```

File 138 **lwarp-footmisc.sty**

§ 235 Package footmisc

(*Emulates or patches code by* Robin Fairbairns.)

Pkg footmisc

footmisc is emulated.

lwarp incidentally happens to emulate the `stable` option.

```
\LWR@ProvidesPackageDrop{footmisc}[2011/06/06]
```

Some nullified commands:

```
\newcommand{\footnotelayout}{}
\newcommand{\setfnsymbol}[1]{}
\NewDocumentCommand{\DefineFNsymbols}{s m o m}{}

\newdimen\footnotemargin
\footnotemargin1.8em\relax

\newcommand*\hangfootparskip{0.5\baselineskip}
\newcommand*\hangfootparindent{0em}%

\let\pagefootnoterule\footnoterule
\let\mpfootnoterule\footnoterule
\def\splitfootnoterule{\kern-3\p@ \hrule \kern2.6\p@}

\providecommand*{\multiplefootnotemarker}{3sp}
\providecommand*{\multfootsep}{,}
```

Using cleveref:

```
\providecommand*{\footref}[1]{\labelcref{#1}}
```

The following work as-is:

```
\newcommand\mpfootnotemark{%
  \@ifnextchar[%
    \@xmpfootnotemark%
    {%
      \stepcounter\@mpfn%
      \protected@xdef\@thefnmark{\thempfn}%
      \@footnotemark%
    }%
}
\def\@xmpfootnotemark[#1]{%
  \begingroup%
    \csname c@\@mpfn\endcsname #1\relax%
    \unrestored@protected@xdef\@thefnmark{\thempfn}%
  \endgroup%
  \@footnotemark%
}
```

File 139 **lwarp-footnote.sty**

§ 236 Package

footnote

(*Emulates or patches code by* Mark Wooding.)

Pkg footnote footnote is used with minor patches.

for HTML output:

```
\LWR@ProvidesPackagePass{footnote}[1997/01/28]
```

Removed print-version formatting:

```
\def\fn@startnote{%
%   \@parboxrestore%
  \protected@edef\@currentlabel{\csname p@\@mpfn\endcsname\@thefnmark}%
%   \color@begingroup% *** conflicts with lwarp
}

% \let\fn@endnote\color@endgroup% *** conflicts with lwarp
\def\fn@endnote{%
\LWR@htmltagc{/\LWR@tagregularparagraph}%
\LWR@orignewline%
}
```

Removed print-version formatting:

```
\def\fn@startfntext{%
  \setbox\z@\vbox\bgroup%
    \fn@startnote%
```

```
    \fn@prefntext%
    \ignorespaces%
}
```

Removed print-version formatting, added closing paragraph tag:

```
\def\fn@endfntext{%
    \LWR@htmltagc{/\LWR@tagregularparagraph}%
    \LWR@orignewline%
    \fn@postfntext%
  \egroup%
  \begingroup%
    \let\@makefntext\@empty%
    \let\@finalstrut\@gobble%
    \LetLtxMacro\rule\@gobbletwo% *8* also the optional argument?
    \@footnotetext{\unvbox\z@}%
  \endgroup%
}
```

These have been redefined, so re-`\let` them again:

```
\let\endfootnote\fn@endfntext
\let\endfootnotetext\endfootnote
```

File 140 **lwarp-footnotehyper.sty**

§ 237 Package

footnotehyper

Pkg footnotehyper

footnotehyper is a hyperref-safe version of footnote. For lwarp, footnotehyper is emulated.

for HTML output:

Discard all options for lwarp-footnotehyper:

```
\RequirePackage{footnote}
\LWR@ProvidesPackageDrop{footnotehyper}[2018/01/23]
```

File 141 **lwarp-footnoterange.sty**

§ 238 Package

footnoterange

(*Emulates or patches code by* H.-Martin Münch.)

Pkg footnoterange

footnoterange is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{footnoterange}[2012/02/17]
```

```
\csletcs{footnoterange}{footnoterange*}
\csletcs{endfootnoterange}{endfootnoterange*}
```

File 142 **lwarp-footnpag.sty**

§ 239 Package footnpag

Pkg footnpag footnpag is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{footnpag}
```

File 143 **lwarp-foreign.sty**

§ 240 Package foreign

(*Emulates or patches code by* Philip G. Ratcliffe.)

Pkg foreign foreign is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{foreign}[2012/09/25]
\renewcommand\foreignabbrfont{\emph}
```

File 144 **lwarp-forest.sty**

§ 241 Package forest

(*Emulates or patches code by* Sašo Živanović.)

Pkg forest 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:

```
\LWR@ProvidesPackagePass{forest}[2017/07/14]

\BeforeBeginEnvironment{forest}{\begin{lateximage}[forest]}

\AfterEndEnvironment{forest}{\end{lateximage}}

\RenewDocumentCommand{\Forest}{s D(){} m}{%
  \forest@config{#2}%
  \IfBooleanTF{#1}{%
        \PackageError{lwarp-forest}%
        {Starred \Forest is not supported}%
     {Lwarp uses an environment for images, but \Forest* cannot work in an environment.}%
        \let\forest@next\forest@env%
    }{\let\forest@next\forest@group@env}%
```

```
    \begin{lateximage}[forest]%     lwarp
  \forest@next{#3}%
    \end{lateximage}%               lwarp
}
```

File 145 **lwarp-framed.sty**

§ 242 Package

framed

(*Emulates or patches code by* Donald Arseneau.)

Pkg framed framed is supported and patched by lwarp.

for HTML output: Accept all options for lwarp-framed:

```
\LWR@ProvidesPackagePass{framed}[2011/10/22]
\RequirePackage{xcolor}% for \convertcolorspec


\renewenvironment{framed}{%
\LWR@forcenewpage
\BlockClass{framed}%
}
{\endBlockClass}

\renewenvironment{oframed}{%
\LWR@forcenewpage
\BlockClass{framed}%
}
{\endBlockClass}


\renewenvironment{shaded}{%
\convertcolorspec{named}{shadecolor}{HTML}\LWR@tempcolor%
\LWR@forcenewpage
\BlockClass[background: \LWR@origpound\LWR@tempcolor]{shaded}%
}
{\endBlockClass}

\renewenvironment{shaded*}{%
\convertcolorspec{named}{shadecolor}{HTML}\LWR@tempcolor%
\LWR@forcenewpage
\BlockClass[background: \LWR@origpound\LWR@tempcolor]{shaded}%
}
{\endBlockClass}


\renewenvironment{leftbar}{%
\LWR@forcenewpage
    \BlockClass{framedleftbar}
  \def\FrameCommand{}%
  \MakeFramed {}
}%
```

```
 {\endMakeFramed\endBlockClass}


\renewenvironment{snugshade}{%
\convertcolorspec{named}{shadecolor}{HTML}\LWR@tempcolor%
\LWR@forcenewpage
\BlockClass[background: \LWR@origpound\LWR@tempcolor]{snugframed}%
}
{\endBlockClass}

\renewenvironment{snugshade*}{%
\convertcolorspec{named}{shadecolor}{HTML}\LWR@tempcolor%
\LWR@forcenewpage
\BlockClass[background: \LWR@origpound\LWR@tempcolor]{snugframed}%
}
{\endBlockClass}

\let\oframed\framed
\let\endoframed\endframed


\RenewEnviron{titled-frame}[1]{%
\CustomFBox{#1}{}{0pt}{0pt}{0pt}{0pt}{\BODY}
}
```

\CustomFBox {⟨*toptitle*⟩} {⟨*botttitle*⟩} {⟨*thicknesstop*⟩} {⟨*bottom*⟩} {⟨*left*⟩} {⟨*right*⟩} {⟨*text contents*⟩}

```
\renewcommand{\CustomFBox}[7]{%
\convertcolorspec{named}{TFFrameColor}{HTML}\LWR@tempcolor%
\LWR@forcenewpage
\begin{BlockClass}[border: 3px solid \LWR@origpound\LWR@tempcolor]{framed}%
\ifthenelse{\isempty{#1}}{}{% not empty
    \begin{BlockClass}[background: \LWR@origpound\LWR@tempcolor]{framedtitle}%
    \textcolor{TFTitleColor}{\textbf{#1}}%
    \end{BlockClass}
}% not empty

#7

\ifthenelse{\isempty{#2}}{}{% not empty
    \convertcolorspec{named}{TFFrameColor}{HTML}\LWR@tempcolor%
    \begin{BlockClass}[background: \LWR@origpound\LWR@tempcolor]{framedtitle}%
    \textcolor{TFTitleColor}{\textbf{#2}}%
    \end{BlockClass}
}% not empty
\end{BlockClass}
}
```

\TitleBarFrame [⟨*marker*⟩] {⟨*title*⟩} {⟨*contents*⟩}

```
\renewcommand\TitleBarFrame[3][]{
\CustomFBox
    {#2}{}%
    \fboxrule\fboxrule\fboxrule\fboxrule
```

```
    {#3}%
}

\renewcommand{\TF@Title}[1]{#1}
```

MakeFramed {⟨*settings*⟩}

```
\let\MakeFramed\relax
\let\endMakeFramed\relax

\NewEnviron{MakeFramed}[1]{%
\FrameCommand{\begin{minipage}{\linewidth}\BODY\end{minipage}}%
}
```

\fb@put@frame {⟨*frame cmd no split*⟩} {⟨*frame cmd split*⟩}

```
\renewcommand*{\fb@put@frame}[2]{%
\relax%
\@tempboxa%
}
```

File 146 **lwarp-ftnright.sty**

§ 243 Package

ftnright

Pkg ftnright ftnright is ignored.

for HTML output: Discard all options for lwarp-ftnright:

```
\LWR@ProvidesPackageDrop{ftnright}[2014/10/28]
```

File 147 **lwarp-fullminipage.sty**

§ 244 Package

fullminipage

Pkg fullminipage fullminipage is nullified.

for HTML output:

```
\LWR@ProvidesPackageDrop{fullminipage}[2014/07/06]

\newenvironment{fullminipage}[1][]{}{}
```

File 148 **lwarp-fullpage.sty**

§ 245 Package

fullpage

Pkg fullpage fullpage is ignored.

for HTML output: Discard all options for lwarp-fullpage:

```
\LWR@ProvidesPackageDrop{fullpage}[1994/06/01]
```

File 149 **lwarp-fullwidth.sty**

§ 246 Package fullwidth

(*Emulates or patches code by* Marco Daniel.)

Pkg fullwidth fullwidth is emulated.

A minipage is used, of no HTML width.

for HTML output:

```
\LWR@ProvidesPackageDrop{fullwidth}[2011/11/18]

\newenvironment*{fullwidth}[1][]{%
\minipagefullwidth%
\minipage{\linewidth}%
}
{%
\endminipage%
}
```

File 150 **lwarp-fwlw.sty**

§ 247 Package fwlw

Pkg fwlw fwlw is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{fwlw}

\newbox\FirstWordBox      \global\setbox\FirstWordBox\hbox{}
\newbox\NextWordBox       \global\setbox\NextWordBox\hbox{}
\newbox\LastWordBox       \global\setbox\LastWordBox\hbox{}
\def\ps@fwlwhead{}
\def\ps@NextWordFoot{}
```

File 151 **lwarp-gentombow.sty**

§ 248 Package gentombow

Pkg gentombow gentombow is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{gentombow}[2018/05/17]
```

```
\newcommand{\settombowbanner}[1]{}
\newcommand{\settombowbannerfont}[1]{}
\newcommand{\settombowwidth}[1]{}
\newcommand{\settombowbleed}[1]{}
\newcommand{\settombowcolor}[1]{}
```

File 152 **lwarp-geometry.sty**

§ 249 Package geometry

(*Emulates or patches code by* Hideo Umeki.)

Pkg geometry 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:

```
\LWR@ProvidesPackageDrop{geometry}[2018/04/16]
```

```
\renewcommand*{\geometry}[1]{}
\renewcommand*{\newgeometry}[1]{}
\renewcommand*{\restoregeometry}{}
\renewcommand*{\savegeometry}[1]{}
\renewcommand*{\loadgeometry}[1]{}
```

File 153 **lwarp-gloss.sty**

§ 250 Package gloss

(*Emulates or patches code by* Jose Luis Díiaz, Javier Bezos.)

Pkg gloss gloss is patched for use by lwarp.

To process the HTML glossary:

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

for HTML output:

```
\LWR@ProvidesPackagePass{gloss}[2002/07/26]
```

```
\xpatchcmd{\gls@gloss@iii}
    {\thepage}
    {\theLWR@previousautopagelabel}
    {}
    {\LWR@patcherror{gloss}{gls@gloss@iii}}

\def\gls@page@i#1#2{%
  \endgroup
  \global\@namedef{glsp@#1}{\nameref{autopage-#2}}}
```

File 154 **lwarp-glossaries.sty**

§ 251 Package

glossaries

(*Emulates or patches code by* Nicola L.C. Talbot.)

Pkg glossaries

processing glossaries

Opt GlossaryCmd

Default: makeglossaries

Opt [lwarpmk] printglossary

Opt [lwarpmk] htmlglossary

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.

for HTML output:

```
\PassOptionsToPackage{xindy}{glossaries}

\LWR@ProvidesPackagePass{glossaries}[2018/07/23]

\setupglossaries{nonumberlist}
\setglossarystyle{index}
```

Patched to fix TOC pointing to the previous page:

```
\renewcommand*{\@p@glossarysection}[2]{%
  \glsclearpage
  \phantomsection
  \ifdefempty\@@glossarysecstar
  {%
    \csname\@@glossarysec\endcsname{#2}%
  }%
  {%
```

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:

```
      \csname\@@glossarysec\endcsname*{#2}%
    \@gls@toc{#1}{\@@glossarysec}% Moved after the previous line.
  }%
  \@@glossaryseclabel
}
```

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.

```
\newcommand*{\LWR@comp@glossaryname}{\translate{Glossary}}

\ifdefstrequal{\glossaryname}{\LWR@comp@glossaryname}{
    \translatelet\LWR@translatetemp{Glossary}
    \edef\glossaryname{\LWR@translatetemp}
}{}

\newcommand*{\LWR@comp@acronymname}{\translate{Acronym}}

\ifdefstrequal{\acronymname}{\LWR@comp@acronymname}{
    \translatelet\LWR@translatetemp{Acronym}
    \edef\acronymname{\LWR@translatetemp}
```

```
}{}

\newcommand*{\LWR@comp@glssymbolsgroupname}{\translate{Symbols (glossaries)}}

\ifdefstrequal{\glssymbolsgroupname}{\LWR@comp@glssymbolsgroupname}{
    \translatelet\LWR@translatetemp{Symbols (glossaries)}
    \edef\glssymbolsgroupname{\LWR@translatetemp}
}{}

\newcommand*{\LWR@comp@glsnumbersgroupname}{\translate{Numbers (glossaries)}}

\ifdefstrequal{\glsnumbersgroupname}{\LWR@comp@glsnumbersgroupname}{
    \translatelet\LWR@translatetemp{Numbers (glossaries)}
    \edef\glsnumbersgroupname{\LWR@translatetemp}
}{}
```

File 155 **lwarp-gmeometric.sty**

§ 252 Package

gmeometric

Pkg gmeometric gmeometric is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{gmeometric}[2008/11/22]
```

File 156 **lwarp-graphics.sty**

§ 253 Package

graphics

(*Emulates or patches code by* D. P. Carlisle.)

Pkg graphics graphics is emulated.

for HTML output:

```
\LWR@ProvidesPackagePass{graphics}[2017/06/25]
```

§ 253.1 Graphics extensions

\DeclareGraphicsExtensions {⟨*list*⟩}

\AtBeginDocument allow svg files instead of pdf:

```
\AtBeginDocument{
\DeclareGraphicsExtensions{.svg,.SVG,.gif,.GIF,.png,.PNG,.jpg,.JPG,.jpeg,.JPEG}
\DeclareGraphicsRule{.svg}{svg}{.svg}{}
\DeclareGraphicsRule{.SVG}{svg}{.SVG}{}
}
```

Inside a lateximage, allow pdf instead of svg:

```
\ifpdf
```

```
\appto\LWR@restoreorigformatting{%
\DeclareGraphicsExtensions{.pdf,.PDF,.gif,.GIF,.png,.PNG,.jpg,.JPG,.jpeg,.JPEG}%
}
\else% \ifpdf
        \ifXeTeX
\appto\LWR@restoreorigformatting{%
\DeclareGraphicsExtensions{.pdf,.PDF,.gif,.GIF,.png,.PNG,.jpg,.JPG,.jpeg,.JPEG}%
}
        \else
\appto\LWR@restoreorigformatting{%
\DeclareGraphicsExtensions{.eps,.EPS,.gif,.GIF,.png,.PNG,.jpg,.JPG,.jpeg,.JPEG}%
}
        \fi
\fi
```

§ 253.2 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.

```
\newlength{\LWR@igwidth}
\newlength{\LWR@igheight}
\newcommand*{\LWR@igwidthstyle}{}
\newcommand*{\LWR@igheightstyle}{}
\newcommand*{\LWR@igorigin}{}
\newcommand*{\LWR@igangle}{}
\newcommand*{\LWR@igxscale}{1}
\newcommand*{\LWR@igyscale}{1}
\newcommand*{\LWR@igclass}{inlineimage}

\newcommand*{\LWR@igalt}{(image)}
```

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:

```
\define@key{igraph}{width}{%
\setlength{\LWR@igwidth}{#1}%
\ifthenelse{\lengthtest{\LWR@igwidth > 0pt}}%
{%
```

Default to use the converted fixed length given:

```
    \renewcommand*{\LWR@igwidthstyle}{width:\LWR@printlength{\LWR@igwidth}}%
```

If ex or em dimensions were given, use those instead:

```
    \IfEndWith{#1}{ex}%
    {\renewcommand*{\LWR@igwidthstyle}{width:#1}}% yes ex
    {}% not ex
    \IfEndWith{#1}{em}%
    {\renewcommand*{\LWR@igwidthstyle}{width:#1}}% yes em
    {}% not em
    \IfEndWith{#1}{\%}%
    {\renewcommand*{\LWR@igwidthstyle}{width:#1}}% yes percent
    {}% not percent
    \IfEndWith{#1}{px}%
    {\renewcommand*{\LWR@igwidthstyle}{width:#1}}% yes px
    {}% not px
}{}% end of length > 0pt
}
```

If an optional height was given, set an HTML style:

```
\define@key{igraph}{height}{%
\setlength{\LWR@igheight}{#1}%
\ifthenelse{\lengthtest{\LWR@igheight > 0pt}}%
{%
```

Default to use the converted fixed length given:

```
    \renewcommand*{\LWR@igheightstyle}{%
    height:\LWR@printlength{\LWR@igheight} % extra space
    }%
```

If ex or em dimensions were given, use those instead:

```
    \IfEndWith{#1}{ex}%
    {\renewcommand*{\LWR@igheightstyle}{height:#1}}% yes ex
    {}% not ex
    \IfEndWith{#1}{em}%
    {\renewcommand*{\LWR@igheightstyle}{height:#1}}% yes em
    {}% not em
    \IfEndWith{#1}{\%}%
    {\renewcommand*{\LWR@igheightstyle}{height:#1}}% yes percent
    {}% not percent
    \IfEndWith{#1}{px}%
    {\renewcommand*{\LWR@igheightstyle}{height:#1}}% yes px
    {}% not px
}{}% end of length > 0pt
}
```

Handle origin key:

```
\define@key{igraph}{origin}[c]{%
    \renewcommand*{\LWR@igorigin}{#1}%
}
```

Handle angle key:

```
\define@key{igraph}{angle}{\renewcommand*{\LWR@igangle}{#1}}
```

Handle class key:

```
\define@key{igraph}{class}{\renewcommand*{\LWR@igclass}{#1}}
```

Handle alt key:

```
\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.

```
\define@key{igraph}{scale}{%
    \renewcommand*{\LWR@igxscale}{#1}%
    \renewcommand*{\LWR@igyscale}{#1}%
}
```

Numerous ignored keys:

```
\define@key{igraph}{bb}{}
\define@key{igraph}{bbllx}{}
\define@key{igraph}{bblly}{}
\define@key{igraph}{bburx}{}
\define@key{igraph}{bbury}{}
\define@key{igraph}{natwidth}{}
\define@key{igraph}{natheight}{}
\define@key{igraph}{hiresbb}[true]{}
\define@key{igraph}{viewport}{}
\define@key{igraph}{trim}{}
\define@key{igraph}{totalheight}{}
\define@key{igraph}{keepaspectratio}[true]{}
\define@key{igraph}{clip}[true]{}
\define@key{igraph}{draft}[true]{}
\define@key{igraph}{type}{}
\define@key{igraph}{ext}{}
\define@key{igraph}{read}{}
\define@key{igraph}{command}{}
```

New in v1.1a:

```
\define@key{igraph}{quite}{}
\define@key{igraph}{page}{}
\define@key{igraph}{pagebox}{}
\define@key{igraph}{interpolate}[true]{}
```

New in v1.1b:

```
\define@key{igraph}{decodearray}{}
```

§ 253.3 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.

```
\newcommand*{\LWR@rotstyle}[2]{%
 #1transform:rotate(-#2deg);
}
```

`\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.

```
\newcommand*{\LWR@scalestyle}[3]{%
 #1transform:scale(#2,#3);
}
```

§ 253.4 `\includegraphics`

Bool `LWR@infloatrow` Used to compute `\linewidth`.

```
\newbool{LWR@infloatrow}
\boolfalse{LWR@infloatrow}
```

`\LWR@opacity` For HTML, used only for `\includegraphics`.

`\LWR@opacity` may be set by the transparent package.

```
\def\LWR@opacity{1}
```

`\LWR@imagesizebox` Used to determine the actual image size if needed.

```
\newsavebox{\LWR@imagesizebox}
```

`\LWR@HTML@Gin@setfile` {⟨*w*⟩} {⟨*h*⟩} {⟨*filename*⟩} Sets the parsed filename for HTML output.

```
\newcommand*{\LWR@HTML@Gin@setfile}[3]{%
    \xdef\LWR@parsedfilename{#3}%
}
```

Key [Gin] `class` CSS class for the image.

Define the new class key for the print-mode version of `\includegraphics`, which is enabled inside a `lateximage`.

```
\AtBeginDocument{
\define@key{Gin}{class}{}
\define@key{Gin}{alt}{}
}
```

`\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.

```
\AtBeginDocument{
\@ifpackageloaded{epstopdf}
{
    \newcommand*{\LWR@replaceEPSSVG}{}
}{%
    \newcommand*{\LWR@replaceEPSSVG}{%
        \StrSubstitute{\LWR@tempone}{.eps}{.svg}[\LWR@tempone]%
        \StrSubstitute{\LWR@tempone}{.EPS}{.SVG}[\LWR@tempone]%
    }
}%
}
```

`\LWR@includegraphicsb` `*` `[`⟨*2: options*⟩`]` `[`⟨*3: options*⟩`]` `{`⟨*4: filename*⟩`}`

graphics syntax is `\includegraphics` `*` `[`⟨*llx,lly*⟩`]` `[`⟨*urx,ury*⟩`]` `{`⟨*file*⟩`}`

graphicx syntax is `\includegraphics` `[`⟨*key values*⟩`]` `{`⟨*file*⟩`}`

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.

```
\NewDocumentCommand{\LWR@includegraphicsb}{s o o m}
{%
```

Start the image tag on a new line, allow PDF output word wrap:

```
\LWR@origtilde \LWR@orignewline%
```

Temporarily compute `\linewidth`, `\textwidth`, `\textheight` arguments with a 6x9 inch size until the next `\endgroup`.

```
\begingroup%
\ifthenelse{\cnttest{\value{LWR@minipagedepth}}{=}{0}}%
{%
    \ifbool{LWR@infloatrow}%
    {}
    {% not in a minipage or a floatrow:
        \setlength{\linewidth}{6in}%
        \setlength{\textwidth}{6in}%
        \setlength{\textheight}{9in}%
    }%
}{}%
```

For correct em sizing during the width and height conversions:

```
\large%
```

Reset some defaults, possibly will be changed below if options were given:

```
\setlength{\LWR@igwidth}{0pt}%
\setlength{\LWR@igheight}{0pt}%
\renewcommand*{\LWR@igwidthstyle}{}%
\renewcommand*{\LWR@igheightstyle}{}%
\renewcommand*{\LWR@igorigin}{}%
\renewcommand*{\LWR@igangle}{}%
\renewcommand*{\LWR@igxscale}{1}%
\renewcommand*{\LWR@igyscale}{1}%
\renewcommand*{\LWR@igclass}{inlineimage}%
```

```
\renewcommand*{\LWR@igalt}{(image)}%
```

If #3 is empty, only one optional argument was given, thus graphicx syntax:

```
\IfValueF{#3}{%
    \IfValueTF{#2}%
        {\setkeys{igraph}{#2}}%
        {\setkeys{igraph}{}}%
}%
```

Fully expand and detokenize the filename, changing the file extension to .svg if necessary.

```
\begingroup%
\LetLtxMacro\Gin@setfile\LWR@HTML@Gin@setfile%
\edef\LWR@tempone{#4}%
\StrSubstitute{\LWR@tempone}{.pdf}{.svg}[\LWR@tempone]%
\StrSubstitute{\LWR@tempone}{.PDF}{.SVG}[\LWR@tempone]%
\LWR@replaceEPSSVG%
\xdef\LWR@parsedfilename{\LWR@tempone}%
\Ginclude@graphics{\detokenize\expandafter{\LWR@tempone}}%
\endgroup%
\filename@parse{\LWR@parsedfilename}%
\LWR@traceinfo{LWR@parsedfilename is \LWR@parsedfilename}%
% \LWR@sanitize{\LWR@parsedfilename}%
```

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:

```
\ifbool{FormatWP}{%
    \begingroup%
    \LWR@restoreorigformatting%
    \ifpdf%
    \appto\LWR@restoreorigformatting{%
   \DeclareGraphicsExtensions{.pdf,.PDF,.gif,.GIF,.png,.PNG,.jpg,.JPG,.jpeg,.JPEG}%
    }%
    \else% \ifpdf
            \ifXeTeX%
```

```
    \appto\LWR@restoreorigformatting{%
  \DeclareGraphicsExtensions{.pdf,.PDF,.gif,.GIF,.png,.PNG,.jpg,.JPG,.jpeg,.JPEG}%
    }%
            \else%
    \appto\LWR@restoreorigformatting{%
  \DeclareGraphicsExtensions{.eps,.EPS,.gif,.GIF,.png,.PNG,.jpg,.JPG,.jpeg,.JPEG}%
    }%
            \fi%
    \fi% \ifpdf
    \define@key{Gin}{angle}{}%
    \IfBooleanTF{#1}%
    {% starred
        \IfValueTF{#3}%
        {%
        \global\sbox{\LWR@imagesizebox}{\LWR@origincludegraphics*[#2][#3]{#4}}%
        }%
        {%
            \IfValueTF{#2}%
            {%
           \global\sbox{\LWR@imagesizebox}{\LWR@origincludegraphics*[#2]{#4}}%
            }{%
               \global\sbox{\LWR@imagesizebox}{\LWR@origincludegraphics*{#4}}%
            }%
        }%
    }% starred
    {% not starred
        \IfValueTF{#3}%
        {%
        \global\sbox{\LWR@imagesizebox}{\LWR@origincludegraphics[#2][#3]{#4}}%
        }%
        {%
            \IfValueTF{#2}%
            {%
            \global\sbox{\LWR@imagesizebox}{\LWR@origincludegraphics[#2]{#4}}%
            }{%
                \global\sbox{\LWR@imagesizebox}{\LWR@origincludegraphics{#4}}%
            }%
        }%
    }% not starred
    \endgroup%
    \settowidth{\LWR@igwidth}{\usebox{\LWR@imagesizebox}}%
  \global\renewcommand*{\LWR@igwidthstyle}{width:\LWR@printlength{\LWR@igwidth}}%
    \settoheight{\LWR@igheight}{\usebox{\LWR@imagesizebox}}%
  \global\renewcommand*{\LWR@igheightstyle}{height:\LWR@printlength{\LWR@igheight}}%
}{}% FormatWP
```

Create the HTML reference with the graphicspath, filename, extension, alt tag, style, and class.

The `\LWR@origtilde` adds space between tags in case this is being done inside a `\savebox` where `\newline` has no effect.

```
\LWR@traceinfo{LWR@includegraphicsb: about to create href}%
\href{\LWR@parsedfilename}%
{% start of href
```

```
\LWR@traceinfo{LWR@includegraphicsb: about to LWR@htmltag}%
\LWR@htmltag{% start of image tags
img\LWR@indentHTML%
src="%

\detokenize\expandafter{\LWR@parsedfilename}%

"\LWR@indentHTML%
```

Only include a style tag if a width, height, angle, or scale was given:

```
\ifthenelse{
    \NOT\equal{\LWR@igwidthstyle}{} \OR
    \NOT\equal{\LWR@igheightstyle}{} \OR
    \NOT\equal{\LWR@igorigin}{} \OR
    \NOT\equal{\LWR@igangle}{} \OR
    \NOT\equal{\LWR@igxscale}{1} \OR
    \NOT\equal{\LWR@igyscale}{1}
}%
{%
    style="\LWR@indentHTML
    \ifthenelse{\NOT\equal{\LWR@igwidthstyle}{}}%
        {\LWR@igwidthstyle;\LWR@indentHTML}{}%
    \ifthenelse{\NOT\equal{\LWR@igheightstyle}{}}%
        {\LWR@igheightstyle;\LWR@indentHTML}{}%
    \ifthenelse{\NOT\equal{\LWR@igorigin}{}}%
        {%
            transform-origin: \LWR@originnames{\LWR@igorigin};\LWR@indentHTML%
        }{}%
    \ifthenelse{\NOT\equal{\LWR@igangle}{}}%
    {%
        \LWR@rotstyle{-ms-}{\LWR@igangle}\LWR@indentHTML
        \LWR@rotstyle{-webkit-}{\LWR@igangle}\LWR@indentHTML
        \LWR@rotstyle{}{\LWR@igangle }\LWR@indentHTML
    }{}%
    \ifthenelse{%
        \NOT\equal{\LWR@igxscale}{1}\OR%
        \NOT\equal{\LWR@igyscale}{1}%
    }%
    {%
        \LWR@scalestyle{-ms-}{\LWR@igxscale}{\LWR@igyscale}\LWR@indentHTML
        \LWR@scalestyle{-webkit-}{\LWR@igxscale}{\LWR@igyscale}\LWR@indentHTML
        \LWR@scalestyle{}{\LWR@igxscale}{\LWR@igyscale}\LWR@indentHTML
    }{}%
    %
    \ifthenelse{\NOT\equal{\LWR@opacity}{1}}%
        {opacity:\LWR@opacity;\LWR@indentHTML}{}%
    %
    "\LWR@indentHTML%
}{}%
```

Set the class and alt tag:

```
class="\LWR@igclass"\LWR@indentHTML%
```

```
alt="\LWR@igalt" \LWR@orignewline%
}% end of image tags
}% end of href
```

Return to original page size and font size:

```
\endgroup
\LWR@traceinfo{LWR@includegraphicsb done}%
}
```

\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.

```
\AtBeginDocument{

\LWR@traceinfo{Patching includegraphics.}

\LetLtxMacro\LWR@origincludegraphics\includegraphics

\renewcommand*{\includegraphics}
{%
```

This graphic should trigger an HTML paragraph even if alone, so ensure that are doing paragraph handling:

```
\LWR@traceinfo{includegraphics}%
\LWR@ensuredoingapar%
\LWR@includegraphicsb%
}% includegraphics
}% AtBeginDocument
```

§ 253.5 Boxes

\LWR@rotboxorigin Holds the origin key letters.

```
\newcommand*{\LWR@rotboxorigin}{}
```

\LWR@originname {⟨*letter*⟩}

Given one LaTeX origin key value, translate into an HTML origin word:

```
\newcommand*{\LWR@originname}[1]{%
    \ifthenelse{\equal{#1}{t}}{top}{}%
    \ifthenelse{\equal{#1}{b}}{bottom}{}%
    \ifthenelse{\equal{#1}{c}}{center}{}%
    \ifthenelse{\equal{#1}{l}}{left}{}%
    \ifthenelse{\equal{#1}{r}}{right}{}%
}
```

\LWR@originnames {⟨*letters*⟩}

Given one- or two-letter LaTeX origin key values, translate into HTML origin words:

```
\newcommand*{\LWR@originnames}[1]{%
\StrChar{#1}{1}[\LWR@strresult]%
\LWR@originname{\LWR@strresult}
\StrChar{#1}{2}[\LWR@strresult]%
\LWR@originname{\LWR@strresult}
}
```

Handle the origin key for \rotatebox:

```
\define@key{krotbox}{origin}{%
\renewcommand*{\LWR@rotboxorigin}{#1}%
}
```

These keys are ignored:

```
\define@key{krotbox}{x}{}
\define@key{krotbox}{y}{}
\define@key{krotbox}{units}{}
```

\rotatebox [⟨*keyval list*⟩] {⟨*angle*⟩} {⟨*text*⟩}

```
\AtBeginDocument{
```

The HTML version:

```
\NewDocumentCommand{\LWR@HTML@rotatebox}{O{} m +m}{%
```

Reset the origin to “none-given”:

```
\renewcommand*{\LWR@rotboxorigin}{}
```

Process the optional keys, which may set \LWR@rotateboxorigin:

```
\setkeys{krotbox}{#1}%
```

Select inline-block so that HTML will transform this span:

```
\LWR@htmltagc{%
    span\LWR@indentHTML
    style="\LWR@indentHTML
    display: inline-block;\LWR@indentHTML
```

If an origin was given, translate and print the origin information:

```
    \ifthenelse{\NOT\equal{\LWR@rotboxorigin}{}}%
      {transform-origin: \LWR@originnames{\LWR@rotboxorigin};\LWR@indentHTML}%
        {}%
```

Print the rotation information:

```
    \LWR@rotstyle{-ms-}{#2}\LWR@indentHTML
    \LWR@rotstyle{-webkit-}{#2}\LWR@indentHTML
    \LWR@rotstyle{}{#2}"\LWR@orignewline%
}\LWR@orignewline%
```

Print the text to be rotated:

```
\begin{LWR@nestspan}%
#3%
```

Close the span:

```
\LWR@htmltagc{/span}%
\end{LWR@nestspan}%
}
```

The high-level interface:

```
\LWR@formatted{rotatebox}

}% AtBeginDocument
```

\scalebox {⟨*h-scale*⟩} [⟨*v-scale*⟩] {⟨*text*⟩}

```
\AtBeginDocument{
```

The HTML version:

```
\NewDocumentCommand{\LWR@HTML@scalebox}{m o m}{%
```

Select `inline-block` so that HTML will transform this span:

```
\LWR@htmltagc{%
    span\LWR@indentHTML
    style="\LWR@indentHTML
    display: inline-block;\LWR@indentHTML
```

Print the scaling information:

```
    \LWR@scalestyle{-ms-}{#1}{\IfNoValueTF{#2}{#1}{#2}}\LWR@indentHTML
    \LWR@scalestyle{-webkit-}{#1}{\IfNoValueTF{#2}{#1}{#2}}\LWR@indentHTML
    \LWR@scalestyle{}{#1}{\IfNoValueTF{#2}{#1}{#2}}
    "\LWR@orignewline
}\LWR@orignewline%
```

Print the text to be scaled:

```
\begin{LWR@nestspan}%
#3%
```

Close the span:

```
\LWR@htmltagc{/span}%
\end{LWR@nestspan}%
}
```

The high-level interface:

```
\LWR@formatted{scalebox}

}% AtBeginDocument
```

`\reflectbox` {⟨*text*⟩}

```
\AtBeginDocument{

\newcommand{\LWR@HTML@reflectbox}[1]{%
\scalebox{-1}[1]{#1}%
}% \reflectbox

\LWR@formatted{reflectbox}

}% AtBeginDocument
```

`\resizebox` {⟨*h-length*⟩} {⟨*v-length*⟩} {⟨*text*⟩}

Simply prints its text argument.

```
\AtBeginDocument{

\NewDocumentCommand{\LWR@HTML@resizebox}{s m m m}{%
#4%
}

\LWR@formatted{resizebox}

}% AtBeginDocument
```

File 157 **lwarp-graphicx.sty**

§ 254 Package

graphicx

`Pkg graphicx` 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:

```
\LWR@ProvidesPackagePass{graphicx}[2017/06/01]
```

File 158 **lwarp-grffile.sty**

§ 255 Package

grffile

Pkg grffile

⚠ 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.

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:

```
\LWR@ProvidesPackagePass{grffile}[2017/06/30]
```

File 159 **lwarp-grid.sty**

§ 256 Package

grid

Pkg grid

grid is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{grid}[2009/06/16]

\newenvironment*{gridenv}{}{}
```

File 160 **lwarp-grid-system.sty**

§ 257 Package

grid-system

(*Emulates or patches code by* Marcus Bitzl.)

Pkg grid-system

grid-system is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{grid-system}[2014/02/16]
```

(\ifdef is in case the older syntax is removed.)

```
\AtBeginEnvironment{Row}{\setlength{\linewidth}{6in}}

\ifdef{\endrow}{
    \AtBeginEnvironment{row}{\setlength{\linewidth}{6in}}
}{}

\renewcommand{\gridsystem@finishcell}{\hspace{\gridsystem@cellsep}}
```

File 161 **lwarp-gridset.sty**

§ 258 Package gridset

Pkg gridset gridset is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{gridset}

\newcommand*{\gridbase}{}
\newcommand*{\gridinterval}{}
\newcommand*{\savepos}[1]{}
\newcounter{gridcnt}
\newcommand*{\vskipnextgrid}{}
\newcommand*{\thegridinfo}[1]{}
\newcommand*{\theposinfo}[1]{}
\newcommand*{\theypos}[1]{}
```

File 162 **lwarp-hang.sty**

§ 259 Package hang

(*Emulates or patches code by* Andreas Nolda.)

Pkg hang hang is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{hang}[2017/02/18]

\newlength{\hangingindent}
\setlength{\hangingindent}{1em}
\newlength{\hangingleftmargin}
\setlength{\hangingleftmargin}{0em}

\newcommand*{\LWR@findhangingleftmargin}{%
\setlength{\LWR@templengthone}{\hangingleftmargin}%
\addtolength{\LWR@templengthone}{\hangingindent}%
}

\newenvironment{hangingpar}
{
    \LWR@findhangingleftmargin%
    \BlockClass[%
        \LWR@print@mbox{margin-left:\LWR@printlength{\LWR@templengthone}} ; %
        \LWR@print@mbox{text-indent:-\LWR@printlength{\hangingindent}}%
    ]%
    {hangingpar}%
}
{\endBlockClass}

```

```
\newenvironment{hanginglist}
{%
    \renewcommand*{\LWR@printcloselist}{\LWR@printcloseitemize}%
    \renewcommand*{\LWR@printopenlist}{%
        \LWR@findhangingleftmargin%
        ul style="%
            \LWR@print@mbox{list-style-type:none;} % extra space
            \LWR@print@mbox{%
                margin-left:\LWR@printlength{\LWR@templengthone}%
            } ; % extra space
            \LWR@print@mbox{%
                text-indent:-\LWR@printlength{\hangingindent}%
            }%
        "%
    }%
    \let\item\LWR@itemizeitem%
    \list{}{}%
}
{\endlist}

\newenvironment{compacthang}
{\hanginglist}
{\endhanginglist}

\newlength{\labeledleftmargin}
\setlength{\labeledleftmargin}{0em}

\newenvironment{labeledpar}[2]
{%
    \BlockClass[%
        \LWR@findhangingleftmargin%
        \LWR@print@mbox{margin-left:\LWR@printlength{\LWR@templengthone}} ; %
        \LWR@print@mbox{text-indent:-\LWR@printlength{\hangingindent}}%
    ]{labeledpar}#2%
}
{\endBlockClass}

\newenvironment{labeledlist}[1]
{\hanginglist}
{\endhanginglist}

\newenvironment{compactlabel}[1]
{\hanginglist}
{\endhanginglist}
```

File 163 **lwarp-hanging.sty**

§ 260 Package

hanging

Pkg hanging hanging is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{hanging}[2009/09/02]
```

```
\@ifclassloaded{memoir}{
\let\hangpara\relax
\let\hangparas\relax
\let\endhangparas\relax
\let\hangpunct\relax
\let\endhangpunct\relax
}{}
```

\hangpara {⟨*indent*⟩} {⟨*afternum*⟩}

Use hangparas instead.

```
\newcommand*{\hangpara}[2]{}
```

Env hangparas {⟨*indent*⟩} {⟨*afternum*⟩}

```
\newenvironment*{hangparas}[2]
{%
    \BlockClass[%
        \LWR@print@mbox{margin-left:\LWR@printlength{#1}} ; %
        \LWR@print@mbox{text-indent:-\LWR@printlength{#1}}%
    ]%
    {hangingpar}%
}
{\endBlockClass}
```

Env hangpunct

```
\newenvironment*{hangpunct}
{\BlockClass{hangpunct}}
{\endBlockClass}

\newcommand{\nhpt}{.}
\newcommand{\nhlq}{‘}
\newcommand{\nhrq}{’}
```

File 164 **lwarp-hypcap.sty**

§ 261 Package

hypcap

Pkg hypcap hypcap is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{hypcap}[2016/05/16]

\newcommand*{\capstart}{}
\newcommand*{\hypcapspace}{}
\newcommand*{\hypcapredef}[1]{}
\newcommand*{\capstartfalse}{}
\newcommand*{\capstarttrue}{}
```

File 165 **lwarp-hypdestopt.sty**

§ 262 Package hypdestopt

Pkg hypdestopt hypdestopt is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{hypdestopt}[2016/05/21]
```

File 166 **lwarp-hypernat.sty**

§ 263 Package hypernat

Pkg hypernat hypernat is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{hypernat}[2001/07/09]
```

File 167 **lwarp-hyperref.sty**

§ 264 Package hyperref

(*Emulates or patches code by* Sebastian Rahtz, Heiko Oberdiek.)

Pkg hyperref hyperref is emulated.

for HTML output:

```
% \LWR@ProvidesPackageDrop{hyperref}
\typeout{Using the lwarp html version of package ‘hyperref’, discarding options.}
\typeout{    Are not using ProvidesPackage, so that other packages}
\typeout{    do not attempt to patch lwarp’s version of ‘hyperref’.}
% \ProvidesPackage{lwarp-#1-#2}
\DeclareOption*{}
\ProcessOptions\relax

\newcommand*{\hypersetup}[1]{}
\newcommand*{\hyperbaseurl}[1]{}
```

\hyperimage {⟨*URL*⟩} {⟨*alt text*⟩}

Insert an image with alt text:

```
\NewDocumentCommand{\LWR@hyperimageb}{m +m}{%
\LWR@ensuredoingapar%
\def\LWR@templink{#1}%
\@onelevel@sanitize\LWR@templink%
\LWR@htmltag{img src="\LWR@templink" alt="#2" class="hyperimage"}%
\LWR@ensuredoingapar%
```

```
\endgroup%
}

\newrobustcmd*{\hyperimage}{%
\begingroup%
\catcode`\#=12%
\catcode`\%=12%
\catcode`\&=12%
\catcode`\~=12%
\catcode`\_=12%
\LWR@hyperimageb%
}

```

\hyperdef {⟨*1: category*⟩} {⟨*2: name*⟩} {⟨*3: text*⟩}

Creates an HTML anchor to `category.name` with the given text.

```
\NewDocumentCommand{\LWR@hyperdefb}{m m +m}{%
\LWR@ensuredoingapar%
\LWR@sublabel{#1.#2}%
#3%
\endgroup%
}

\newcommand*{\hyperdef}{%
\begingroup%
\catcode`\#=12%
\catcode`\%=12%
\catcode`\&=12%
\catcode`\~=12%
\catcode`\_=12%
\LWR@hyperdefb%
}

```

\LWR@hyperrefb {⟨*1: URL*⟩} {⟨*2: category*⟩} {⟨*3: name*⟩} {⟨*4: text*⟩}

Creates an HTML link to `URL#category.name` with the given text.

```
\newcommand{\LWR@hyperreffinish}[1]{%
\begingroup%
\RenewDocumentCommand{\ref}{}{\LWR@ref@ignorestar}%
#1%
\endgroup%
\LWR@htmltag{/a}%
}

\newcommand*{\LWR@hyperrefbb}[3]{%
\LWR@htmltag{%
    a href="%
        \detokenize\expandafter{#1}\LWR@hashmark%
        \detokenize\expandafter{#2}.\detokenize\expandafter{#3}%
    "%
}%
\endgroup%
```

```
\LWR@hyperreffinish%
}

\newrobustcmd*{\LWR@hyperrefb}{%
\begingroup%
\catcode`\#=12%
\catcode`\%=12%
\catcode`\&=12%
\catcode`\~=12%
\catcode`\_=12%
\LWR@hyperrefbb%
}
```

`\LWR@hyperrefc` `[`⟨*label*⟩`]` `{`⟨*text*⟩`}`

Creates text as an HTML link to the LaTeX label.

```

\NewDocumentCommand{\LWR@hyperrefcb}{O{label}}{%
\LWR@startref{#1}%
\endgroup%
\LWR@hyperreffinish%
}

\newcommand*{\LWR@hyperrefc}{%
\begingroup%
\catcode`\#=12%
\catcode`\%=12%
\catcode`\&=12%
\catcode`\~=12%
\catcode`\_=12%
\LWR@hyperrefcb%
}
```

`\hyperref` `{`⟨*1: URL*⟩`}` `{`⟨*2: category*⟩`}` `{`⟨*3: name*⟩`}` `{`⟨*4: text*⟩`}` —or— `[`⟨*1: label*⟩`]` `{`⟨*2: text*⟩`}`

```
\DeclareRobustCommand*{\hyperref}{%
\LWR@ensuredoingapar%
\@ifnextchar[\LWR@hyperrefc\LWR@hyperrefb%
}
```

`\hypertarget` `{`⟨*name*⟩`}` `{`⟨*text*⟩`}`

Creates an anchor to `name` with the given text.

```
\NewDocumentCommand{\LWR@hypertargetb}{m +m}{%
\label{LWR-ht-#1}%
#2%
\endgroup%
}

\newcommand*{\hypertarget}{%
\begingroup%
\catcode`\#=12%
\catcode`\%=12%
```

```
\catcode`\&=12%
\catcode`\~=12%
\catcode`\_=12%
\LWR@hypertargetb%
}
```

`\hyperlink` {⟨*name*⟩} {⟨*text*⟩}

Creates a link to the anchor created by hypertarget, with the given link text.

Declared because also defined by memoir.

```
\DeclareDocumentCommand{\LWR@hyperlinkb}{m}{%
\LWR@hyperrefcb[LWR-ht-#1]%
}

\DeclareDocumentCommand{\hyperlink}{}{%
\LWR@ensuredoingapar%
\begingroup%
\catcode`\#=12%
\catcode`\%=12%
\catcode`\&=12%
\catcode`\~=12%
\catcode`\_=12%
\LWR@hyperlinkb%
}
```

`\autoref` * {⟨*label*⟩}

For HTML, `\cleveref` is used instead.

```
\NewDocumentCommand{\autoref}{s m}{%
    \IfBooleanTF{#1}{\ref{#2}}{\cref{#2}}%
}
```

`\autopageref` {⟨*label*⟩}

For HTML, `\cleveref` is used instead.

```
\NewDocumentCommand{\autopageref}{s m}{%
    \IfBooleanTF{#1}{\cpageref{#2}}{\cref{#2}}%
}
```

`\pdfstringdef` {⟨*macroname*⟩} {⟨*TEXstring*⟩}

```
\newcommand{\pdfstringdef}[2]{}
```

`\pdfbookmark` [⟨*level*⟩] {⟨*text*⟩} {⟨*name*⟩}

```
\newcommand{\pdfbookmark}[3][]{}
```

`\currentpdfbookmark` {⟨*text*⟩} {⟨*name*⟩}

```
\newcommand{\currentpdfbookmark}[2]{}
```

\subpdfbookmark {⟨*text*⟩} {⟨*name*⟩}

```
\newcommand{\subpdfbookmark}[2]{}
```

\belowpdfbookmark {⟨*text*⟩} {⟨*name*⟩}

```
\newcommand{\belowpdfbookmark}[2]{}
```

\texorpdfstring {⟨*TEXstring*⟩} {⟨*PDFstring*⟩}

```
\newcommand{\texorpdfstring}[2]{#1}
```

\hypercalcbp {⟨*dimen*⟩} From hyperref.

```
\def\hypercalcbp#1{%
\strip@pt\dimexpr 0.99626401\dimexpr(#1)\relax\relax
}%
```

\Acrobatmenu {⟨*menuoption*⟩} {⟨*text*⟩}

```
\newcommand{\Acrobatmenu}[2]{}
```

\TextField [⟨*parameters*⟩] {⟨*label*⟩}

```
\DeclareRobustCommand{\TextField}[2][]{}
```

\CheckBox [⟨*parameters*⟩] {⟨*label*⟩}

```
\DeclareRobustCommand{\CheckBox}[2][]{}
```

\ChoiceMenu [⟨*parameters*⟩] {⟨*label*⟩} {⟨*choices*⟩}

```
\DeclareRobustCommand{\ChoiceMenu}[3][]{}
```

\PushButton [⟨*parameters*⟩] {⟨*label*⟩}

```
\DeclareRobustCommand{\PushButton}[2][]{}
```

\Submit [⟨*parameters*⟩] {⟨*label*⟩}

```
\DeclareRobustCommand{\Submit}[2][]{}
```

\Reset [⟨*parameters*⟩] {⟨*label*⟩}

```
\DeclareRobustCommand{\Reset}[2][]{}
```

\Gauge [⟨*parameters*⟩] {⟨*label*⟩}

```
\DeclareRobustCommand{\Gauge}[2][]{}
```

\LayoutTextField {⟨*label*⟩} {⟨*field*⟩}

```
146 \newcommand*{\LayoutTextField}[2]{}
```

\LayoutChoiceField {⟨*label*⟩} {⟨*field*⟩}

```
147 \newcommand*{\LayoutChoiceField}[2]{}
```

\LayoutCheckField {⟨*label*⟩} {⟨*field*⟩}

```
148 \newcommand*{\LayoutCheckField}[2]{}
```

\MakeRadioField {⟨*width*⟩} {⟨*height*⟩}

```
149 \newcommand*{\MakeRadioField}[2]{}
```

\MakeCheckField {⟨*width*⟩} {⟨*height*⟩}

```
150 \newcommand*{\MakeCheckField}[2]{}
```

\MakeTextField {⟨*width*⟩} {⟨*height*⟩}

```
151 \newcommand*{\MakeTextField}[2]{}
```

\MakeChoiceField {⟨*width*⟩} {⟨*height*⟩}

```
152 \newcommand*{\MakeChoiceField}[2]{}
```

\MakeFieldButton {⟨*text*⟩}

```
153 \newcommand{\MakeFieldButton}[1]{}
```

File 168 **lwarp-hyperxmp.sty**

§ 265 Package hyperxmp

Pkg hyperxmp Emulated.

for HTML output: Discard all options for lwarp-hyperxmp:

```
1 \LWR@ProvidesPackageDrop{hyperxmp}[2018/11/27]
```

File 169 **lwarp-hyphenat.sty**

§ 266 Package hyphenat

Pkg hyphenat hyphenat is emulated during HTML output, while the print-mode version is used inside

a lateximage.

for HTML output:

```
\LWR@ProvidesPackagePass{hyphenat}[2009/09/02]

\LetLtxMacro\LWRHYNAT@origtextnhtt\textnhtt
\LetLtxMacro\LWRHYNAT@orignhttfamily\nhttfamily
\LetLtxMacro\LWRHYNAT@orignohyphens\nohyphens
\LetLtxMacro\LWRHYNAT@origbshyp\bshyp
\LetLtxMacro\LWRHYNAT@origfshyp\fshyp
\LetLtxMacro\LWRHYNAT@origdothyp\dothyp
\LetLtxMacro\LWRHYNAT@origcolonhyp\colonhyp
\LetLtxMacro\LWRHYNAT@orighyp\hyp

\LetLtxMacro\textnhtt\texttt
\LetLtxMacro\nhttfamily\ttfamily

\renewcommand{\nohyphens}[1]{#1}
\renewrobustcmd{\bshyp}{%
    \ifmmode\backslash\else\textbackslash\fi%
}
\renewrobustcmd{\fshyp}{/}
\renewrobustcmd{\dothyp}{.}
\renewrobustcmd{\colonhyp}{:}
\renewrobustcmd{\hyp}{-}

\appto\LWR@restoreorigformatting{%
\LetLtxMacro\textnhtt\LWRHYNAT@origtextnhtt%
\LetLtxMacro\nhttfamily\LWRHYNAT@orignhttfamily%
\LetLtxMacro\nohyphens\LWRHYNAT@orignohyphens%
\LetLtxMacro\bshyp\LWRHYNAT@origbshyp%
\LetLtxMacro\fshyp\LWRHYNAT@origfshyp%
\LetLtxMacro\dothyp\LWRHYNAT@origdothyp%
\LetLtxMacro\colonhyp\LWRHYNAT@origcolonhyp%
\LetLtxMacro\hyp\LWRHYNAT@orighyp%
}
```

File 170 **lwarp-idxlayout.sty**

§ 267 Package

idxlayout

(*Emulates or patches code by* Thomas Titz.)

Pkg idxlayout Emulated.

for HTML output: Discard all options for lwarp-idxlayout:

```
\LWR@ProvidesPackageDrop{idxlayout}[2012/03/30]

\newcommand{\LWR@indexprenote}{}

\preto\printindex{

\LWR@orignewpage
```

```
\LWR@startpars

\LWR@indexprenote

}

\newcommand{\setindexprenote}[1]{\renewcommand{\LWR@indexprenote}{#1}}
\newcommand*{\noindexprenote}{\renewcommand{\LWR@indexprenote}{}}

\newcommand{\idxlayout}[1]{}
\newcommand*{\indexfont}{}
\newcommand*{\indexjustific}{}
\newcommand*{\indexsubsdelim}{}
\newcommand*{\indexstheadcase}{}
```

File 171 **lwarp-ifoddpage.sty**

§ 268 Package ifoddpage

(*Emulates or patches code by* Martin Scharrer.)

Pkg ifoddpage ifoddpage is emulated.

for HTML output: Discard all options for lwarp-ifoddpage:

```
\LWR@ProvidesPackageDrop{ifoddpage}[2016/04/23]

\newif\ifoddpage

\newif\ifoddpageoroneside

\DeclareRobustCommand{\checkoddpage}{\oddpagetrue\oddpageoronesidetrue}

\def\oddpage@page{1}

\def\@ifoddpage{%
        \expandafter\@firstoftwo
}

\def\@ifoddpageoroneside{%
        \expandafter\@firstoftwo
}
```

File 172 **lwarp-imakeidx.sty**

§ 269 Package imakeidx

(*Emulates or patches code by* Enrico Gregorio.)

Pkg imakeidx 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 9.6.17 for how to setup *lwarpmk* to process the indexes with imakeidx, both with and without shell escape.

for HTML output:

```
\LWR@ProvidesPackagePass{imakeidx}[2016/10/15]
```

Use the new HTML suffix:

```
\catcode`\_=12%
\define@key{imki}{name}{\def\imki@name{#1_html}}
\catcode`\_=8%
```

\printindex

The HTML version of \printindex:

```
\catcode`\_=12%

\renewcommand*{\printindex}[1][\imki@jobname]{%
\LWR@orignewpage%
\LWR@startpars%
\ifstrequal{#1}{\imki@jobname}{%
  \@ifundefined{#1@idxfile}{%
        \imki@error{#1}%
    }{%
        \imki@putindex{#1}%
    }%
}{%
 \@ifundefined{#1_html@idxfile}{\imki@error{#1_html}}{\imki@putindex{#1_html}}%
}%
}

\catcode`\_=8%
```

\@index

The HTML version of \index:

```
\catcode`\_=12%

\def\@index[#1]{%
    \ifstrequal{#1}{\imki@jobname}%
    {%
        \@ifundefined{#1@idxfile}%
        {%
            \PackageWarning{imakeidx}{Undefined index file `#1'}%
            \begingroup
            \@sanitize
            \imki@nowrindex%
        }%
        {%
```

```
            \edef\@idxfile{#1}%
            \begingroup
            \@sanitize
            \@wrindex\@idxfile%
        }%
    }%
    {%
        \@ifundefined{#1_html@idxfile}%
        {%
            \PackageWarning{imakeidx}{Undefined index file ‘#1_html’}%
            \begingroup
            \@sanitize
            \imki@nowrindex%
        }%
        {%
            \edef\@idxfile{#1_html}%
            \begingroup
            \@sanitize
            \@wrindex\@idxfile%
        }%
    }%
}

\catcode‘\_=8%
```

\item
\subitem
\subsubitem

HTML versions of \item, etc.:

```
\appto\theindex{%
    \let\item\LWR@indexitem%
    \let\subitem\LWR@indexsubitem%
    \let\subsubitem\LWR@indexsubsubitem%
}
```

\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:

```
\renewcommand\imki@wrindexentrysplit[3]{%
\addtocounter{LWR@autoindex}{1}%
\LWR@new@label{LWRindex-\arabic{LWR@autoindex}}%
  \expandafter\protected@write\csname#1@idxfile\endcsname{}%
    {\string\indexentry{#2}{\arabic{LWR@autoindex}}}%
}

\renewcommand\imki@wrindexentryunique[3]{%
\addtocounter{LWR@autoindex}{1}%
\LWR@new@label{LWRindex-\arabic{LWR@autoindex}}%
  \protected@write\@indexfile{}%
    {\string\indexentry[#1]{#2}{\arabic{LWR@autoindex}}}%
}
```

```
\def\imki@wrindexsplit#1#2{%
\imki@wrindexentrysplit{#1}{#2}{\thepage}%
\endgroup\imki@showidxentry{#1}{#2}%
  \@esphack%
}

\def\imki@wrindexunique#1#2{%
\imki@wrindexentryunique{#1}{#2}{\thepage}%
\endgroup\imki@showidxentry{#1}{#2}%
  \@esphack%
  }

```

`\LWR@imki@setxdydefopts`

Sets the xindy HTML options, ignoring the user's settings.

```
\newcommand*{\LWR@imki@setxdydefopts}{%
    \edef\imki@options{ \space %
        -M \space \LWR@xindyStyle\space %
        -L \space \LWR@xindyLanguage\space %
        -C \space \LWR@xindyCodepage\space %
    }%
}
```

`\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.

```
\newcommand*{\LWR@imki@setdefopts}[1]{%
\ifblank{#1}{%
    \edef\imki@options{\space -s \space \LWR@makeindexStyle \space}%
    \ifdefstring{\imki@progdefault}{xindy}{\LWR@imki@setxdydefopts}{}%
    \ifdefstring{\imki@progdefault}{texindy}{\LWR@imki@setxdydefopts}{}%
    \ifdefstring{\imki@progdefault}{truexindy}{\LWR@imki@setxdydefopts}{}%
}{%
    \edef\imki@options{\space #1 \space}%
}%
}
```

`\imki@makeindex` Use the new HTML options:

```
\xpatchcmd{\imki@makeindex}
    {\let\imki@options\space}
    {\LWR@imki@setdefopts{}}%
    {}
    {\LWR@patcherror{imakeidx}{makeindex}}
```

Use the new HTML options.

```
\define@key{imki}{options}{\LWR@imki@setdefopts{#1}}
```

\imki@resetdefaults Use the new HTML options:

```
\xpatchcmd{\imki@resetdefaults}
    {\def\imki@options{ }}
    {\LWR@imki@setdefopts{}}
    {}
    {\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

```
\AtBeginDocument{
\renewenvironment*{theindex}{%
    \imki@maybeaddtotoc
     \imki@indexlevel{\indexname}
    \let\item\LWR@indexitem%
    \let\subitem\LWR@indexsubitem%
    \let\subsubitem\LWR@indexsubsubitem%
}{}
}% AtBeginDocument
```

Update to the new defaults:

```
\imki@resetdefaults
```

Update to the new patches:

\AtBeginDocument is because \@wrindex is previously defined as \AtBeginDocument in the lwarp core.

```
\ifimki@splitindex
  \let\imki@startidx\imki@startidxunique
  \AtBeginDocument{\let\@wrindex\imki@wrindexunique}
  \let\imki@putindex\imki@putindexunique
  \let\imki@wrindexentry\imki@wrindexentryunique
  \let\imki@startidxsplit\@undefined
  \let\imki@wrindexsplit\@undefined
  \let\imki@putindexsplit\@undefined
\else
  \let\imki@startidx\imki@startidxsplit
  \AtBeginDocument{\let\@wrindex\imki@wrindexsplit}
  \let\imki@putindex\imki@putindexsplit
  \let\imki@wrindexentry\imki@wrindexentrysplit
  \let\imki@startidxunique\@undefined
  \let\imki@wrindexunique\@undefined
  \let\imki@putindexunique\@undefined
\fi
```

File 173 **lwarp-index.sty**

§ 270 Package

index

(*Emulates or patches code by* David M. Jones.)

Pkg index index is patched for use by lwarp.

for HTML output:

```
\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.

```
\xpatchcmd{\newindex}
    {\x@newindex[thepage]}
    {%
        \@tempswatrue%
        \x@newindex[theLWR@autoindex]%
    }
    {}
    {\LWR@patcherror{index}{newindex}}

\xpatchcmd{\renewindex}
    {\x@renewindex[thepage]}
    {%
        \@tempswatrue%
        \x@renewindex[theLWR@autoindex]%
    }
    {}
    {\LWR@patcherror{index}{renewindex}}
```

Patched to set a new autoindex:

```
\xpatchcmd{\@wrindex}
    {\begingroup}
    {%
        \addtocounter{LWR@autoindex}{1}%                    lwarp
        \LWR@new@label{LWRindex-\arabic{LWR@autoindex}}%    lwarp
        \begingroup%
    }
    {}
    {\LWR@patcherror{index}{@wrindex}}
```

\AtBeginDocument lwarp core \lets \@wrindex to \LWR@wrindex. Since the index package has been loaded, \let to its version instead:

```
\let\LWR@index@wrindex\@wrindex

\AtBeginDocument{
\let\@wrindex\LWR@index@wrindex
}
```

Modified to add \index@prologue:

```
\AtBeginDocument{
\renewenvironment*{theindex}{%
    \LWR@indexsection{\indexname}%
    \ifx\index@prologue\@empty\else
        \index@prologue
        \bigskip
    \fi
    \let\item\LWR@indexitem%
    \let\subitem\LWR@indexsubitem%
    \let\subsubitem\LWR@indexsubsubitem%
}{}
}% AtBeginDocument
```

Disabled:

```
\def\@showidx#1{}
\let\@texttop\relax
\renewcommand*{\raggedbottom}{}
\renewcommand*{\flushbottom}{}
\renewcommand*{\markboth}[2]{}
\renewcommand*{\markright}[1]{}
```

File 174 **lwarp-intopdf.sty**

§ 271 Package intopdf

Pkg intopdf intopdf is emulated.

The MIME type and description are ignored for now.

for HTML output:

```
\LWR@ProvidesPackageDrop{intopdf}[2018/03/15]
```

```
\NewDocumentCommand{\attachandlink}{m o m m}{%
    \href{#1}{#4}%
}
```

File 175 **lwarp-karnaugh-map.sty**

§ 272 Package karnaugh-map

(*Emulates or patches code by* Mattias Jacobsson.)

Pkg karnaugh-map karnaugh-map is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{karnaugh-map}[2017/02/20]
```

(It is hard to patch this macro, so the entire thing is redefined here, with the lwarp modificaitions identified in comments.)

```
\RenewDocumentEnvironment{karnaugh-map}{s O{4} O{4} O{1} O{$X_1X_0$} O{$X_3X_2$} O{$X_5X_4$}} {%
  \begingroup
    % store map size {[START]
      \renewcommand{\@karnaughmap@var@mapsizex@}{#2}%
      \renewcommand{\@karnaughmap@var@mapsizey@}{#3}%
      \renewcommand{\@karnaughmap@var@mapsizez@}{#4}%
    % [END]}
    % determinate if markings should be color or black and white
    \IfBooleanTF{#1}{%
      % should be black and white
      \renewcommand{\@karnaughmap@var@bw@}{1}%
    }{%
      % should be color
      \renewcommand{\@karnaughmap@var@bw@}{0}%
    }%
    %
    % find matching matrix template and alignment parameters {[START]
    \newcommand{\@karnaughmap@local@matrixtemplate@}{0}% '0' is considered as missing matrix template
      \newcommand{\@karnaughmap@local@maprealignmentx@}{0}%
      \newcommand{\@karnaughmap@local@maprealignmenty@}{0}%
    \ifnum\@karnaughmap@var@mapsizex@\@karnaughmap@var@mapsizey@\@karnaughmap@var@mapsizez@=221
        \renewcommand{\@karnaughmap@local@matrixtemplate@}{%
                  \&                  0 \&                   1 \& \phantom{0} \\
             0 \& |(000000)|  \phantom{0} \& |(000001)|  \phantom{0} \&         \\
             1 \& |(000010)|  \phantom{0} \& |(000011)|  \phantom{0} \&         \\
        \phantom{0}  \&                    \&                    \&          \\
        }%
      \fi
    \ifnum\@karnaughmap@var@mapsizex@\@karnaughmap@var@mapsizey@\@karnaughmap@var@mapsizez@=241
        \renewcommand{\@karnaughmap@local@matrixtemplate@}{%
                  \&                  0 \&                   1 \& \phantom{00} \\
            00 \& |(000000)|  \phantom{0} \& |(000001)|  \phantom{0} \&         \\
            01 \& |(000010)|  \phantom{0} \& |(000011)|  \phantom{0} \&         \\
            11 \& |(000110)|  \phantom{0} \& |(000111)|  \phantom{0} \&         \\
            10 \& |(000100)|  \phantom{0} \& |(000101)|  \phantom{0} \&         \\
        \phantom{00} \&                    \&                    \&          \\
        }%
      \fi
    \ifnum\@karnaughmap@var@mapsizex@\@karnaughmap@var@mapsizey@\@karnaughmap@var@mapsizez@=421
        \renewcommand{\@karnaughmap@local@matrixtemplate@}{%
                \&               00 \&               01 \&               11 \&               10 \& \phanto
             0  \& |(000000)|  \phantom{0} \& |(000001)|  \phantom{0} \& |(000011)|  \phantom{0} \& |(00001
             1  \& |(000100)|  \phantom{0} \& |(000101)|  \phantom{0} \& |(000111)|  \phantom{0} \& |(00011
       \phantom{00} \&                  \&                  \&                  \&                  \&
        }%
      \fi
    \ifnum\@karnaughmap@var@mapsizex@\@karnaughmap@var@mapsizey@\@karnaughmap@var@mapsizez@=441
        \renewcommand{\@karnaughmap@local@matrixtemplate@}{%
                \&               00 \&               01 \&               11 \&               10 \& \phanto
            00 \& |(000000)|  \phantom{0} \& |(000001)|  \phantom{0} \& |(000011)|  \phantom{0} \& |(0000
            01 \& |(000100)|  \phantom{0} \& |(000101)|  \phantom{0} \& |(000111)|  \phantom{0} \& |(0001
            11 \& |(001100)|  \phantom{0} \& |(001101)|  \phantom{0} \& |(001111)|  \phantom{0} \& |(0011
```

```
            10 \& |(001000)| \phantom{0} \& |(001001)| \phantom{0} \& |(001011)| \phantom{0} \& |(0010
     \phantom{00} \&                \&                \&                \&                \&
      }%
    \fi
  \ifnum\@karnaughmap@var@mapsizex@\@karnaughmap@var@mapsizey@\@karnaughmap@var@mapsizez@=442
      \renewcommand{\@karnaughmap@local@matrixtemplate@}{%
              \&              00 \&              01 \&              11 \&              10 \& \phanto
            00 \& |(000000)| \phantom{0} \& |(000001)| \phantom{0} \& |(000011)| \phantom{0} \& |(0000
            01 \& |(000100)| \phantom{0} \& |(000101)| \phantom{0} \& |(000111)| \phantom{0} \& |(0001
            11 \& |(001100)| \phantom{0} \& |(001101)| \phantom{0} \& |(001111)| \phantom{0} \& |(0011
            10 \& |(001000)| \phantom{0} \& |(001001)| \phantom{0} \& |(001011)| \phantom{0} \& |(0010
     \phantom{00} \&                \&                \&                \&                \&
      }%
      \renewcommand{\@karnaughmap@local@maprealignmentx@}{2.5}%
    \fi
  \ifnum\@karnaughmap@var@mapsizex@\@karnaughmap@var@mapsizey@\@karnaughmap@var@mapsizez@=444
      \renewcommand{\@karnaughmap@local@matrixtemplate@}{%
              \&              00 \&              01 \&              11 \&              10 \& \phanto
            00 \& |(000000)| \phantom{0} \& |(000001)| \phantom{0} \& |(000011)| \phantom{0} \& |(0000
            01 \& |(000100)| \phantom{0} \& |(000101)| \phantom{0} \& |(000111)| \phantom{0} \& |(0001
            11 \& |(001100)| \phantom{0} \& |(001101)| \phantom{0} \& |(001111)| \phantom{0} \& |(0011
            10 \& |(001000)| \phantom{0} \& |(001001)| \phantom{0} \& |(001011)| \phantom{0} \& |(0010
     \phantom{00} \&                \&                \&                \&                \&
            00 \& |(100000)| \phantom{0} \& |(100001)| \phantom{0} \& |(100011)| \phantom{0} \& |(1000
            01 \& |(100100)| \phantom{0} \& |(100101)| \phantom{0} \& |(100111)| \phantom{0} \& |(1001
            11 \& |(101100)| \phantom{0} \& |(101101)| \phantom{0} \& |(101111)| \phantom{0} \& |(1011
            10 \& |(101000)| \phantom{0} \& |(101001)| \phantom{0} \& |(101011)| \phantom{0} \& |(1010
     \phantom{00} \&                \&                \&                \&                \&
      }%
      \renewcommand{\@karnaughmap@local@maprealignmentx@}{2.5}%
      \renewcommand{\@karnaughmap@local@maprealignmenty@}{-2.5}%
    \fi
  % [END]}
 % test if a matrix template is found or not(aka "\@karnaughmap@local@matrixtemplate@" equals to '0')
  \ifdefstring{\@karnaughmap@local@matrixtemplate@}{0}{%  lwarp
%   \ifnum0=\@karnaughmap@local@matrixtemplate@%  original
    % print error if no template could be found
    \PackageError{karnaugh-map}{%
    Can not find a template fitting your specification (\@karnaughmap@var@mapsizex@\space x \@karnaughma
    }{%
    Existing templates have the following dimensions: 2x2x1, 2x4x1, 4x2x1, 4x4x1, 4x4x2, and 4x4x4.
    }%
%   \fi   original
  }{}%    lwarp
  \begin{tikzpicture}
    % grid
    % for all dimensions
  \draw[color=black, ultra thin] (0,0) grid (\@karnaughmap@var@mapsizex@,\@karnaughmap@var@mapsizey@);
    % when there are 2 sub maps
    \ifnum\@karnaughmap@var@mapsizez@=2
      \draw[color=black, ultra thin] (5,0) grid (9,4);
    \fi
    % when there are 4 sub maps
    \ifnum\@karnaughmap@var@mapsizez@=4
      \draw[color=black, ultra thin] (5,0) grid (9,4);
```

```
109         \draw[color=black, ultra thin] (0,-5) grid (4,-1);
110         \draw[color=black, ultra thin] (5,-5) grid (9,-1);
111       \fi
112       % labels
113       % for all dimensions
114     \node[above] at (\@karnaughmap@var@mapsizex@*0.5,\@karnaughmap@var@mapsizey@+0.9) {\small{#5}};
115       \node[left] at (-0.9,\@karnaughmap@var@mapsizey@*0.5) {\small{#6}};
116       % when there are 2 sub maps
117       \ifnum\@karnaughmap@var@mapsizez@=2
118         \node[above] at (7,4.9) {\small{#5}};
119         % extra sub maps labels
120         \node[below] at (2,-0.1) {\small{#7$=0$}};
121         \node[below] at (7,-0.1) {\small{#7$=1$}};
122       \fi
123       % when there are 4 sub maps
124       \ifnum\@karnaughmap@var@mapsizez@=4
125         \node[above] at (7,4.9) {\small{#5}};
126         \node[left] at (-0.9,-3) {\small{#6}};
127         % extra sub maps labels
128         \node[below] at (2,-0.1) {\small{#7$=00$}};
129         \node[below] at (7,-0.1) {\small{#7$=01$}};
130         \node[below] at (2,-5.1) {\small{#7$=10$}};
131         \node[below] at (7,-5.1) {\small{#7$=11$}};
132       \fi
133       % data
134       \matrix[
135         matrix of nodes,
136         ampersand replacement=\&,
137         column sep={1cm,between origins},
138         row sep={1cm,between origins},
139     ] at (\@karnaughmap@var@mapsizex@*0.5+\@karnaughmap@local@maprealignmentx@,\@karnaughmap@var@mapsiz
140         \@karnaughmap@local@matrixtemplate@%
141       };
142 }{
143     \end{tikzpicture}
144   \endgroup
145 }
```

File 176 **lwarp-keyfloat.sty**

§ 273 Package

keyfloat

(*Emulates or patches code by* Brian Dunn.)

Pkg keyfloat keyfloat is supported with minor adjustments.

⚠ keywrap If placing a `\keyfig[H]` inside a keywrap, use an absolute width for `\keyfig`, instead of `lw`-proportional widths. (The `[H]` option forces the use of a minipage, which internally adjusts for a virtual 6-inch wide minipage, which then corrupts the `lw` option.)

for HTML output:

```
1 \LWR@ProvidesPackagePass{keyfloat}[2017/05/12]
```

After keyfloat has loaded:

```
\AtBeginDocument{

\RenewDocumentCommand{\KFLT@onefigureimage}{}
{%
\LWR@traceinfo{KFLT@onefigureimage}%
% \begin{lrbox}{\KFLT@envbox}%
\ifthenelse{\NOT\equal{\KFLT@lw}{}}%
    {%
        \includegraphics%
        [scale=\KFLT@s,width=\KFLT@imagewidth]{\KFLT@i}%
    }%
    {% not linewidth
        \ifthenelse{\dimtest{\KFLT@w}{>}{0pt}}%
        {% width is given
            \ifthenelse{\dimtest{\KFLT@h}{>}{0pt}}%
            {% w and h
                \includegraphics%
                [scale=\KFLT@s,%
                width=\KFLT@imagewidth,height=\KFLT@h]{\KFLT@i}%
            }% w and h
            {% only w
                \includegraphics%
                [scale=\KFLT@s,width=\KFLT@imagewidth]{\KFLT@i}%
            }% only w
        }% width is given
        {% width is not given
            \ifthenelse{\dimtest{\KFLT@h}{>}{0pt}}%
            {%
                \includegraphics%
                [scale=\KFLT@s,height=\KFLT@h]{\KFLT@i}%
            }%
            {%
                \includegraphics%
                [scale=\KFLT@s]{\KFLT@i}%
            }%
        }% width is not given
    }% not linewidth
% \end{lrbox}%
% \unskip%
% \KFLT@findenvboxwidth%
% \begin{turn}{\KFLT@r}%
% \KFLT@frame{\usebox{\KFLT@envbox}}%
% \unskip%
% \end{turn}%
\LWR@traceinfo{KFLT@onefigureimage: done}%
}

\RenewDocumentEnvironment{KFLT@boxinner}{}
{%
\LWR@traceinfo{KFLT@boxinner}%
\LWR@stoppars%
}
{
```

```
\LWR@startpars%
\LWR@traceinfo{KFLT@boxinner: done}%
}

\DeclareDocumentEnvironment{KFLT@marginfloat}{O{-1.2ex} m}
{%
\LWR@BlockClassWP{float:right; width:2in; margin:10pt}{}{marginblock}%
\captionsetup{type=#2}%
}
{%
\endLWR@BlockClassWP%
}

\DeclareDocumentEnvironment{marginfigure}{o}
  {\begin{KFLT@marginfloat}{figure}}
  {\end{KFLT@marginfloat}}

\DeclareDocumentEnvironment{margintable}{o}
  {\begin{KFLT@marginfloat}{table}}
  {\end{KFLT@marginfloat}}

\DeclareDocumentEnvironment{keywrap}{m +m}
{%
\LWR@ensuredoingapar%
\setlength{\LWR@templengthone}{#1}%
\begin{LWR@BlockClassWP}{%
    float:right; width:\LWR@printlength{\LWR@templengthone};  % extra space
    margin:10pt%
}%
{%
    width:\LWR@printlength{\LWR@templengthone}%
}%
{marginblock}%
\setlength{\linewidth}{.95\LWR@templengthone}%
#2%
\end{LWR@BlockClassWP}%
}
{%
}

}% AtBeginDocument
```

File 177 **lwarp-layaureo.sty**

§ 274 Package

layaureo

Pkg layaureo layaureo is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{layaureo}[2004/09/16]
```

File 178 **lwarp-layout.sty**

§ 275 Package layout

(*Emulates or patches code by* Kent McPherson, Johannes Braams, Hideo Umeki.)

Pkg layout layout is emulated.

for HTML output: Discard all options for lwarp-layout:

```
\LWR@ProvidesPackageDrop{layout}[2014/10/28]
\NewDocumentCommand{\layout}{s}{}
```

File 179 **lwarp-leading.sty**

§ 276 Package leading

Pkg leading leading is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{leading}[2008/12/11]
\newcommand\leading[1]{}
```

File 180 **lwarp-letterspace.sty**

§ 277 Package letterspace

(*Emulates or patches code by* R Schlicht.)

Pkg letterspace 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:

```
\LWR@ProvidesPackageDrop{letterspace}[2018/01/14]
\newcommand*\lsstyle{}
\newcommand\textls[2][]{}
\def\textls#1#{}
\newcommand*\lslig[1]{#1}
```

File 181 **lwarp-lettrine.sty**

§ 278 Package

lettrine

(*Emulates or patches code by* Daniel Flipo.)

Pkg lettrine Emulated.

for HTML output: Discard all options for lwarp-lettrine:

```
\LWR@ProvidesPackageDrop{lettrine}[2018-08-28]
```

The initial letter is in a of class `lettrine`, and the following text is in a of class `lettrinetext`. `\lettrine` [⟨*keys*⟩] {⟨*letter*⟩} {⟨*additional text*⟩}

```
\DeclareDocumentCommand{\lettrine}{o m m}{%
\InlineClass{lettrine}{#2}\InlineClass{lettrinetext}{#3} % extra space
}

\newcounter{DefaultLines}
\setcounter{DefaultLines}{2}
\newcounter{DefaultDepth}
\newcommand*{\DefaultOptionsFile}{\relax}
\newcommand*{\DefaultLoversize}{0}
\newcommand*{\DefaultLraise}{0}
\newcommand*{\DefaultLhang}{0}
\newdimen\DefaultFindent
\setlength{\DefaultFindent}{\z@}
\newdimen\DefaultNindent
\setlength{\DefaultNindent}{0.5em}
\newdimen\DefaultSlope
\setlength{\DefaultSlope}{\z@}
\newdimen\DiscardVskip
\setlength{\DiscardVskip}{0.2\p@}
\newif\ifLettrineImage
\newif\ifLettrineOnGrid
\newif\ifLettrineRealHeight

\newcommand*{\LettrineTextFont}{\scshape}
\newcommand*{\LettrineFontHook}{}
\newcommand*{\LettrineFont}[1]{\InlineClass{lettrine}{#1}}
\newcommand*{\LettrineFontEPS}[1]{\includegraphics[height=1.5ex]{#1}}
```

File 182 **lwarp-lineno.sty**

§ 279 Package

lineno

(*Emulates or patches code by* Stephan I. Böttcher.)

Pkg lineno lineno is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{lineno}[2005/11/02]

\newcommand*\resetlinenumber[1][\@ne]{}

\def\linenumbers{%
     \@ifnextchar[{\resetlinenumber}%]
                  {\@ifstar{\resetlinenumber}{}}%
     }

\newcommand*{\nolinenumbers}{}

\@namedef{linenumbers*}{\par\linenumbers*}
\@namedef{runninglinenumbers*}{\par\runninglinenumbers*}

\def\endlinenumbers{\par}
\let\endrunninglinenumbers\endlinenumbers
\let\endpagewiselinenumbers\endlinenumbers
\expandafter\let\csname endlinenumbers*\endcsname\endlinenumbers
\expandafter\let\csname endrunninglinenumbers*\endcsname\endlinenumbers
\let\endnolinenumbers\endlinenumbers

\def\pagewiselinenumbers{\linenumbers\setpagewiselinenumbers}

\def\runninglinenumbers{\setrunninglinenumbers\linenumbers}

\def\setpagewiselinenumbers{}

\def\setrunninglinenumbers{}

\def\linenomath{}%
\@namedef{linenomath*}{}%
\def\endlinenomath{}
\expandafter\let\csname endlinenomath*\endcsname\endlinenomath

\let\linelabel\label

\def\switchlinenumbers{\@ifstar{}{}}
\def\setmakelinenumbers#1{\@ifstar{}{}}

\def\leftlinenumbers{\@ifstar{}{}}
\def\rightlinenumbers{\@ifstar{}{}}

\newcounter{linenumber}
\newcount\c@pagewiselinenumber
\let\c@runninglinenumber\c@linenumber

\def\runningpagewiselinenumbers{}
\def\realpagewiselinenumbers{}


\NewDocumentCommand\modulolinenumbers{s o}{}

\chardef\c@linenumbermodulo=5
```

```
\modulolinenumbers[1]

\newcommand*\firstlinenumber[1]{}

\newcommand\internallinenumbers{}
\let\endinternallinenumbers\endlinenumbers
\@namedef{internallinenumbers*}{\internallinenumbers*}
\expandafter\let\csname endinternallinenumbers*\endcsname\endlinenumbers

\newcommand*{\linenoplaceholder}[1]{% redefine per language
    (line number reference for \detokenize\expandafter{#1})
}

\newcommand*{\lineref}[2][]{\linenoplaceholder{#2}}
\newcommand*{\linerefp}[2][]{\linenoplaceholder{#2}}
\newcommand*{\linerefr}[2][]{\linenoplaceholder{#2}}

\newcommand\quotelinenumbers
   {\@ifstar\linenumbers{\@ifnextchar[\linenumbers{\linenumbers*}}}

\newdimen\linenumbersep
\newdimen\linenumberwidth
\newdimen\quotelinenumbersep

\quotelinenumbersep=\linenumbersep
\let\quotelinenumberfont\linenumberfont

\def\linenumberfont{\normalfont\tiny\sffamily}


\linenumberwidth=10pt
\linenumbersep=10pt

\def\thelinenumber{}

\def\LineNumber{}
\def\makeLineNumber{}
\def\makeLineNumberLeft{}
\def\makeLineNumberRight{}
\def\makeLineNumberOdd{}
\def\makeLineNumberEven{}
\def\makeLineNumberRunning{}


\newenvironment{numquote}     {\quote}{\endquote}
\newenvironment{numquotation} {\quotation}{\endquotation}
\newenvironment{numquote*}    {\quote}{\endquote}
\newenvironment{numquotation*}{\quotation}{\endquotation}

\newdimen\bframerule
\bframerule=\fboxrule

\newdimen\bframesep
\bframesep=\fboxsep

```

```
\newenvironment{bframe}
{%
    \LWR@forceminwidth{\bframerule}%
    \BlockClass[
        border:\LWR@printlength{\LWR@atleastonept} solid black ; %
        padding:\LWR@printlength{\bframesep}%
    ]{bframe}
}
{\endBlockClass}
```

File 183 **lwarp-lips.sty**

§ 280 Package lips

(*Emulates or patches code by* Matt Swift.)

Pkg lips lips is emulated.

```
% \LWR@ProvidesPackageDrop{lips}
\PackageInfo{lwarp}{Using the lwarp version of package ‘lips’.}%
\ProvidesPackage{lwarp-lips}[2001/08/31]

\NewDocumentCommand{\Lips}{}{\textellipsis}

\NewDocumentCommand{\BracketedLips}{}{[\textellipsis]}

\let\lips\Lips
\let\olips\lips

\DeclareOption*{}
\DeclareOption{mla}{
\let\lips\BracketedLips
}
\ProcessOptions\relax

\newcommand \LPNobreakList {}
```

File 184 **lwarp-listings.sty**

§ 281 Package listings

(*Emulates or patches code by* Carsten Heinz, Brooks Moses, Jobst Hoffmann.)

Pkg listings listings is supported with some limitations. Text formatting is not yet supported.

for HTML output:

```
\begin{warpHTML}

\LWR@ProvidesPackagePass{listings}[2015/06/04]
```

Force flexible columns. Fixed columns inserts spaces in the PDF output.

```
\lst@column@flexible
```

Patches to embed listings inside pre tags:

```
\let\LWR@origlst@Init\lst@Init
\let\LWR@origlst@DeInit\lst@DeInit

\let\LWR@origlsthkEveryPar\lsthk@EveryPar

\renewcommand{\l@lstlisting}[2]{\hypertocfloat{1}{lstlisting}{lol}{#1}{#2}}
```

`\lstset` {⟨*options*⟩}

Use the listings `literate` option to replace HTML entities:

```
\def\lstset@#1{\endgroup%
%  \ifx\@empty#1%
%         \@empty%
%     \else%
        \setkeys{lst}{%
            #1%
            ,literate=%
            {<}{\HTMLentity{lt}}{4}%
            {>}{\HTMLentity{gt}}{4}%
            {\&}{\HTMLentity{amp}}{5}%
        }%
%     \fi%
}
```

`\lst@Init` {⟨*backslash-processing*⟩} Done at the start of a listing.

```
\renewcommand{\lst@Init}[1]{%
```

Perform the listings initialization:

```
\LWR@traceinfo{lst@Init}%
\renewcommand*{\@captype}{lstlisting}%
\let\lst@aboveskip\z@\let\lst@belowskip\z@%
\gdef\lst@boxpos{t}%
\let\lst@frame\@empty
    \let\lst@frametshape\@empty
    \let\lst@framershape\@empty
    \let\lst@framebshape\@empty
    \let\lst@framelshape\@empty
\lstframe@\lst@frameround ffff\relax%
\lst@multicols\@empty%
\LWR@origlst@Init{#1}\relax%
```

Avoids extra horizontal space:

```
\def\lst@framelr{}%
```

```
\LWR@traceinfo{finished origlst@Init}%
\lst@ifdisplaystyle%
```

Creating a display.

Disable line numbers, produce the <pre>, then reenable line numbers.

```
\LWR@traceinfo{About to create verbatim.}%
\let\lsthk@EveryPar\relax%
\LWR@forcenewpage
\LWR@atbeginverbatim{2}{programlisting}%

\let\lsthk@EveryPar\LWR@origlsthkEveryPar%
\else%
```

Inline, so open a :

```
\ifbool{LWR@verbtags}{\LWR@htmltag{span class="inlineprogramlisting"}}{}%
\fi%
}

```

`\lst@DeInit` Done at the end of a listing.

```
\renewcommand*{\lst@DeInit}{%
\lst@ifdisplaystyle%
```

Creating a display.

Disable line numbers, produce the </pre>, then reenable line numbers:

```
\let\lsthk@EveryPar\relax%
\LWR@afterendverbatim{0}%
\let\lsthk@EveryPar\LWR@origlsthkEveryPar%
\else%
```

Inline, so create the closing :

```
\ifbool{LWR@verbtags}{\noindent\LWR@htmltag{/span}}{}%
\fi%
```

Final listings deinit:

```
\LWR@origlst@DeInit%
}
```

`\lst@MakeCaption` {⟨*t/b*⟩}

This is called BOTH at the top and at the bottom of each listing.

Patched for lwarp.

```
\def\lst@MakeCaption#1{%
\LWR@traceinfo{MAKING CAPTION at #1}%
  \lst@ifdisplaystyle
\LWR@traceinfo{making a listings display caption}%
    \ifx #1t%
         \ifx\lst@@caption\@empty\expandafter\lst@HRefStepCounter \else
                                    \expandafter\refstepcounter
         \fi {lstlisting}%
\LWR@traceinfo{About to assign label: !\lst@label!}%
%         \ifx\lst@label\@empty\else
% \label{\lst@label}\fi
\LWR@traceinfo{Finished assigning the label.}%
```

```
        \let\lst@arg\lst@intname \lst@ReplaceIn\lst@arg\lst@filenamerpl
        \global\let\lst@name\lst@arg \global\let\lstname\lst@name
        \lst@ifnolol\else
            \ifx\lst@@caption\@empty
                \ifx\lst@caption\@empty
                    \ifx\lst@intname\@empty \else \def\lst@temp{ }%
                    \ifx\lst@intname\lst@temp \else
```

This code places a contents entry for a non-float. This would have to be modified for lwarp:

```
\LWR@traceinfo{addcontents lst@name: -\lst@name-}%
%                           \addcontentsline{lol}{lstlisting}{\lst@name}
                    \fi\fi
                \fi
            \else
```

This would have to be modified for lwarp:

```
\LWR@traceinfo{addcontents lst@@caption: -\lst@@caption-}%
                 \addcontentsline{lol}{lstlisting}%
{\protect\numberline{\thelstlisting}%
{\protect\ignorespaces \LWR@isolate{\lst@@caption} \protect\relax}}%
            \fi
         \fi
     \fi
    \ifx\lst@caption\@empty\else
\LWR@traceinfo{lst@caption not empty-}%
        \lst@IfSubstring #1\lst@captionpos
            {\begingroup
\LWR@traceinfo{at the selected position}%
```

These space and box commands are not needed for HTML output:

```
%             \let\@@vskip\vskip
%             \def\vskip{\afterassignment\lst@vskip \@tempskipa}%
%             \def\lst@vskip{\nobreak\@@vskip\@tempskipa\nobreak}%
%             \par\@parboxrestore\normalsize\normalfont % \noindent (AS)
%             \ifx #1t\allowbreak \fi
             \ifx\lst@title\@empty
```

New lwarp code to create a caption:

```
                 \lst@makecaption\fnum@lstlisting{\ignorespaces \lst@caption}
             \else
```

New lwarp code to create a title:

```
%                 \lst@maketitle\lst@title % (AS)
\LWR@traceinfo{Making title: \lst@title}%
\begin{BlockClass}{lstlistingtitle}% lwarp
\lst@maketitle\lst@title% lwarp
\end{BlockClass}% lwarp
             \fi
\LWR@traceinfo{About to assign label: !\lst@label!}%
        \ifx\lst@label\@empty\else
\leavevmode% gets rid of bad space factor error
\GetTitleStringExpand{\lst@caption}%
\edef\LWR@lntemp{\GetTitleStringResult}%
\edef\@currentlabelname{\detokenize\expandafter{\LWR@lntemp}}%
```

```
\label{\lst@label}\fi
\LWR@traceinfo{Finished assigning the label.}%
```

Not needed for lwarp:

```
%              \ifx #1b\allowbreak \fi
             \endgroup}{}%
    \fi
\LWR@traceinfo{end of making a listings display caption}%
  \else
\LWR@traceinfo{INLINE}%
  \fi
\LWR@traceinfo{DONE WITH CAPTION at #1}%
}
```

line numbers Patched to keep left line numbers outside of the left margin, and place right line numbers in a field `\VerbatimHTMLWidth` wide.

```
\lst@Key{numbers}{none}{%
    \let\lst@PlaceNumber\@empty
    \lstKV@SwitchCases{#1}%
    {none&\\%
     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.

```
%         \LWR@origllap{
            \LWR@orignormalfont%
            \lst@numberstyle{\thelstnumber}\kern\lst@numbersep%
%         }
    }\\%
     right&\def\lst@PlaceNumber{\LWR@origrlap{\LWR@orignormalfont
                \kern 6in \kern\lst@numbersep
                \lst@numberstyle{\thelstnumber}}}%
    }{\PackageError{Listings}{Numbers #1 unknown}\@ehc}}

\end{warpHTML}
```

File 185 **lwarp-lltjext.sty**

§ 282 Package

lltjext

(*Emulates or patches code by* The LuaTeX-ja project team.)

Pkg lltjext lltjext is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{lltjext}[2018/10/07]

\protected\def\yoko{%
  \directlua{luatexja.direction.set_list_direction(4, 'yoko')}%
}
```

```
\protected\def\tate{\yoko}
\protected\def\dtou{\yoko}
\protected\def\utod{\yoko}

\define@key[ltj]{japaram}{direction}{}

\yoko

\DeclareExpandableDocumentCommand{\rensuji}{s o m}{#3}

\DeclareDocumentCommand{\layoutfloat}{d() o m}{}

\DeclareDocumentCommand{\DeclareLayoutCaption}{m d<> d() o}{}

\LetLtxMacro\pcaption\caption

\DeclareDocumentCommand{\layoutcaption}{d<> d() o}{}

\let\captiondir\relax
\RenewDocumentEnvironment{LWR@HTML@minipage}{d<> O{t} O{} O{t} m}
    {\LWR@HTML@sub@minipage{#2}{#3}{#4}{#5}}
    {\endLWR@HTML@sub@minipage}

\RenewDocumentCommand{\LWR@HTML@parbox}{d<> O{t} O{} O{t} m +m}
{
\LWR@traceinfo{parbox of width #4}%
\begin{minipage}[#2][#3][#4]{#5}%
#6
\end{minipage}%
}

\RenewDocumentCommand{\pbox}{d<> O{0pt} O{c} m}{%
\global\booltrue{LWR@minipagefullwidth}%
\parbox{#2}{#4}%
}
```

File 186 **lwarp-longtable.sty**

§ 283 Package

longtable

(*Emulates or patches code by* David Carlisle.)

Pkg longtable longtable is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{longtable}[2014/10/28]
```

⚠ Misplaced \noalign

Longtable `\endhead`, `\endfoot`, and `\endlastfoot` rows are not used for HTML, and these rows should be disabled. Use

```
\warpprintonly{row contents}
```

instead of

```
\begin{warpprint} . . . \end{warpprint}
```

Doing so helps avoid “Misplaced \noalign.” when using \begin{warpprint}.

Keep the \endfirsthead row, which is still relevent to HTML output.

⚠ \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

Env longtable * [⟨*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 thrown away.

```
\newenvironment{longtable*}[2][]{%
\LWR@floatbegin{table}%
\setcaptiontype{\LTcaptype}%
\caption@setoptions{longtable}%
\caption@setoptions{@longtable}%
\caption@LT@setup%
\booltrue{LWR@starredlongtable}%
\let\captionlistentry\LWR@LTcaptionlistentry%
\tabular{#2}
}
{\endtabular\LWR@floatend}

\newenvironment{longtable}[2][]{%
\LWR@floatbegin{table}%
\setcaptiontype{\LTcaptype}%
\caption@setoptions{longtable}%
\caption@setoptions{@longtable}%
\caption@LT@setup%
\refstepcounter{\LTcaptype}%
\let\captionlistentry\LWR@LTcaptionlistentry%
\tabular{#2}
}
{\endtabular\LWR@floatend}
```

Provided for compatibility, but ignored:

```
\newcounter{LTchunksize}
\def\endhead{\LWR@tabularendofline}% throws away options //[dim] and //*
\def\endfirsthead{\LWR@tabularendofline}
\def\endfoot{\LWR@tabularendofline}
```

```
\def\endlastfoot{\LWR@tabularendofline}
\newcommand\tabularnewline{\LWR@tabularendofline}
\newcommand{\setlongtables}{}% Obsolete command, does nothing.
\newlength{\LTleft}
\newlength{\LTright}
\newlength{\LTpre}
\newlength{\LTpost}
\newlength{\LTcapwidth}

\LetLtxMacro\LWR@origkill\kill
\renewcommand*{\kill}{\LWR@tabularendofline}
\appto\LWR@restoreorigformatting{%
\LetLtxMacro\kill\LWR@origkill%
}
```

File 187 **lwarp-lscape.sty**

§ 284 Package lscape

(*Emulates or patches code by* D. P. Carlisle.)

Pkg lscape lscape is emulated.

for HTML output: Discard all options for lwarp-lscape.

```
\LWR@ProvidesPackageDrop{lscape}[2000/10/22]

\newenvironment*{landscape}{}{}
```

File 188 **lwarp-ltablex.sty**

§ 285 Package ltablex

(*Emulates or patches code by* Anil K. Goel.)

Pkg ltablex ltablex is emulated by lwarp.

for HTML output: Relies on tabularx.

```
\RequirePackage{tabularx}

\LWR@ProvidesPackageDrop{ltablex}[2014/08/13]

\DeclareDocumentEnvironment{tabularx}{m o m}
{\longtable{#3}}
{\endlongtable}

\DeclareDocumentEnvironment{tabularx*}{m o m}
{\longtable{#3}}
{\endlongtable}
```

```
12
13 \newcommand*{\keepXColumns}{}
14 \newcommand*{\convertXColumns}{}
```

File 189 **lwarp-ltcaption.sty**

§ 286 Package ltcaption

(*Emulates or patches code by* Axel Sommerfeldt.)

Pkg ltcaption ltcaption is emulated.

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 190 **lwarp-ltxgrid.sty**

§ 287 Package ltxgrid

Pkg ltxgrid 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 191 **lwarp-ltxtable.sty**

§ 288 Package ltxtable

Pkg ltxtable 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 \LWR@ProvidesPackageDrop{ltxtable}[1995/12/11]
```

`\LTXtable` {⟨*width*⟩} {⟨*file*⟩}

```
\newcommand*{\LTXtable}[2]{%
\input{#2}%
}
```

File 192 **`lwarp-lua-check-hyphen.sty`**

§ 289 Package

lua-check-hyphen

Pkg `lua-check-hyphen` lua-check-hyphen is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{lua-check-hyphen}[2018/04/19]
\newcommand*{\LuaCheckHyphen}[1]{}
```

File 193 **`lwarp-lua-visual-debug.sty`**

§ 290 Package

lua-visual-debug

Pkg `lua-visual-debug` lua-visual-debug is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{lua-visual-debug}[2016/05/30]
```

File 194 **`lwarp-luacolor.sty`**

§ 291 Package

luacolor

Pkg `luacolor` luacolor is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{luacolor}[2016/05/16]
\newcommand{\luacolorProcessBox}[1]{}
```

File 195 **`lwarp-luatodonotes.sty`**

§ 292 Package

luatodonotes

(*Emulates or patches code by* Fabian Lipp.)

Pkg `luatodonotes` 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:

```
\LWR@ProvidesPackagePass{luatodonotes}[2017/09/30]
```

Nullify options:

```
\@todonotes@additionalMarginEnabledfalse

\if@todonotes@disabled
\else

\newcommand{\ext@todo}{tdo}

\renewcommand{\l@todo}[2]{\hypertocfloat{1}{todo}{ldo}{#1}{#2}}


\let\LWRTODONOTES@orig@todototoc\todototoc

\renewcommand*{\todototoc}{%
\phantomsection%
\LWRTODONOTES@orig@todototoc%
}


\renewcommand{\@todonotes@drawMarginNoteWithLine}{%
\fcolorbox
    {\@todonotes@currentbordercolor}
    {\@todonotes@currentbackgroundcolor}
    {\arabic{@todonotes@numberoftodonotes}}
\marginpar{\@todonotes@drawMarginNote}
}

\renewcommand{\@todonotes@drawInlineNote}{%
\fcolorboxBlock%
    {\@todonotes@currentbordercolor}%
    {\@todonotes@currentbackgroundcolor}%
    {%
        \if@todonotes@authorgiven%
        {\@todonotes@author:\,}%
        \fi%
        \@todonotes@text%
    }%
}

\newcommand{\@todonotes@drawMarginNote}{%
    \if@todonotes@authorgiven%
        \@todonotes@author\par%
    \fi%
    \arabic{@todonotes@numberoftodonotes}: %
    \fcolorbox%
    {\@todonotes@currentbordercolor}%
    {\@todonotes@currentbackgroundcolor}%
    {%
```

```
        \@todonotes@sizecommand%
        \@todonotes@text %
    }%
}%

\renewcommand{\missingfigure}[2][]{%
\setkeys{todonotes}{#1}%
\addcontentsline{tdo}{todo}{\@todonotes@MissingFigureText: #2}%
\fcolorboxBlock%
    {\@todonotes@currentbordercolor}%
    {\@todonotes@currentfigcolor}%
    {%
        \setlength{\fboxrule}{4pt}%
        \fcolorbox{red}{white}{Missing figure} \quad #2%
    }
}

\LetLtxMacro\LWRTODONOTES@orig@todocommon\@todocommon

\RenewDocumentCommand{\@todocommon}{m m}{%
\begingroup%
\renewcommand*{\phantomsection}{}%
\LWRTODONOTES@orig@todocommon{#1}{#2}%
\endgroup%
}

\renewcommand{\@todoarea}[3][]{%
    \@todonotes@areaselectedtrue%
    \@todocommon{#1}{#2}%
    \todonotes@textmark@highlight{#3}%
    \zref@label{@todonotes@\arabic{@todonotes@numberoftodonotes}@end}%
}%


\DeclareDocumentCommand{\todonotes@textmark@highlight}{m}{%
\InlineClass[background:\LWR@origpound{}B3FFB3]{highlight}{#1}%
}

\fi% \if@todonotes@disabled
```

File 196 **lwarp-magaz.sty**

§ 293 Package

magaz

Pkg magaz magaz is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{magaz}[2011/11/24]

\newcommand\FirstLine[1]{%
    \begingroup%
    \FirstLineFont{%
        \LWR@textcurrentcolor{%
            \LWR@textcurrentfont{%
```

```
                #1%
            }%
        }%
    }%
    \endgroup%
}

\providecommand\FirstLineFont{\scshape}
```

File 197 **lwarp-makeidx.sty**

§ 294 Package makeidx

(*Emulates or patches code by* LaTeX Project Team.)

Pkg makeidx makeidx is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{makeidx}[2014/09/29]
```

`\@wrindex` is redefined `\AtBeginDocument` by the lwarp core.

`\printindex`

```
\preto\printindex{%
    \LWR@orignewpage%
    \LWR@startpars%
}
```

File 198 **lwarp-marginal.sty**

§ 295 Package marginal

Pkg marginal marginal is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{marginal}

\newcommand*{\showlostmarginals}{}
\newcommand*{\enlargefreelist}{}
\newcommand*{\onesidemarginals}{}
```

File 199 **lwarp-marginfit.sty**

§ 296 Package marginfit

Pkg marginfit marginfit is ignored.

for HTML output: Discard all options for lwarp-marginfit:

```
\LWR@ProvidesPackageDrop{marginfit}[2018/06/08]
```

File 200 **lwarp-marginfix.sty**

§ 297 Package marginfix

(*Emulates or patches code by* Stephen Hicks.)

Pkg marginfix Emulated.

for HTML output: Discard all options for lwarp-marginfix:

```
\LWR@ProvidesPackageDrop{marginfix}[2013/09/08]

\newcommand*{\marginskip}[1]{}
\newcommand*{\clearmargin}{}
\newcommand*{\softclearmargin}{}
\newcommand*{\extendmargin}[1]{}
\newcommand*{\mparshift}[1]{}
\newdimen\marginheightadjustment
\newdimen\marginposadjustment
\newcommand*{\blockmargin}[1][]{}
\newcommand*{\unblockmargin}[1][]{}
\newcommand*{\marginphantom}[2][]{}
```

File 201 **lwarp-marginnote.sty**

§ 298 Package marginnote

(*Emulates or patches code by* Markus Kohm.)

Pkg marginnote Emulated.

for HTML output: Discard all options for lwarp-marginnote:

```
\LWR@ProvidesPackageDrop{marginnote}[2018/08/09]

\NewDocumentCommand{\marginnote}{+o +m o}{\marginpar{#2}}

\newcommand*{\marginnoteleftadjust}{}
\newcommand*{\marginnoterightadjust}{}
\newcommand*{\marginnotetextwidth}{}
\let\marginnotetextwidth\textwidth
\newcommand*{\marginnotevadjust}{}
\newcommand*{\marginfont}{}
\newcommand*{\raggedleftmarginnote}{}
\newcommand*{\raggedrightmarginnote}{}
```

File 202 **lwarp-marvosym.sty**

§ 299 Package marvosym

(*Emulates or patches code by* Thomas Henlich, Mojca Miklavec.)

Pkg marvosym 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:

```
\LWR@ProvidesPackagePass{marvosym}[2011/07/20]

\renewcommand{\mvchr}[1]{%
    \begin{lateximage}*[(symbol #1)][marvosym #1]%
    \mvs\char#1%
    \end{lateximage}%
}

\renewcommand{\textmvs}[1]{%
    \begin{lateximage}%
    \mvs #1%
    \end{lateximage}%
}
```

File 203 **lwarp-mcaption.sty**

§ 300 Package mcaption

(*Emulates or patches code by* Stephan Hennig.)

Pkg mcaption mcaption is nullified.

for HTML output: Discard all options for lwarp-mcaption:

```
\LWR@ProvidesPackageDrop{mcaption}[2009/03/13]

\newenvironment{margincap}{}{}
\newcommand*{\margincapalign}{}
\newlength{\margincapsep}
```

File 204 **lwarp-mdframed.sty**

§ 301 Package mdframed

(*Emulates or patches code by* Marco Daniel, Elke Schubert.)

Pkg mdframed mdframed is loaded with options forced to `framemethod=none`.

§ 301.1 Limitations

support Most basic functionality is supported, including frame background colors and single-border colors and thickness, title and subtitle background colors and borders and thickness, border radius, and shadow. CSS classes are created for mdframed environments and frame titles.

⚠ loading When used, lwarp loads mdframed in HTML with `framemethod=none`.

font For title font, use

```
frametitlefont=\textbf,
```

instead of

```
frametitlefont=\bfseries,
```

where `\textbf` must appear just before the comma and will receive the following text as its argument (since the text happens to be between braces in the mdframed source). Since lwarp does not support `\bfseries` and friends, only one font selection may be made at a time.

theoremtitlefont `theoremtitlefont` is not supported, since the following text is not in braces in the mdframed source.

ignored options `userdefinedwidth` and `align` are currently ignored.

css classes Environments created or encapsulated by mdframed are enclosed in a `<div>` of class `mdframed`, and also class `md<environmentname>` for new environments.

Frame titles are placed in a `<div>` of class |mdframedtitle|. Subtitles are in a `<div>` of class |mdframedsubtitle|, and likewise for subsubtitles.

Pre-existing hooks are used to patch extra functions before and after the frames.

§ 301.2 Package loading

for HTML output:

```
\RequirePackage{xcolor}% for \convertcolorspec

\LWR@ProvidesPackageDrop{mdframed}[2013/07/01]
```

Do not require Ti*k*z or pstricks:

```
\LWR@origRequirePackage[framemethod=none]{mdframed}
```

§ 301.3 Patches

Patch to remove PDF formatting and add HTML tags:

```
\AtBeginDocument{
\def\mdf@trivlist#1{%
  \edef\mdf@temp{%
%     \topsep=\the\topsep\relax%
%     \partopsep=\the\partopsep\relax%
%     \parsep=\the\parsep\relax%
```

```
  }%
%   \setlength{\topsep}{#1}%
%   \topskip\z@%
%   \partopsep\z@%
%   \parsep\z@%
%   \@nmbrlistfalse%
%   \@trivlist%
%   \labelwidth\z@%
%   \leftmargin\z@%
%   \itemindent\z@%
  \let\@itemlabel\@empty%
  \def\makelabel##1{##1}%
%   \item\relax\mdf@temp\relax%
}

\renewcommand*{\endmdf@trivlist}{%
\LWR@traceinfo{endmdf@trivlist}%
% \endtrivlist%
\LWR@listend%
}
}% AtBeginDocument
```

§ 301.4 Initial setup

To handle CSS and paragraphs, patch code at start and end of environment and contents. `\LWR@print@raggedright` helps avoid hyphenation.

```
\mdfsetup{
startcode={\LWR@mdframedstart\LWR@print@raggedright},
endcode={\LWR@mdframedend},
startinnercode={\LWR@startpars\LWR@print@raggedright},
endinnercode={\LWR@stoppars},
}
```

§ 301.5 Color and length HTML conversion

`\LWR@mdfprintcolor` {⟨*mdfcolorkey*⟩}

Given the mdframed key, print the color.

```
\newcommand*{\LWR@mdfprintcolor}[1]{%
\convertcolorspec{named}{\@nameuse{mdf@#1}}{HTML}\LWR@tempcolor%
\LWR@origpound\LWR@tempcolor
}
```

`\LWR@mdfprintlength` {⟨*mdflengthkey*⟩}

Given the mdframed key, print the length.

```
\newcommand*{\LWR@mdfprintlength}[1]{%
\LWR@forceminwidth{\@nameuse{mdf@#1@length}}%
\LWR@printlength{\LWR@atleastonept}%
}
```

§ 301.6 Environment encapsulation

\LWR@mdframedstart Actions before an mdframe starts.

Encapsulate a frame inside a of the desired class.

```
\newcommand*{\LWR@mdframedstart}{%
\LWR@traceinfo{LWR@mdframedstart start}%
```

Turn off paragraph handling during the generation of the encapsulating tags:

```
\LWR@stoppars%
```

Open a and with custom class and custom style. A BlockClass environment is not used because this is created by the mdframed startcode and endcode settings, which do not properly nest the inside the mdframed environment.

```
\LWR@htmltagc{div class="%
mdframed%
\ifdefstring{\LWR@mdthisenv}{mdframed}{}{ \LWR@mdthisenv}%
" \LWR@orignewline
style=" \LWR@orignewline
```

Convert and print the background color:

```
background: \LWR@mdfprintcolor{backgroundcolor} ; \LWR@orignewline
```

Convert and print the border color and width:

```
border: \LWR@mdfprintlength{linewidth} solid
\LWR@mdfprintcolor{linecolor} ; \LWR@orignewline
```

Convert and print the border radius:

```
border-radius: \LWR@mdfprintlength{roundcorner} ; \LWR@orignewline
```

Convert and print the shadow:

```
\ifbool{mdf@shadow}{%
    box-shadow:
    \LWR@mdfprintlength{shadowsize}
    \LWR@mdfprintlength{shadowsize}
    \LWR@mdfprintlength{shadowsize}
    \LWR@mdfprintcolor{shadowcolor} ;
}
{box-shadow: none ;}
\LWR@orignewline

"}
% \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:

```
\LWR@select@print@hspace%
\renewcommand*{\rule}{\LWR@print@rule}
\LetLtxMacro\makebox\LWR@print@makebox%

\LWR@startpars%
\LWR@traceinfo{LWR@mdframedstart done}%
}
```

\LWR@mdframedend Actions after an mdframe ends.

After closing the , globally restore to the default environment type:

```
\newcommand*{\LWR@mdframedend}{
\LWR@traceinfo{LWR@mdframedend start}%
```

Close the custom :

```
\LWR@htmldivclassend{\LWR@mdthisenv}
```

Reset future custom class to the default:

```
\gdef\LWR@mdthisenv{mdframed}
```

Resume paragraph handling:

```
\LWR@startpars%
\LWR@traceinfo{LWR@mdframedend done}%
}
```

§ 301.7 Mdframed environment

```
\renewenvironment{mdframed}[1][]{%
 \color@begingroup%
   \mdfsetup{userdefinedwidth=\linewidth,#1}%
   \mdf@startcode%
   \mdf@preenvsetting%
   \ifdefempty{\mdf@firstframetitle}{}%
            {\let\mdf@frametitlesave\mdf@frametitle%
             \let\mdf@frametitle\mdf@firstframetitle%
            }%
   \ifvmode\nointerlineskip\fi%
        \ifdefempty{\mdf@frametitle}{}%
            {\mdfframedtitleenv{\mdf@frametitle}%
%             \mdf@@frametitle@use%
            }%
   \mdf@trivlist{\mdf@skipabove@length}%%
   \mdf@settings%
%    \mdf@lrbox{\mdf@splitbox@one}%
%    \mdf@startinnercode%
  }%
  {%
%    \mdf@@ignorelastdescenders%
   \par%
%     \unskip\ifvmode\nointerlineskip\hrule \@height\z@ \@width\hsize\fi%%
   \ifmdf@footnoteinside%
      \def\mdf@reserveda{%
        \mdf@footnoteoutput%
%         \mdf@endinnercode%
%         \endmdf@lrbox%
%         \ifdefempty{\mdf@frametitle}{}%
%             {\mdfframedtitleenv{\mdf@frametitle}\mdf@@frametitle@use}%
%         \detected@mdf@put@frame
      }%
   \else%
      \def\mdf@reserveda{%
%         \mdf@endinnercode%
%         \endmdf@lrbox%
```

```
%         \ifdefempty{\mdf@frametitle}{}%
%             {\mdfframedtitleenv{\mdf@frametitle}\mdf@@frametitle@use}%
%         \detected@mdf@put@frame%
        \mdf@footnoteoutput%
        }%
   \fi%
   \mdf@reserveda%
  \aftergroup\endmdf@trivlist%
 \color@endgroup%
 \mdf@endcode%
}
```

\mdf@footnoteoutput

```
\renewrobustcmd*\mdf@footnoteoutput{%
    \LWR@printpendingmpfootnotes%
}
```

§ 301.8 Titles and subtitles

\mdfframedtitleenv {⟨*title*⟩}

Place the title inside a of class mdframedtitle:

```
\newlength{\LWR@titleroundcorner}

\renewrobustcmd\mdfframedtitleenv[1]{%
\LWR@traceinfo{LWR@mdframedtitleenv start}%
```

Open a with a custom class and custom style:

```
\begin{BlockClass}[%
```

Convert and print the title background color:

```
background:
\LWR@mdfprintcolor{frametitlebackgroundcolor}
; \LWR@orignewline
```

Convert and print the title rule:

```
\ifbool{mdf@frametitlerule}{%
    border-bottom:
    \LWR@mdfprintlength{frametitlerulewidth}
    solid
    \LWR@mdfprintcolor{frametitlerulecolor}
    ; \LWR@orignewline
}{}%
```

Finish the custom style and the opening tag:

```
]{mdframedtitle}%
```

Print the title inside the :

```
\mdf@frametitlefont{\LWR@textcurrentfont{#1}}%
```

Close the :

```
\end{BlockClass}%
\LWR@traceinfo{LWR@mdframedtitleenv end}%
}
```

\LWR@mdfsubtitlecommon {⟨*sub -or- subsub*⟩} [⟨*options*⟩] {⟨*title*⟩}

Common code for \LWR@mdfsubtitle and \LWR@mdfsubsubtitle.

Encapsulate the subtitle inside a of class mdframedsubtitle:

```
\NewDocumentCommand{\LWR@mdfsubtitlecommon}{m o m}
{% the following empty line is required

\LWR@traceinfo{LWR@mdframedsubtitlecommon start}%
```

Open a with a custom class and custom style:

```
\begin{BlockClass}[%
```

Convert and print the background color:

```
background:
\LWR@mdfprintcolor{#1titlebackgroundcolor}
; \LWR@orignewline
```

Convert and print the above line:

```
\ifbool{mdf@#1titleaboveline}{%
    border-top:
    \LWR@mdfprintlength{#1titleabovelinewidth}
    solid
    \LWR@mdfprintcolor{#1titleabovelinecolor}
    ; \LWR@orignewline
}{}%
```

Convert and print the below line:

```
\ifbool{mdf@#1titlebelowline}{%
    border-bottom:
    \LWR@mdfprintlength{#1titlebelowlinewidth}
    solid
    \LWR@mdfprintcolor{#1titlebelowlinecolor}
    ; \LWR@orignewline
}{}%
```

Finish the custom style and the opening tag:

```
]{mdframed#1title}%
```

Perform the original subtitle action:

```
\IfNoValueTF{#2}
{\@nameuse{LWR@origmdf#1title}{\csuse{mdf@#1titlefont}{\LWR@textcurrentfont{#3}}}}%
{\@nameuse{LWR@origmdf#1title}[#2]{\csuse{mdf@#1titlefont}{\LWR@textcurrentfont{#3}}}}%
```

Close the :

```
\end{BlockClass}%
\LWR@traceinfo{LWR@mdframedsubtitlecommon end}%
}
```

\LWR@mdfsubtitle [⟨*options*⟩] {⟨*title*⟩}

```
\newcommand*{\LWR@mdfsubtitle}{%
\LWR@mdfsubtitlecommon{sub}%
```

```
}
\let\mdfsubtitle\LWR@mdfsubtitle
```

`\LWR@mdfsubsubtitle` `[⟨options⟩] {⟨title⟩}`

```
\newcommand*{\LWR@mdfsubsubtitle}{%
\LWR@mdfsubtitlecommon{subsub}%
}
\let\mdfsubsubtitle\LWR@mdfsubsubtitle
```

§ 301.9 New environments

`\LWR@mdthisenv` Stores the environment of the frame about to be created:

```
\newcommand*{\LWR@mdthisenv}{mdframed}
```

`\newmdenv` `[⟨options⟩] {⟨env-name⟩}`

Modified from the original to remember the environment.

```
\renewrobustcmd*\newmdenv[2][]{%
\newenvironment{#2}%
{%
\mdfsetup{#1}%
\renewcommand*{\LWR@mdthisenv}{md#2}%
\begin{mdframed}%
}
{\end{mdframed}}%
}
```

`\surroundwithmdframed` `[⟨options⟩] {⟨environment⟩}`

Modified from the original to remember the environment.

```
\renewrobustcmd*{\surroundwithmdframed}[2][]{%
\BeforeBeginEnvironment{#2}{%
\renewcommand*{\LWR@mdthisenv}{md#2}%
\begin{mdframed}[#1]}%
\AfterEndEnvironment{#2}{\end{mdframed}}%
}
```

`\mdtheorem` `[⟨mdframed-options⟩] {⟨envname⟩} [⟨numberedlike⟩] {⟨caption⟩} [⟨within⟩]`

Modified from the original to remember the environment.

```
\DeclareDocumentCommand{\mdtheorem}{ O{} m o m o }%
 {\ifcsdef{#2}%
   {\mdf@PackageWarning{Environment #2 already exits\MessageBreak}}%
   {%
    \IfNoValueTF {#3}%
     {%#3 not given -- number relationship
      \IfNoValueTF {#5}%
        {%#3+#5 not given
        \@definecounter{#2}%
        \expandafter\xdef\csname the#2\endcsname{\@thmcounter{#2}}%
        \newenvironment{#2}[1][]{%
          \refstepcounter{#2}%
```

```
      \ifstrempty{##1}%
        {\let\@temptitle\relax}%
        {%
         \def\@temptitle{\mdf@theoremseparator%
                         \mdf@theoremspace%
                         \mdf@theoremtitlefont%
                         \LWR@textcurrentfont{##1}}% lwarp
         \mdf@thm@caption{#2}{{#4}{\csname the#2\endcsname}{##1}}%
         }%
      \begin{mdframed}[#1,frametitle={\strut#4\ \csname the#2\endcsname%
                                      \@temptitle}]}%
      {\end{mdframed}}%
    \newenvironment{#2*}[1][]{%
      \ifstrempty{##1}{\let\@temptitle\relax}{\def\@temptitle{:\ ##1}}%
      \begin{mdframed}[#1,frametitle={\strut#4\@temptitle}]}%
      {\end{mdframed}}%
    }%
    {%#5 given -- reset counter
    \@definecounter{#2}\@newctr{#2}[#5]%
    \expandafter\xdef\csname the#2\endcsname{\@thmcounter{#2}}%
    \expandafter\xdef\csname the#2\endcsname{%
           \expandafter\noexpand\csname the#5\endcsname \@thmcountersep%
              \@thmcounter{#2}}%
    \newenvironment{#2}[1][]{%
      \refstepcounter{#2}%
      \ifstrempty{##1}%
        {\let\@temptitle\relax}%
        {%
         \def\@temptitle{\mdf@theoremseparator%
                         \mdf@theoremspace%
                         \mdf@theoremtitlefont%
                         \LWR@textcurrentfont{##1}}% lwarp
         \mdf@thm@caption{#2}{{#4}{\csname the#2\endcsname}{##1}}%
         }
      \begin{mdframed}[#1,frametitle={\strut#4\ \csname the#2\endcsname%
                                      \@temptitle}]}%
      {\end{mdframed}}%
    \newenvironment{#2*}[1][]{%
      \ifstrempty{##1}%
        {\let\@temptitle\relax}%
        {%
         \def\@temptitle{\mdf@theoremseparator%
                         \mdf@theoremspace%
                         \mdf@theoremtitlefont%
                         \LWR@textcurrentfont{##1}}% lwarp
         \mdf@thm@caption{#2}{{#4}{\csname the#2\endcsname}{##1}}%
         }%
      \begin{mdframed}[#1,frametitle={\strut#4\@temptitle}]}%
      {\end{mdframed}}%
    }%
  }%
  {%#3 given -- number relationship
     \global\@namedef{the#2}{\@nameuse{the#3}}%
     \newenvironment{#2}[1][]{%
       \refstepcounter{#3}%
```

```
          \ifstrempty{##1}%
            {\let\@temptitle\relax}%
            {%
             \def\@temptitle{\mdf@theoremseparator%
                             \mdf@theoremspace%
                             \mdf@theoremtitlefont%
                             \LWR@textcurrentfont{##1}}% lwarp
             \mdf@thm@caption{#2}{{#4}{\csname the#2\endcsname}{##1}}%
             }
          \begin{mdframed}[#1,frametitle={\strut#4\ \csname the#2\endcsname%
                                          \@temptitle}]}%
          {\end{mdframed}}%
        \newenvironment{#2*}[1][]{%
          \ifstrempty{##1}{\let\@temptitle\relax}{\def\@temptitle{:\ ##1}}%
          \begin{mdframed}[#1,frametitle={\strut#4\@temptitle}]}%
          {\end{mdframed}}%
     }%
    \BeforeBeginEnvironment{#2}{\renewcommand*{\LWR@mdthisenv}{md#2}}% lwarp
    \BeforeBeginEnvironment{#2*}{\renewcommand*{\LWR@mdthisenv}{md#2}}% lwarp
   }%
}
```

\newmdtheoremenv `[`⟨*1: mdframed-options*⟩`]` `{`⟨*2: envname*⟩`}` `[`⟨*3: numberedlike*⟩`]` `{`⟨*4: caption*⟩`}` `[`⟨*5: within*⟩`]`

Modified from the original to remember the environment.

```
\DeclareDocumentCommand\newmdtheoremenv{O{} m o m o }{%
 \ifboolexpr{ test {\IfNoValueTF {#3}} and test {\IfNoValueTF {#5}} }%
    {\newtheorem{#2}{#4}}%
    {%
     \IfValueT{#3}{\newtheorem{#2}[#3]{#4}}%
     \IfValueT{#5}{\newtheorem{#2}{#4}[#5]}%
    }%
 \BeforeBeginEnvironment{#2}{%
 \renewcommand*{\LWR@mdthisenv}{md#2}%
 \begin{mdframed}[#1]}%
 \AfterEndEnvironment{#2}{%
 \end{mdframed}}%
}
```

File 205 **lwarp-memhfixc.sty**

§ 302 Package

memhfixc

Pkg memhfixc memhfixc is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{memhfixc}[2013/05/30]
```

File 206 **lwarp-metalogo.sty**

§ 303 Package metalogo

(*Emulates or patches code by* Andrew Gilbert Moschou.)

Pkg metalogo metalogo is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{metalogo}[2010/05/29]

\newcommand\setlogokern[2]{}
\newcommand\setlogodrop[2][XeTeX]{}
\newcommand\setLaTeXa[1]{}
\newcommand\setLaTeXee[1]{}
\newcommand\seteverylogo[1]{}
\newcommand\everylogo[1]{}
```

File 207 **lwarp-mhchem.sty**

§ 304 Package mhchem

(*Emulates or patches code by* Martin Hensel.)

Pkg mhchem mhchem is patched for use by lwarp.

mhchem expressions are converted to SVG math. Inline expressions use hashed filenames to allow reuse, and assume that any mhchem options are global.

⚠ MathJax and mhchem The MathJax mhchem extension is not yet used. If MathJax is used for math in the rest of the document, lwarp converts standalone mhchem expressions into SVG math images, but expressions inside math must be placed between `\displaymathother` and `\displaymathnormal`:

```
\displaymathother
\[ \ce{ . . . } \]      . . .      $ \ce { . . . } $
\displaymathnormal
```

⚠ nested math When producing HTML output, 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:

```
\LWR@ProvidesPackagePass{mhchem}[2018/06/22]
```

The original definition of \ce:

```
\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.

```
\newcommand{\LWR@mhchem@HTML@ce}[1]{%
\begin{lateximage}*[\textbackslash{}ce\{\LWR@HTMLsanitize{#1}\}]%
\LWR@mhchem@origce{#1}%
\end{lateximage}%
\endgroup%
\addtocounter{LWR@mhchem@cedepth}{-1}%
}
```

Only set math shift if outer depth:

```
\newcounter{LWR@mhchem@cedepth}
\setcounter{LWR@mhchem@cedepth}{0}
```

The new \ce. Sets math shift then continues.

```
\renewcommand{\ce}{%
\begingroup%
\ifnumequal{\value{LWR@mhchem@cedepth}}{0}{%
    \catcode`\$=3% math shift
}{}%
\addtocounter{LWR@mhchem@cedepth}{1}%
\LWR@mhchem@HTML@ce%
}
```

The original definition of \cesplit:

```
\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.

```
\newcommand*{\LWR@mhchem@HTML@cesplit}[2]
```

```
{%
\begin{lateximage}*[\textbackslash{}cesplit\{\LWR@HTMLsanitize{#2}\}]%
\LWR@mhchem@origcesplit{#1}{#2}%
\end{lateximage}%
\endgroup%
}
```

Only set math shift if outer depth:

```
\newcounter{LWR@mhchem@cesplitdepth}
\setcounter{LWR@mhchem@cesplitdepth}{0}
```

The new `\cesplit`. Sets math shift then continues.

```
\renewcommand{\cesplit}{%
\begingroup%
\ifnumequal{\value{LWR@mhchem@cesplitdepth}}{0}{%
    \catcode`\$=3% math shift
}{}%
\addtocounter{LWR@mhchem@cesplitdepth}{1}%
\LWR@mhchem@HTML@cesplit%
}
```

Resore originals inside a `lateximage`:

```
\appto\LWR@restoreorigformatting{%
\LetLtxMacro\ce\LWR@mhchem@origce%
\LetLtxMacro\cesplit\LWR@mhchem@origcesplit%
}
```

File 208 **lwarp-microtype.sty**

§ 305 Package

microtype

(*Emulates or patches code by* R Schlicht.)

Pkg microtype

microtype is pre-loaded by lwarp. All user options and macros are ignored and disabled.

for HTML output:

Discard all options for lwarp-microtype:

```
\LWR@ProvidesPackageDrop{microtype}[2018/01/14]
```

```
\DeclareDocumentCommand{\DeclareMicrotypeSet}{o m m}{}
\DeclareDocumentCommand{\UseMicrotypeSet}{o m}{}
\DeclareDocumentCommand{\DeclareMicrotypeSetDefault}{o m}{}
\DeclareDocumentCommand{\SetProtrusion}{o m m}{}
\DeclareDocumentCommand{\SetExpansion}{o m m}{}
\DeclareDocumentCommand{\SetTracking}{o m m}{}
\DeclareDocumentCommand{\SetExtraKerning}{o m m}{}
\DeclareDocumentCommand{\SetExtraSpacing}{o m m}{}
\DeclareDocumentCommand{\DisableLigatures}{o m}{}
```

```
\DeclareDocumentCommand{\DeclareCharacterInheritance}{o m m}{}
\DeclareDocumentCommand{\DeclareMicrotypeVariants}{m}{}
\DeclareDocumentCommand{\DeclareMicrotypeAlias}{m m}{}
\DeclareDocumentCommand{\LoadMicrotypeFile}{m}{}
\DeclareDocumentCommand{\DeclareMicrotypeBabelHook}{m m}{}
\DeclareDocumentCommand{\microtypesetup}{m}{}
\DeclareDocumentCommand{\microtypecontext}{m}{}
\DeclareDocumentCommand{\textmicrotypecontext}{m m}{#2}
\@ifpackageloaded{letterspace}{\let\MT@textls\relax}{%
\DeclareDocumentCommand{\lsstyle}{}{}
\DeclareDocumentCommand{\textls}{o +m}{}
\DeclareDocumentCommand{\lslig}{m}{#1}
}
\def\DeclareMicrotypeSet#1#{\@gobbletwo}
\def\DeclareMicrotypeVariants#1#{\@gobble}
\@onlypreamble\DeclareMicrotypeSet
\@onlypreamble\UseMicrotypeSet
\@onlypreamble\DeclareMicrotypeSetDefault
\@onlypreamble\DisableLigatures
\@onlypreamble\DeclareMicrotypeVariants
\@onlypreamble\DeclareMicrotypeBabelHook
```

File 209 **lwarp-midfloat.sty**

§ 306 Package midfloat

(*Emulates or patches code by* Sigitas Tolušis.)

Pkg midfloat midfloat is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{midfloat}[2012/05/29]

\newenvironment{strip}[1][]{}{}
\newskip\stripsep
```

File 210 **lwarp-midpage.sty**

§ 307 Package midpage

Pkg midpage midpage is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{midpage}[2009/09/03]

\newenvironment{midpage}
{\begin{BlockClass}[%
    \LWR@print@mbox{margin-top:6ex} ; \LWR@print@mbox{margin-bottom:6ex}%
]{midpage}}
{\end{BlockClass}}
```

File 211 **lwarp-morefloats.sty**

§ 308 Package morefloats

Pkg morefloats morefloats is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{morefloats}[2015/07/22]
```

File 212 **lwarp-moreverb.sty**

§ 309 Package moreverb

(*Emulates or patches code by* Robin Fairbairns.)

Pkg moreverb moreverb is supported with some patches.

for HTML output:

```
\begin{warpHTML}
\LWR@ProvidesPackagePass{moreverb}[2008/06/03]
\BeforeBeginEnvironment{verbatimtab}{%
\LWR@forcenewpage
\LWR@atbeginverbatim{3}{Verbatim}%
}
\AfterEndEnvironment{verbatimtab}{%
\LWR@afterendverbatim{1}%
}


\LetLtxMacro\LWRMV@orig@verbatimtabinput\@verbatimtabinput

\renewcommand{\@verbatimtabinput}[2][]{%
\LWR@forcenewpage
\LWR@atbeginverbatim{3}{Verbatim}%
\LWRMV@orig@verbatimtabinput[#1]{#2}%
\LWR@afterendverbatim{1}%
}

\BeforeBeginEnvironment{listing}{%
\LWR@forcenewpage
\LWR@atbeginverbatim{3}{programlisting}%
}

\AfterEndEnvironment{listing}{%
\LWR@afterendverbatim{1}%
}

\BeforeBeginEnvironment{listingcont}{%
```

```
\LWR@forcenewpage
\LWR@atbeginverbatim{3}{programlisting}%
}

\AfterEndEnvironment{listingcont}{%
\LWR@afterendverbatim{1}%
}

\LetLtxMacro\LWRMV@@listinginput\@listinginput

\renewcommand{\@listinginput}[3][]{
\LWR@forcenewpage
\LWR@atbeginverbatim{3}{programlisting}%
\LWRMV@@listinginput[#1]{#2}{#3}%
\LWR@afterendverbatim{1}%
}


\renewenvironment*{boxedverbatim}
{
\LWR@forcenewpage
\LWR@atbeginverbatim{3}{boxedverbatim}%
\verbatim%
}
{
\endverbatim%
\LWR@afterendverbatim{1}%
}

\end{warpHTML}
```

File 213 **lwarp-mparhack.sty**

§ 310 Package mparhack

Pkg mparhack Ignored.

for HTML output: Discard all options for lwarp-mparhack:

```
\LWR@ProvidesPackageDrop{mparhack}[2005/04/17]
```

File 214 **lwarp-multicap.sty**

§ 311 Package multicap

Pkg multicap multicap is emualted.

for HTML output:

```
\LWR@ProvidesPackageDrop{multicap}[2002/05/04]
```

```
\newcommand*{\mfcaption}{\captionof{figure}}
\newcommand*{\mtcaption}{\captionof{table}}
\newcounter{mcapsize}
\newcounter{mcapskip}
\newlength{\abvmcapskip}
\newlength{\blwmcapskip}
```

File 215 **lwarp-multicol.sty**

§ 312 Package

multicol

(*Emulates or patches code by* Frank Mittelbach.)

Pkg multicol multicol is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{multicol}[2018/12/27]
```

Multicols are converted into a 1–3 column display, browser-supported.

The optional multicols heading is placed inside a of class multicolsheading.

The content is placed inside a of class multicols.

```
\begin{warpHTML}
```

Env multicols * {⟨*numcols*⟩} [⟨*heading*⟩]

```
\NewDocumentEnvironment{multicols}{s m o}
```

HTML class to contain everything:

```
{
    \LWR@forcenewpage
    \BlockClass{multicols}
```

Optional HTML class for the heading:

```
    \IfValueT{#3}{\begin{BlockClass}{multicolsheading}#3\end{BlockClass}}%
```

Change \linewidth to compensate for expected size:

```
    \setlength{\linewidth}{\linewidth/#2}
```

Locally force any minipages to be fullwidth:

```
    \booltrue{LWR@forceminipagefullwidth}
}
```

When done with the environment, close the :

```
{\endBlockClass}
```

Emulated null functions which are not used in HTML:

```
\newcommand*{\columnbreak}{}
\newcommand*{\RLmulticolcolumns}{}
```

```
\newcommand*{\LRmulticolcolumns}{}

\newlength{\premulticols}
\newlength{\postmulticols}
\newlength{\multicolsep}
\newlength{\multicolbaselineskip}
\newlength{\multicoltolerance}
\newlength{\multicolpretolerance}
\newcommand*{\columnseprulecolor}{\normalcolor}
\newcounter{columnbadness}
\newcounter{finalcolumnbadness}
\newcounter{collectmore}
\newcounter{unbalance}
\newlength{\multicolovershoot}
\newlength{\multicolundershoot}

\NewDocumentCommand{\docolaction}{s o m m m}{%
    \IfValueTF{#2}{#2}{#3}%
}

\end{warpHTML}
```

File 216 **lwarp-multicolrule.sty**

§ 313 Package multicolrule

Pkg multicolrule multicolrule is ignored.

for HTML output:

```
\RequirePackage{multicol}

\LWR@ProvidesPackageDrop{multicolrule}[2019/01/01]

\newcommand*{\SetMCRule}[1]{}
\NewDocumentCommand{\DeclareMCRulePattern}{m m}{}
```

File 217 **lwarp-multirow.sty**

§ 314 Package multirow

(*Emulates or patches code by* Piet van Oostrum, Øystein Bache, Jerry Leichter.)

Pkg multirow multirow is emulated during HTML output, and used as-is while inside a `lateximage`.

In a `lateximage`, the original print-mode versions are temporarily restored by `\LWR@restoreorigformatting`.

See section 71.23 for the print-mode versions.

for HTML output: Remove the placeholder macro which was used if multirow was not loaded:

```
\LetLtxMacro\multirow\relax
```

```
\LWR@ProvidesPackagePass{multirow}[2018/08/03]
```

\LWR@multirowborder Set to `left` or `right` to create a thick border for the cell, for use by bigdelim:

```
\newcommand{\LWR@multirowborder}{}
```

§ 314.1 Multirow

\multirow [⟨*vpos*⟩] {⟨*numrows*⟩} [⟨*bigstruts*⟩] {⟨*width*⟩} [⟨*fixup*⟩] {⟨*text*⟩}

```
\NewDocumentCommand{\LWR@HTML@multirow}{O{c} m o m o +m}%
{%
\LWR@traceinfo{*** LWR@HTML@multirow #1 #2 #4}%
\LWR@maybenewtablerow%
\LWR@tabularleftedge%
```

Print the start of a new table data cell:

```
\LWR@htmltag{td rowspan="#2" %
```

The vertical alignment, if given:

```
\IfValueT{#1}{%
\ifstrequal{#1}{b}{style="\LWR@print@mbox{vertical-align:bottom}" }{}%
\ifstrequal{#1}{t}{style="\LWR@print@mbox{vertical-align:top}" }{}%
}%
```

The left/right border, if given:

```
\ifdefvoid{\LWR@multirowborder}{}{%
style="\LWR@print@mbox{border-\LWR@multirowborder:} 2px dotted black ; %
\LWR@print@mbox{padding-\LWR@multirowborder:} 2px" %
}%
```

A class adds the column spec and the rule:

```
class="td%
```

Append this column's spec:

```
\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.

```
\LWR@addcmidruletrim%
\LWR@addleftmostbartag%
\LWR@printbartag{\arabic{LWR@tableLaTeXcolindex}}%
"%


\LWR@tdstartstyles%
\LWR@addcmidrulewidth%
```

```
\LWR@addcdashline%
\LWR@addtabularrulecolors%
\LWR@tdendstyles%
}%
```

The column's < spec:

```
\LWR@getexparray{LWR@colbeforespec}{\arabic{LWR@tableLaTeXcolindex}}%
```

While printing the text, redefine \\ to generate a new line

```
\begingroup\LetLtxMacro{\\}{\LWR@endofline}#6\endgroup%
\LWR@stoppars%
\global\boolfalse{LWR@intabularmetadata}%
\renewcommand{\LWR@multirowborder}{}%
\LWR@traceinfo{*** LWR@HTML@multirow done}%
}%

\LWR@formatted{multirow}
```

§ 314.2 Combined multicolumn and multirow

⚠ \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

⚠ 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:

```
. . . &  \multicolumnrow{2}{c}[c]{3}[0]{1in}[0pt]{Text}  & . . .
. . . &  \mcolrowcell          &          \mcolrowcell   & . . .
. . . &  \mcolrowcell          &          \mcolrowcell   & . . .
```

vposn

Note that recent versions of multirow include a new optional vposn argument.

\multicolumnrow {⟨*1:cols*⟩} {⟨*2:halign*⟩} [⟨*3:vpos*⟩] {⟨*4:numrows*⟩} [⟨*5:bigstruts*⟩] {⟨*6:width*⟩} [⟨*7:fixup*⟩] {⟨*8:text*⟩}

\@ifpackageloaded{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.

```
\AtBeginDocument{
```

```
40
41 \NewExpandableDocumentCommand{\LWR@HTML@multicolumnrow}{m m O{} m O{} m O{} +m}{%
```

Figure out how many extra HTML columns to add for `@` and `!` columns:

```
42 \LWR@tabularhtmlcolumns{\arabic{LWR@tableLaTeXcolindex}}{#1}
```

Create the multicolumn/multirow tag:

```
43 \begingroup%
44 \LetLtxMacro{\\}{\LWR@endofline}%
45 \LWR@domulticolumn[#3][#4]{#1}{\arabic{LWR@tabhtmlcoltotal}}{#2}{#8}%
46 \endgroup%
```

Move to the next LaTeX column:

```
47 \addtocounter{LWR@tableLaTeXcolindex}{#1}%
48 \addtocounter{LWR@tableLaTeXcolindex}{-1}%
```

Skip any trailing `@` or `!` columns for this cell:

```
49 \global\booltrue{LWR@skipatbang}%
50 }
51
52 \LWR@expandableformatted{multicolumnrow}
53
54 }% \AtBeginDocument
```

File 218 **lwarp-multitoc.sty**

§ 315 Package multitoc

Pkg multitoc 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 219 **lwarp-musicography.sty**

§ 316 Package musicography

(*Emulates or patches code by* Andrew A. Cashner.)

Pkg musicography musicography is patched for use by lwarp.

Images are used for the meter symbols, since the HTML fonts tend not to be the correct size. 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. Copy/paste into a text editor works well.

for HTML output:

```
\LWR@ProvidesPackagePass{musicography}[2018/05/21]

\RenewDocumentCommand{\musSymbol}{ O{\musFont} m m m m }{%
\begin{lateximage}%
{#1\kern#2\raisebox{#3}{#5}\kern#4}%
\end{lateximage}%
}

\RenewDocumentCommand{\musStemmedNote}{ m }{%
\begin{lateximage}%
\musSymbol{0.05em}{0.5ex}{0.2em}{#1\musStem}%
\end{lateximage}%
}

\RenewDocumentCommand{\musFlaggedNote}{ m m }{%
\begin{lateximage}%
\musSymbol{0.05em}{0.5ex}{0pt}{#1\musStem}%
\musSymbol{0pt}{0pt}{0.9em}{#2}%
\end{lateximage}%
}

\RenewDocumentCommand{\musDottedNote}{ m }{%
\begin{lateximage}%
#1\musDot%
\end{lateximage}%
}

\RenewDocumentCommand{\musMeter}{ m m }{%
\begin{lateximage}*[#1/#2][#1#2]%
\musStack{#1}{#2}\kern0.05em%
\end{lateximage}%
}

\RenewDocumentCommand{\meterCplus}{ m }{%
\begin{lateximage}*[C#1]%
    \meterC{}\kern-0.7pt#1%
\end{lateximage}%
}

\RenewDocumentCommand{\meterC}{}{%
\begin{lateximage}*[C]%
\musSymbolMeter{\symbol{83}}%
\end{lateximage}%
}

\RenewDocumentCommand{\meterCutC}{}{%
\begin{lateximage}*[C|]%
\musSymbolMeter{\symbol{82}}%
```

```
\end{lateximage}%
}

\RenewDocumentCommand{\meterCThreeTwo}{}{%
\begin{lateximage}*[C3/2]%
\meterCplus{\musStack{3}{2}}%
\end{lateximage}%
}

\RenewDocumentCommand{\musFlat}       {}{\HTMLunicode{266D}}
\RenewDocumentCommand{\musDoubleFlat} {}{\HTMLunicode{1D12B}}
\RenewDocumentCommand{\musSharp}      {}{\HTMLunicode{266F}}
\RenewDocumentCommand{\musDoubleSharp}{}{\HTMLunicode{1D12A}}
\RenewDocumentCommand{\musNatural}    {}{\HTMLunicode{266E}}

\RenewDocumentCommand{\musWhole}{}{\HTMLunicode{1D15D}}
\RenewDocumentCommand{\musHalf}{}{\HTMLunicode{1D15E}}
\RenewDocumentCommand{\musQuarter}{}{\HTMLunicode{1D15F}}
\RenewDocumentCommand{\musEighth}{}{\HTMLunicode{1D160}}
\RenewDocumentCommand{\musSixteenth}{}{\HTMLunicode{1D161}}

\RenewDocumentCommand{\musWholeDotted}{}{\HTMLunicode{1D15D}\HTMLunicode{1D16D}}
\RenewDocumentCommand{\musHalfDotted}{}{\HTMLunicode{1D15E}\HTMLunicode{1D16D}}
\RenewDocumentCommand{\musQuarterDotted}{}{\HTMLunicode{1D15F}\HTMLunicode{1D16D}}
\RenewDocumentCommand{\musEighthDotted}{}{\HTMLunicode{1D160}\HTMLunicode{1D16D}}
\RenewDocumentCommand{\musSixteenthDotted}{}{\HTMLunicode{1D161}\HTMLunicode{1D16D}}
```

File 220 **lwarp-nameauth.sty**

§ 317 Package nameauth

(*Emulates or patches code by* Charles P. Schaum.)

Pkg nameauth nameauth is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{nameauth}[2017/03/22]
```

lwarp formatting is inserted in the following.

```
\renewcommand*\@nameauth@Hook[1]
{%
  \if@nameauth@Lock
    \@nameauth@InHooktrue%
    \protected@edef\test{#1}%
    \expandafter\@nameauth@TestDot\expandafter{\test}%
    \if@nameauth@InAKA
      \if@nameauth@AlwaysFormat
        \@nameauth@FirstFormattrue%
      \else
        \unless\if@nameauth@AKAFormat
        \@nameauth@FirstFormatfalse\fi
      \fi
      \if@nameauth@MainFormat
```

```
      \if@nameauth@FirstFormat
        \bgroup\NamesFormat{%
          \LWR@textcurrentcolor{\LWR@textcurrentfont{#1}}%    lwarp
        }\egroup%
      \else
        \bgroup\MainNameHook{%
          \LWR@textcurrentcolor{\LWR@textcurrentfont{#1}}%    lwarp
        }\egroup%
      \fi
    \else
      \if@nameauth@FirstFormat
        \bgroup\FrontNamesFormat{%
          \LWR@textcurrentcolor{\LWR@textcurrentfont{#1}}%    lwarp
        }\egroup%
      \else
        \bgroup\FrontNameHook{%
          \LWR@textcurrentcolor{\LWR@textcurrentfont{#1}}%    lwarp
        }\egroup%
      \fi
    \fi
  \else
    \if@nameauth@AlwaysFormat
      \@nameauth@FirstFormattrue%
    \fi
    \if@nameauth@MainFormat
      \if@nameauth@FirstFormat
        \bgroup\NamesFormat{%
          \LWR@textcurrentcolor{\LWR@textcurrentfont{#1}}%    lwarp
        }\egroup%
      \else
        \bgroup\MainNameHook{%
          \LWR@textcurrentcolor{\LWR@textcurrentfont{#1}}%    lwarp
        }\egroup%
      \fi
    \else
      \if@nameauth@FirstFormat
        \bgroup\FrontNamesFormat{%
          \LWR@textcurrentcolor{\LWR@textcurrentfont{#1}}%    lwarp
        }\egroup%
      \else
        \bgroup\FrontNameHook{%
          \LWR@textcurrentcolor{\LWR@textcurrentfont{#1}}%    lwarp
        }\egroup%
      \fi
    \fi
  \fi
  \@nameauth@FirstFormatfalse%
  \@nameauth@InHookfalse%
 \fi
}
```

File 221 **lwarp-nameref.sty**

§ 318 Package nameref

Pkg nameref nameref is emulated by lwarp.

for HTML output: Discard all options for lwarp-nameref:

```
\typeout{Using the lwarp html version of package ‘nameref’, discarding options.}
\typeout{    Are not using ProvidesPackage, so that other packages}
\typeout{    do not attempt to patch lwarp’s version of ‘nameref’.}
\DeclareOption*{}
\ProcessOptions\relax
```

File 222 **lwarp-natbib.sty**

§ 319 Package natbib

(*Emulates or patches code by* Patrick W. Daly.)

Pkg natbib natbib is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{natbib}[2010/09/13]
```

Replace math < and > with `\textless` and `\textgreater`:

A macro to compare:

```
\newcommand{\LWRNB@NAT@open}{$<$}
```

To patch `\NAT@open` and `\NAT@close`

```
\newcommand{\LWRNB@patchnatbibopenclose}{
\ifdefstrequal{\NAT@open}{\LWRNB@NAT@open}
{
    \renewcommand{\NAT@open}{\textless}
    \renewcommand{\NAT@close}{\textgreater}
}{}
}
```

Do it now in case `angle` was selected as an option:

```
\LWRNB@patchnatbibopenclose
```

Also patch `\setcitestyle` to patch after settings are made:

```
\let\LWRNB@origsetcitestyle\setcitestyle

```

```
\renewcommand{\setcitestyle}[1]{%
\LWRNB@origsetcitestyle{#1}%
\LWRNB@patchnatbibopenclose%
}
```

File 223 **lwarp-nccfancyhdr.sty**

§ 320 Package nccfancyhdr

(*Emulates or patches code by* Alexander I. Rozhenko.)

Pkg nccfancyhdr nccfancyhdr is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{nccfancyhdr}[2004/12/07]

\newcommand*{\headrulewidth}{}
\newcommand*{\footrulewidth}{}
\newcommand{\headstrutheight}{}
\newcommand{\footstrutheight}{}
\newcommand*{\headrule}{}
\newcommand*{\footrule}{}

\newdimen\headwidth
\newcommand*{\extendedheaders}{}
\newcommand*{\normalheaders}{}

\newcommand*{\fancyhead}[2][]{}
\newcommand*{\fancyfoot}[2][]{}
\newcommand*{\fancyhf}[2][]{}
\newcommand*{\fancypagestyle}[2]{}
\newcommand*{\lhead}[2][]{}
\newcommand*{\chead}[2][]{}
\newcommand*{\rhead}[2][]{}
\newcommand*{\lfoot}[2][]{}
\newcommand*{\cfoot}[2][]{}
\newcommand*{\rfoot}[2][]{}

\newcommand{\nouppercase}[1]{#1}

\NewDocumentCommand{\fancycenter}{o o m m m}{}

\NewDocumentCommand{\newpagestyle}{m o m}{}

\newcommand*{\iffloatpage}[2]{#2}
\newcommand*{\ifftopfloat}[2]{#2}
\newcommand*{\iffbotfloat}[2]{#2}
```

File 224 **lwarp-needspace.sty**

§ 321 Package needspace

(*Emulates or patches code by* Peter Wilson.)

Pkg needspace needspace is not used during HTML conversion.

for HTML output: Discard all options for lwarp-needspace:

```
\LWR@ProvidesPackageDrop{needspace}[2010/09/12]

\DeclareDocumentCommand{\needspace}{m}{}
\DeclareDocumentCommand{\Needspace}{s m}{}
```

File 225 **lwarp-nextpage.sty**

§ 322 Package nextpage

(*Emulates or patches code by* Peter Wilson.)

Pkg nextpage nextpage is nullified.

for HTML output: Discard all options for lwarp-nextpage.

```
\LWR@ProvidesPackageDrop{nextpage}[2009/09/03]

\DeclareDocumentCommand{\cleartoevenpage}{o}{}
\DeclareDocumentCommand{\movetoevenpage}{o}{}
\DeclareDocumentCommand{\cleartooddpage}{o}{}
\DeclareDocumentCommand{\movetooddpage}{o}{}
```

File 226 **lwarp-nicefrac.sty**

§ 323 Package nicefrac

(*Emulates or patches code by* Axel Reichert.)

Pkg nicefrac nicefrac is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{nicefrac}[1998/08/04]

\DeclareRobustCommand*{\LWR@HTML@@UnitsNiceFrac}[3][]{%
    {% localize font selection
        #1{%
            \LWR@textcurrentfont{%
```

```
                \InlineClass{numerator}{#2}%
                /%
                \InlineClass{denominator}{#3}%
            }%
        }%
    }%
}

\LWR@formatted{@UnitsNiceFrac}

\DeclareRobustCommand*{\LWR@HTML@@UnitsUglyFrac}[3][]{%
    {% localize font selection
        #1{\LWR@textcurrentfont{#2/#3}}%
    }%
}

\LWR@formatted{@UnitsUglyFrac}
```

For Mathjax:

```
\CustomizeMathJax{\newcommand{\nicefrac}[3][]{#2/#3}}
```

File 227 **lwarp-nomencl.sty**

§ 324 Package nomencl

(*Emulates or patches code by* Boris Veytsman, Bernd Schandl, Lee Netherton, CV Radhakrishnan.)

Pkg nomencl 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:

```
\LWR@ProvidesPackagePass{nomencl}[2005/09/22]

\def\@@@nomenclature[#1]#2#3{%
 \def\@tempa{#2}\def\@tempb{#3}%
\protected@write\@nomenclaturefile{}%
 {\string\nomenclatureentry{#1\nom@verb\@tempa @[{\nom@verb\@tempa}]%
     \begingroup\nom@verb\@tempb\protect\nomeqref{\theequation}%
       |nompageref}{\theLWR@previousautopagelabel}}%   lwarp
 \endgroup
 \@esphack}

\renewcommand*{\pagedeclaration}[1]{, \nameref{autopage-#1}}
```

File 228 **lwarp-nonfloat.sty**

§ 325 Package nonfloat

(*Emulates or patches code by* Kai Rascher.)

Pkg nonfloat nonfloat is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{nonfloat}[1999/07/05]

\LetLtxMacro\topcaption\caption
\newcommand{\figcaption}{\def\@captype{figure}\caption}
\newcommand{\tabcaption}{\def\@captype{table}\topcaption}
\newenvironment{narrow}[2]{}{}
```

File 229 **lwarp-nonumonpart.sty**

§ 326 Package nonumonpart

Pkg nonumonpart nonumonpart is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{nonumonpart}[2011/04/15]
```

File 230 **lwarp-nopageno.sty**

§ 327 Package nopageno

Pkg nopageno nopageno is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{nopageno}[1989/01/01]
```

File 231 **lwarp-notes.sty**

§ 328 Package notes

Pkg notes notes is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{notes}[2002/10/29]

\newcommand*{\LWR@notes@onenote}[2]{%
\newenvironment{#1}
    {
```

```
        \BlockClass{notes#1}
        \begin{BlockClass}{notesicon}\textcircled{~#2~}\end{BlockClass}
        \BlockClass{notescontents}
    }
    {\endBlockClass\endBlockClass}
}

\LWR@notes@onenote{importantnote}{!}

\LWR@notes@onenote{warningnote}{--}

\LWR@notes@onenote{informationnote}{i}
```

File 232 **lwarp-notespages.sty**

§ 329 Package notespages

Pkg notespages notespages is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{notespages}[2016/08/21]

\newcommand*{\npnotesname}{}
\newcommand*{\npnotestext}{}
\newcommand*{\remainingtextheight}{}
\newdimen\remainingtextheight
\newcommand*{\notestitletext}{}
\newcommand*{\notesareatext}{}
\newcommand*{\npnpinfo}[1]{}
\newcommand*{\tracingnpmarks}{}
\newcommand*{\notespage}[1][]{}
\newcommand*{\notespages}[1][]{}
\newcommand*{\notesfill}[1][]{}
\newcommand*{\setnotespages}[1]{}
\newcommand*{\definenotesoption}[2]{}
\newcommand{\definenotesstyle}[2]{}
\newcommand{\definetitlestyle}[2]{}
\newcommand{\nppatchchapter}[1]{}
\newcommand{\npunpatchchapter}{}
```

File 233 **lwarp-nowidow.sty**

§ 330 Package nowidow

(*Emulates or patches code by* Raphaël Pinson.)

Pkg nowidow nowidow is not used during HTML conversion.

Discard all options for lwarp-nowidow:

for HTML output:

```
\LWR@ProvidesPackageDrop{nowidow}[2011/09/20]
```

\nowidow [⟨*lines*⟩]
\setnowidow [⟨*lines*⟩]

```
\newcommand*{\nowidow}[1][]{}
\newcommand*{\setnowidow}[1][]{}
```

\noclub [⟨*lines*⟩]
\setnoclub [⟨*lines*⟩]

```
\newcommand*{\noclub}[1][]{}
\newcommand*{\setnoclub}[1][]{}
```

File 234 **lwarp-ntheorem.sty**

§ 331 Package ntheorem

(*Emulates or patches code by* Wolfgang May, Andreas Schedler.)

Pkg ntheorem ntheorem is patched for use by lwarp.

Table 15: Ntheorem package — css styling of theorems and proofs

Theorem: of class theorembody<theoremstyle>

Theorem Header: of class theoremheader<style>

where <theoremstyle> is plain, break, etc.

§ 331.1 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 $\mathcal{AMS}$ environmentswhen 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.

§ 331.2 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:

```
\DeclareOption{thref}{}


\newbool{LWR@ntheoremmarks}
\boolfalse{LWR@ntheoremmarks}

\DeclareOption{thmmarks}{
\booltrue{LWR@ntheoremmarks}
\newif\ifsetendmark\setendmarktrue
}


\newbool{LWR@ntheoremamsthm}
\boolfalse{LWR@ntheoremamsthm}

\DeclareOption{amsthm}{\booltrue{LWR@ntheoremamsthm}}


\DeclareOption{amsmath}{}
\DeclareOption{hyperref}{}

\LWR@ProvidesPackagePass{ntheorem}[2011/08/15]
```

§ 331.3 Remembering the theorem style

Storage for the style being used for new theorems.

```
\newcommand{\LWR@newtheoremstyle}{plain}


\AtBeginDocument{
\@ifpackageloaded{cleveref}{
\gdef\@thm#1#2#3{%
   \if@thmmarks
     \stepcounter{end\InTheoType ctr}%
   \fi
   \renewcommand{\InTheoType}{#1}%
   \if@thmmarks
     \stepcounter{curr#1ctr}%
     \setcounter{end#1ctr}{0}%
   \fi
   \refstepcounter[#1]{#2}%  <<< cleveref modification
   \theorem@prework
    \LWR@forcenewpage% lwarp
    \BlockClass{theorembody#1}%\LWR@thisthmstyle% lwarp
   \trivlist % latex's \trivlist, calling latex's \@trivlist unchanged
   \ifuse@newframeskips  % cf. latex.ltx for topsepadd: \@trivlist
     \ifthm@inframe
       \thm@topsep\theoreminframepreskipamount
       \thm@topsepadd\theoreminframepostskipamount
      \else
       \thm@topsep\theorempreskipamount
       \thm@topsepadd\theorempostskipamount
      \fi
```

```
    \else% oldframeskips
      \thm@topsep\theorempreskipamount
      \thm@topsepadd \theorempostskipamount
      \ifvmode\advance\thm@topsepadd\partopsep\fi
  \fi
  \@topsep\thm@topsep
  \@topsepadd\thm@topsepadd
  \advance\linewidth -\theorem@indent
  \advance\linewidth -\theorem@rightindent
  \advance\@totalleftmargin \theorem@indent
  \parshape \@ne \@totalleftmargin \linewidth
  \@ifnextchar[{\@ythm{#1}{#2}{#3}}{\@xthm{#1}{#2}{#3}}
}
}{% not @ifpackageloaded{cleveref}
\gdef\@thm#1#2#3{%
  \if@thmmarks
    \stepcounter{end\InTheoType ctr}%
  \fi
  \renewcommand{\InTheoType}{#1}%
  \if@thmmarks
    \stepcounter{curr#1ctr}%
    \setcounter{end#1ctr}{0}%
  \fi
  \refstepcounter{#2}%
  \theorem@prework
   \LWR@forcenewpage% lwarp
   \BlockClass{theorembody#1}%\LWR@thisthmstyle% lwarp
  \trivlist % latex's \trivlist, calling latex's \@trivlist unchanged
  \ifuse@newframeskips  % cf. latex.ltx for topsepadd: \@trivlist
    \ifthm@inframe
      \thm@topsep\theoreminframepreskipamount
      \thm@topsepadd\theoreminframepostskipamount
     \else
      \thm@topsep\theorempreskipamount
      \thm@topsepadd\theorempostskipamount
     \fi
   \else% oldframeskips
     \thm@topsep\theorempreskipamount
     \thm@topsepadd \theorempostskipamount
     \ifvmode\advance\thm@topsepadd\partopsep\fi
  \fi
  \@topsep\thm@topsep
  \@topsepadd\thm@topsepadd
  \advance\linewidth -\theorem@indent
  \advance\linewidth -\theorem@rightindent
  \advance\@totalleftmargin \theorem@indent
  \parshape \@ne \@totalleftmargin \linewidth
  \@ifnextchar[{\@ythm{#1}{#2}{#3}}{\@xthm{#1}{#2}{#3}}
}
}
}% AtBeginDocument
```

Patched to remember the style being used for new theorems:

```
\gdef\theoremstyle#1{%
```

```
   \@ifundefined{th@#1}{\@warning
          {Unknown theoremstyle '#1'. Using 'plain'}%
          \theorem@style{plain}
            \renewcommand{\LWR@newtheoremstyle}{plain}% lwarp
            }%
      {
        \theorem@style{#1}
        \renewcommand{\LWR@newtheoremstyle}{#1}% lwarp
        }
}
```

Patched to remember the style for this theorem type, and set it later when the environment is started.

```

\gdef\@xnthm#1#2[#3]{%
  \ifthm@tempif
     \csedef{LWR@thmstyle#1}{\LWR@newtheoremstyle}% lwarp
     \expandafter\@ifundefined{c@#1}%
        {\@definecounter{#1}}{}%
     \@newctr{#1}[#3]%
     \expandafter\xdef\csname the#1\endcsname{%
       \expandafter\noexpand\csname the#3\endcsname \@thmcountersep
          {\noexpand\csname\the\theoremnumbering\endcsname{#1}}}%
     \expandafter\gdef\csname mkheader@#1\endcsname
       {\csname setparms@#1\endcsname
        \@thm{#1}{#1}{#2}
        }%
     \global\@namedef{end#1}{\@endtheorem}
    \AtBeginEnvironment{#1}{\edef\LWR@thisthmstyle{\@nameuse{LWR@thmstyle#1}}}% lwarp
  \fi
}

\gdef\@ynthm#1#2{%
  \ifthm@tempif
     \csedef{LWR@thmstyle#1}{\LWR@newtheoremstyle}% lwarp
     \expandafter\@ifundefined{c@#1}%
        {\@definecounter{#1}}{}%
     \expandafter\xdef\csname the#1\endcsname
        {\noexpand\csname\the\theoremnumbering\endcsname{#1}}%
     \expandafter\gdef\csname mkheader@#1\endcsname
       {\csname setparms@#1\endcsname
        \@thm{#1}{#1}{#2}
        }%
     \global\@namedef{end#1}{\@endtheorem}
    \AtBeginEnvironment{#1}{\edef\LWR@thisthmstyle{\@nameuse{LWR@thmstyle#1}}}% lwarp
  \fi
}

\gdef\@othm#1[#2]#3{%
  \@ifundefined{c@#2}{\@nocounterr{#2}}%
   {\ifthm@tempif
     \csedef{LWR@thmstyle#1}{\LWR@newtheoremstyle}% lwarp
     \global\@namedef{the#1}{\@nameuse{the#2}}%
     \expandafter\protected@xdef\csname num@addtheoremline#1\endcsname{%
```

```
              \noexpand\@num@addtheoremline{#1}{#3}}%
     \expandafter\protected@xdef\csname nonum@addtheoremline#1\endcsname{%
              \noexpand\@nonum@addtheoremline{#1}{#3}}%
    \theoremkeyword{#3}%
    \expandafter\protected@xdef\csname #1Keyword\endcsname
             {\the\theoremkeyword}%
     \expandafter\gdef\csname mkheader@#1\endcsname
       {\csname setparms@#1\endcsname
                \@thm{#1}{#2}{#3}
        }%
     \global\@namedef{end#1}{\@endtheorem}
   \AtBeginEnvironment{#1}{\edef\LWR@thisthmstyle{\@nameuse{LWR@thmstyle#1}}}% lwarp
  \fi}
}
```

§ 331.4 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.

```
\newcommand{\LWR@inctheorem}{%
\addtocounter{LWR@thisautoid}{1}%
\LWR@stoppars%
\LWR@htmltag{a id="\LWR@print@mbox{autoid-\arabic{LWR@thisautoid}}"}\LWR@htmltag{/a}%
\LWR@startpars%
}
```

§ 331.5 \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>`.

```
\gdef\newtheoremstyle#1#2#3{%
  \expandafter\@ifundefined{th@#1}%
   {\expandafter\gdef\csname th@#1\endcsname{%
    \def\@begintheorem####1####2{%
    \LWR@inctheorem% lwarp
    #2}%
    \def\@opargbegintheorem####1####2####3{%
    \LWR@inctheorem% lwarp
    #3}%
}%
}%
{\PackageError{\basename}{Theorem style #1 already defined}\@eha}
}
```

§ 331.6 Standard styles

```
\renewtheoremstyle{plain}%
  {\item[
    \InlineClass{theoremheaderplain}{##1\ ##2\theorem@separator}]}%
  {\item[
    \InlineClass{theoremheaderplain}{##1\ ##2\ (##3)\theorem@separator}]}

\renewtheoremstyle{break}%
  {\item[
    \InlineClass{theoremheaderbreak}{##1\ ##2\theorem@separator}\newline
    ]}%
  {\item[
    \InlineClass{theoremheaderbreak}%
        {##1\ ##2\ (##3)\theorem@separator}\newline
    ]}

\renewtheoremstyle{change}%
  {\item[
    \InlineClass{theoremheaderchange}{##2\ ##1\theorem@separator}]}%
  {\item[
    \InlineClass{theoremheaderchange}{##2\ ##1\ (##3)\theorem@separator}]}

\renewtheoremstyle{changebreak}%
  {\item[
        \InlineClass{theoremheaderchangebreak}%
            {##2\ ##1\theorem@separator}\newline
    ]}%
  {\item[
        \InlineClass{theoremheaderchangebreak}%
            {##2\ ##1\ (##3)\theorem@separator}\newline
    ]}

\renewtheoremstyle{margin}%
  {\item[
        \InlineClass{theoremheadermargin}{##2 \qquad ##1\theorem@separator}
    ]}%
  {\item[
     \InlineClass{theoremheadermargin}{##2 \qquad ##1\ (##3)\theorem@separator}
    ]}

\renewtheoremstyle{marginbreak}%
  {\item[
    \InlineClass{theoremheadermarginbreak}%
        {##2 \qquad ##1\theorem@separator}\newline
    ]}%
  {\item[
    \InlineClass{theoremheadermarginbreak}%
        {##2 \qquad ##1\ (##3)\theorem@separator}\newline
    ]}

\renewtheoremstyle{nonumberplain}%
  {\item[
    \InlineClass{theoremheaderplain}{##1\theorem@separator}]}%
  {\item[
    \InlineClass{theoremheaderplain}{##1\ (##3)\theorem@separator}]}
```

```
238
239 \renewtheoremstyle{nonumberbreak}%
240   {\item[
241     \InlineClass{theoremheaderbreak}{##1\theorem@separator}\newline
242     ]}%
243   {\item[
244     \InlineClass{theoremheaderbreak}{##1\ (##3)\theorem@separator}\newline
245     ]}
246
247 \renewtheoremstyle{empty}%
248   {\item[]}%
249   {\item[
250     \InlineClass{theoremheaderplain}{##3}]}
251
252 \renewtheoremstyle{emptybreak}%
253   {\item[]}%
254   {\item[
255     \InlineClass{theoremheaderplain}{##3}] \ \newline}
```

§ 331.7 Additional objects

The following manually adjust the CSS for the standard configuration objects which are not a purely plain style:

```
256 \ifbool{LWR@ntheoremamsthm}{}{%
```

Upright text via CSS:

```
257     \newtheoremstyle{plainupright}%
258     {\item[
259         \InlineClass{theoremheaderplain}{##1\ ##2\theorem@separator}]}%
260     {\item[
261         \InlineClass{theoremheaderplain}{##1\ ##2\ (##3)\theorem@separator}]}
```

Upright text and small caps header via CSS:

```
262     \newtheoremstyle{nonumberplainuprightsc}%
263     {\item[
264         \InlineClass{theoremheadersc}{##1\theorem@separator}]}%
265     {\item[
266         \InlineClass{theoremheadersc}{##1\ (##3)\theorem@separator}]}
267 }% not amsthm
```

§ 331.8 Renewed standard configuration

The following standard configuration is renewed using the new CSS:

```
268 \ifbool{LWR@ntheoremamsthm}{}{%
```

```
269 \ifx\thm@usestd\@undefined
270 \else
271     \theoremnumbering{arabic}
272     \theoremstyle{plain}
273     \RequirePackage{latexsym}
```

```
    \theoremsymbol{\Box}
    \theorembodyfont{\itshape}
    \theoremheaderfont{\normalfont\bfseries}
    \theoremseparator{}
    \renewtheorem{Theorem}{Theorem}
    \renewtheorem{theorem}{Theorem}
    \renewtheorem{Satz}{Satz}
    \renewtheorem{satz}{Satz}
    \renewtheorem{Proposition}{Proposition}
    \renewtheorem{proposition}{Proposition}
    \renewtheorem{Lemma}{Lemma}
    \renewtheorem{lemma}{Lemma}
    \renewtheorem{Korollar}{Korollar}
    \renewtheorem{korollar}{Korollar}
    \renewtheorem{Corollary}{Corollary}
    \renewtheorem{corollary}{Corollary}

    \theoremstyle{plainupright}
    \theorembodyfont{\upshape}
    \theoremsymbol{\HTMLunicode{25A1}}% UTF-8 white box
    \renewtheorem{Example}{Example}
    \renewtheorem{example}{Example}
    \renewtheorem{Beispiel}{Beispiel}
    \renewtheorem{beispiel}{Beispiel}
    \renewtheorem{Bemerkung}{Bemerkung}
    \renewtheorem{bemerkung}{Bemerkung}
    \renewtheorem{Anmerkung}{Anmerkung}
    \renewtheorem{anmerkung}{Anmerkung}
    \renewtheorem{Remark}{Remark}
    \renewtheorem{remark}{Remark}
    \renewtheorem{Definition}{Definition}
    \renewtheorem{definition}{Definition}

    \theoremstyle{nonumberplainuprightsc}
    \theoremsymbol{\HTMLunicode{220E}}% UTF-8 end-of-proof
    \renewtheorem{Proof}{Proof}
    \renewtheorem{proof}{Proof}
    \renewtheorem{Beweis}{Beweis}
    \renewtheorem{beweis}{Beweis}
    \qedsymbol{\HTMLunicode{220E}}% UTF-8 end-of-proof

    \theoremsymbol{}
\fi
}% not amsthm
```

§ 331.9 amsthm option

Only if the amsthm option was given:

```
\ifbool{LWR@ntheoremamsthm}{

\gdef\th@plain{%
  \def\theorem@headerfont{\normalfont\bfseries}\itshape%
  \def\@begintheorem##1##2{%
        \LWR@inctheorem% lwarp
```

```
      \item[
  \InlineClass{theoremheaderplain}{##1\ ##2.}
        ]}%
  \def\@opargbegintheorem##1##2##3{%
        \LWR@inctheorem% lwarp
     \item[
  \InlineClass{theoremheaderplain}{##1\ ##2\ (##3).}
        ]}}

\gdef\th@nonumberplain{%
  \def\theorem@headerfont{\normalfont\bfseries}\itshape%
  \def\@begintheorem##1##2{%
        \LWR@inctheorem% lwarp
      \item[
  \InlineClass{theoremheaderplain}{##1.}
        ]}%
  \def\@opargbegintheorem##1##2##3{%
        \LWR@inctheorem% lwarp
     \item[
  \InlineClass{theoremheaderplain}{##1\ (##3).}
        ]}}

\gdef\th@definition{%
  \def\theorem@headerfont{\normalfont\bfseries}\normalfont%
  \def\@begintheorem##1##2{%
        \LWR@inctheorem% lwarp
      \item[
  \InlineClass{theoremheaderdefinition}{##1\ ##2.}
        ]}%
  \def\@opargbegintheorem##1##2##3{%
        \LWR@inctheorem% lwarp
     \item[
  \InlineClass{theoremheaderdefinition}{##1\ ##2\ (##3).}
        ]}}

\gdef\th@nonumberdefinition{%
  \def\theorem@headerfont{\normalfont\bfseries}\normalfont%
  \def\@begintheorem##1##2{%
        \LWR@inctheorem% lwarp
      \item[
  \InlineClass{theoremheaderdefinition}{##1.}
        ]}%
  \def\@opargbegintheorem##1##2##3{%
        \LWR@inctheorem% lwarp
     \item[
  \InlineClass{theoremheaderdefinition}{##1\ (##3).}
        ]}}

\gdef\th@remark{%
  \def\theorem@headerfont{\itshape}\normalfont%
  \def\@begintheorem##1##2{%
        \LWR@inctheorem% lwarp
      \item[
  \InlineClass{theoremheaderremark}{##1\ ##2.}
        ]}%
```

```
  \def\@opargbegintheorem##1##2##3{%
        \LWR@inctheorem% lwarp
     \item[
  \InlineClass{theoremheaderremark}{##1\ ##2\ (##3).}
        ]}}

\gdef\th@nonumberremark{%
  \def\theorem@headerfont{\itshape}\normalfont%
  \def\@begintheorem##1##2{%
        \LWR@inctheorem% lwarp
      \item[
  \InlineClass{theoremheaderremark}{##1.}
        ]}%
  \def\@opargbegintheorem##1##2##3{%
        \LWR@inctheorem% lwarp
     \item[
  \InlineClass{theoremheaderremark}{##1\ (##3).}
        ]}}

\gdef\th@proof{%
  \def\theorem@headerfont{\normalfont\bfseries}\itshape%
  \def\@begintheorem##1##2{%
        \LWR@inctheorem% lwarp
      \item[
  \InlineClass{theoremheaderproof}{##1.}
        ]}%
  \def\@opargbegintheorem##1##2##3{%
        \LWR@inctheorem% lwarp
     \item[
  \InlineClass{theoremheaderproof}{##1\ (##3).}
        ]}}



\newcounter{proof}%
\if@thmmarks
    \newcounter{currproofctr}%
    \newcounter{endproofctr}%
\fi

\gdef\proofSymbol{\openbox}

\newcommand{\proofname}{Proof}

\newenvironment{proof}[1][\proofname]{
    \th@proof
    \def\theorem@headerfont{\itshape}%
    \normalfont
    \theoremsymbol{\HTMLunicode{220E}}% UTF-8 end-of-proof
    \@thm{proof}{proof}{#1}
}%
{\@endtheorem}

}{}% amsthm option
```

§ 331.10 Ending a theorem

Patched for css:

```
\let\LWR@origendtheorem\@endtheorem
\renewcommand{\@endtheorem}{%
\ifbool{LWR@ntheoremmarks}{%
    \ifsetendmark%
    \InlineClass{theoremendmark}{\csname\InTheoType Symbol\endcsname}%
    \setendmarkfalse%
    \fi%
}{}%
\LWR@origendtheorem% also does \@endtrivlist
\ifbool{LWR@ntheoremmarks}{\global\setendmarktrue}{}%
\endBlockClass%
}
```

§ 331.11 `\NoEndMark`

```
\gdef\NoEndMark{\global\setendmarkfalse}
```

§ 331.12 List-of

Redefined to reuse the float mechanism to add list-of-theorem links:

`\thm@thmline` {⟨*1: printed type*⟩} {⟨*2: #*⟩} {⟨*3: optional*⟩} {⟨*4: page*⟩}

```
\renewcommand{\thm@@thmline@noname}[4]{%
\hypertocfloat{1}{theorem}{thm}{#2 #3}{}%
}

\renewcommand{\thm@@thmline@name}[4]{%
\hypertocfloat{1}{theorem}{thm}{#1 #2 #3}{}%
}
```

This was redefined by ntheorem when loaded, so it is now redefined for lwarp:

```
\def\thm@@thmline{\thm@@thmline@name}
```

Patch for css:

```
\def\listtheorems#1{
\LWR@htmlelementclass{nav}{lothm}%
\begingroup
\c@tocdepth=-2%
\def\thm@list{#1}\thm@processlist
\endgroup
\LWR@htmlelementclassend{nav}{lothm}%
}
```

§ 331.13 Symbols

Proof QED symbol:

```
\newcommand{\qed}{\qquad\the\qedsymbol}

\AtBeginDocument{
\@ifundefined{LWR@orig@openbox}{
\LetLtxMacro\LWR@orig@openbox\openbox
\LetLtxMacro\LWR@orig@blacksquare\blacksquare
\LetLtxMacro\LWR@orig@Box\Box

\def\openbox{\text{\HTMLunicode{25A1}}}% UTF-8 white box
\def\blacksquare{\text{\HTMLunicode{220E}}}% UTF-8 end-of-proof
\def\Box{\text{\HTMLunicode{25A1}}}% UTF-8 white box

\appto\LWR@restoreorigformatting{%
\LetLtxMacro\openbox\LWR@orig@openbox%
\LetLtxMacro\blacksquare\LWR@orig@blacksquare%
\LetLtxMacro\Box\LWR@orig@Box%
}% appto
}{}% @ifundefined
}% AtBeginDocument
```

§ 331.14 Cross-referencing

`\thref {`⟨*label*⟩`}`

```
\newcommand*{\thref}[1]{\cref{#1}}%
```

File 235 **lwarp-octave.sty**

§ 332 Package octave

(*Emulates or patches code by* Andrew A. Cashner.)

Pkg octave octave is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{octave}[2017/10/31]
```

Remove the leading 1pt kern:

```
\RenewDocumentCommand{\@PrintTicks}{ m }{%
\kern-1pt%  lwarp
\@TickNum = #1%
\loop
\@Tick{}%
\advance\@TickNum by -1
\ifnum\@TickNum > 0
\repeat
}
```

Use unicode for the prime character:

```
\RenewDocumentCommand{\@Tick}{}{\HTMLunicode{2032}}
```

Catch the inline font:

```
\RenewDocumentCommand{\pitch}{ m o m }{%
\if@OctaveNumber%
{%
    \pitchfont{%
        \LWR@textcurrentfont{%  lwarp
            \MakeUppercase{#1}%
            \IfValueTF{#2}{#2}{}\textsubscript{#3}%
        }%
    }%
}%
\else%
{%
    \pitchfont{%
        \LWR@textcurrentfont{%  lwarp
            \@GetOctaveTick{#1}[#2]{#3}%
        }%
    }%
}%
\fi%
}
```

The original was hard to adapt to lwarp's handling of &.

```
\StartDefiningTabulars
\renewcommand{\octavetable}{%
\begin{tabular}{ll}
\octaveprimes \pitch{C}{0} & \octavenumbers \pitch{C}{0} \\
\octaveprimes \pitch{C}{1} & \octavenumbers \pitch{C}{1} \\
\octaveprimes \pitch{C}{2} & \octavenumbers \pitch{C}{2} \\
\octaveprimes \pitch{C}{3} & \octavenumbers \pitch{C}{3} \\
\octaveprimes \pitch{C}{4} & \octavenumbers \pitch{C}{4} \\
\octaveprimes \pitch{C}{5} & \octavenumbers \pitch{C}{5} \\
\octaveprimes \pitch{C}{6} & \octavenumbers \pitch{C}{6} \\
\octaveprimes \pitch{C}{7} & \octavenumbers \pitch{C}{7} \\
\end{tabular}
}
\StopDefiningTabulars
```

File 236 **lwarp-overpic.sty**

§ 333 Package overpic

(*Emulates or patches code by* Rolf Niepraschk.)

Pkg overpic overpic is patched for use by lwarp.

⚠ scaling The macros \overpicfontsize and \overpicfontskip are used during HTML genera-

tion. 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 83.2 for the print-mode version of \overpicfontsize and \overpicfontskip.

for HTML output:

```
\LWR@ProvidesPackagePass{overpic}[2017/10/06]

\newcommand*{\overpicfontsize}{12}
\newcommand*{\overpicfontskip}{14}

\BeforeBeginEnvironment{overpic}{%
    \begin{lateximage}%
    \fontsize{\overpicfontsize}{\overpicfontskip}%
    \selectfont%
}

\AfterEndEnvironment{overpic}{\end{lateximage}}

\BeforeBeginEnvironment{Overpic}{%
    \begin{lateximage}%
    \fontsize{\overpicfontsize}{\overpicfontskip}%
    \selectfont%
}

\AfterEndEnvironment{Overpic}{\end{lateximage}}
```

File 237 **lwarp-pagegrid.sty**

§ 334 Package pagegrid

Pkg pagegrid pagegrid is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pagegrid}[2016/05/16]

\newcommand*{\pagegridsetup}[1]{}
```

File 238 **lwarp-pagenote.sty**

§ 335 Package pagenote

Pkg pagenote pagenote works as-is, but the page option is disabled.

for HTML output:

```
\DeclareOption{page}{}
\LWR@ProvidesPackagePass{pagenote}[2009/09/03]
```

File 239 **lwarp-pagesel.sty**

§ 336 Package

pagesel

Pkg pagesel pagesel is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pagesel}[2016/05/16]
```

File 240 **lwarp-paralist.sty**

§ 337 Package

paralist

(*Emulates or patches code by* Bernd Schandl.)

Pkg paralist paralist is supported with minor changes.

for HTML output:

```
\LWR@ProvidesPackagePass{paralist}[2017/01/22]
```

The compact environments are identical to the regular ones:

```
\LetLtxMacro\compactitem\itemize
\LetLtxMacro\compactenum\enumerate
\LetLtxMacro\compactdesc\description
\LetLtxMacro\endcompactitem\enditemize
\LetLtxMacro\endcompactenum\endenumerate
\LetLtxMacro\endcompactdesc\enddescription
```

For the inline environments, revert `\item` to its original print-mode version:

```
\AtBeginEnvironment{inparaitem}{\LetLtxMacro\item\LWR@origitem}
\AtBeginEnvironment{inparaenum}{\LetLtxMacro\item\LWR@origitem}
\AtBeginEnvironment{inparadesc}{\LetLtxMacro\item\LWR@origitem}
```

Manual formatting of the description labels:

```
\def\paradescriptionlabel#1{{\normalfont\textbf{#1}}}
```

File 241 **lwarp-parnotes.sty**

§ 338 Package

parnotes

(*Emulates or patches code by* Chelsea Hughes.)

Pkg parnotes parnotes is supported with some patches.

for HTML output:

```
\LWR@ProvidesPackagePass{parnotes}[2016/08/15]
```

```
\long\def\PN@parnote@real#1#2{%
    \parnotemark{#1}%
    % Unless this is the first parnote in \PN@text, add a separator first
    \unless\ifx\PN@text\@empty\g@addto@macro\PN@text{\parnoteintercmd}\fi
    % Redefine \@currentlabel to the parnote label, so \label works
    \g@addto@macro\PN@text{%
%         \phantomsection%
        \def\@currentlabel{#1}%
        \def\cref@currentlabel{%        lwarp
            [parnotemark][\arabic{parnotemark}][]\theparnotemark%
        }%
    }%
    \g@addto@macro\PN@text{%
        \LWR@textcurrentfont{%          lwarp
            \parnotemark{#1}\nolinebreak\thinspace#2%
        }%
    }%
}

\def\PN@parnotes@real{%
    % We call \par later, so this avoids recursion with \PN@parnotes@auto
    \PN@inparnotestrue
    \unless\ifvmode\par\fi
    % Avoid page breaks between a paragraph and its parnotes
    \nopagebreak\addvspace{\parnotevskip}%
    \LWR@forcenewpage%                  lwarp
    \begin{BlockClass}{footnotes}%      lwarp
    {\parnotefmt{\PN@text}\par}%
    \end{BlockClass}%                   lwarp
    \global\def\PN@text{}%
    \addvspace{\parnotevskip}%
    %
    % These can be enabled or disabled by package options
    %
    \PN@disable@indent
    \PN@reset@optional
    \PN@inparnotesfalse
}

\AtBeginDocument{
\crefname{parnotemark}{paragraph note}{paragraph notes}%
\Crefname{parnotemark}{Paragraph note}{Paragraph notes}
}
```

File 242 **lwarp-parskip.sty**

§ 339 Package

parskip

Pkg parskip parskip is ignored.

for HTML output: Discard all options for lwarp-parskip.

```
\LWR@ProvidesPackageDrop{parskip}[2001/04/09]
```

File 243 **lwarp-pbox.sty**

§ 340 Package pbox

(*Emulates or patches code by* Simon Law.)

Pkg pbox pbox is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{pbox}[2011/12/07]

\NewDocumentCommand{\pbox}{O{t} O{} O{t} m +m}{%
\global\booltrue{LWR@minipagefullwidth}%
\parbox[#1][#2][#3]{#4}{#5}%
}

\newcommand{\settominwidth}[3][\columnwidth]{%
\settowidth{#2}{#3}%
}

\newcommand{\widthofpbox}[1]{%
\widthof{#1}%
}
```

File 244 **lwarp-pdfcomment.sty**

§ 341 Package pdfcomment

Pkg pdfcomment pdfcomment is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pdfcomment}[2016/06/13]

\newenvironment{pdfsidelinecomment}[2][]{}{}
\newcommand{\pdfcomment}[2][]{}
\newcommand{\pdfmargincomment}[2][]{}
\newcommand{\pdfmarkupcomment}[3][]{#2}
\newcommand{\pdffreetextcomment}[2][]{}
\newcommand{\pdfsquarecomment}[2][]{}
\newcommand{\pdfcirclecomment}[2][]{}
\newcommand{\pdflinecomment}[2][]{}
\newcommand{\pdftooltip}[3][]{#2}
\newcommand{\pdfcommentsetup}[2][]{}
\newcommand{\listofpdfcomments}[1][]{}
\newcommand{\setliststyle}[1]{}
\newcommand{\defineliststyle}[2]{}
\newcommand{\defineavatar}[2]{}
\newcommand{\definestyle}[2]{}
```

File 245 **lwarp-pdflscape.sty**

§ 342 Package

pdflscape

Pkg pdflscape pdflscape is ignored.

for HTML output: Discard all options for lwarp-pdflscape:

```
\LWR@ProvidesPackageDrop{pdflscape}[2016/05/14]
```

File 246 **lwarp-pdfmarginpar.sty**

§ 343 Package

pdfmarginpar

Pkg pdfmarginpar pdfmarginpar is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pdfmarginpar}[2011/08/05]

\newcommand{\pdfmarginpar}[2][]{}
\newcommand{\pdfmarginparset}[1]{}
```

File 247 **lwarp-pdfpages.sty**

§ 344 Package

pdfpages

(*Emulates or patches code by* Andreas Matthias.)

Pkg pdfpages 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:

```
\LWR@ProvidesPackagePass{pdfpages}[2017-10-31]
```

Disable option which have no meaning for HTML output:

```
\define@key{pdfpages}{fitpaper}[false]{}
\define@key{pdfpages}{landscape}[false]{}
```

```
\define@key{pdfpages}{openright}[false]{}
\define@key{pdfpages}{signature}{}
\define@key{pdfpages}{signature*}{}
\define@key{pdfpages}{booklet}[false]{}
\define@key{pdfpages}{rotateoversize}[false]{}
\define@key{pdfpages}{doublepages}[false]{}
\define@key{pdfpages}{doublepagestwist}[false]{}
\define@key{pdfpages}{doublepagestwistodd}[false]{}
\define@key{pdfpages}{doublepagestwist*}[false]{}
\define@key{pdfpages}{doublepagestwistodd*}[false]{}
\define@key{pdfpages}{duplicatepages}[2]{}
\define@key{pdfpages}{thread}[false]{}
\define@key{pdfpages}{threadname}{}
\define@key{pdfpages}{linkfit}{}
\define@key{pdfpages}{linktodoc}[false]{}
\define@key{pdfpages}{linktodocfit}{}
\define@key{pdfpages}{linkfilename}{}
\define@key{pdfpages}{survey}[false]{}
\define@key{pdfpages}{survey-nolink}[false]{}
\define@key{pdfpages}{newwindow}[false]{}
```

Use print mode while measuring the page numbers:

```
\xpretocmd{\AM@getpagecount}{\LWR@restoreorigformatting}{}{}
```

Emulate a bit of eso-pic:

```
\newif\ifESO@texcoord

\newcommand{\ESO@HookIIBG}{}

\renewcommand{\AM@AddToShipoutPicture}{\g@addto@macro\ESO@HookIIBG}

\renewcommand{\ClearShipoutPicture}{}
```

`\LWR@esopic@newpage` At each `\newpage`.

```
\newcommand*{\LWR@esopic@newpage}{%
```

Is there something to draw?

```
\ifdefvoid{\ESO@HookIIBG}%
{}%
{%
```

If the `link` option was specified, add a hyper taraget:

```
    \ifAM@link%
        \hypertarget{\AM@linkname.\AM@page}{}%
    \fi%
```

Draw inside a picture environment of the size of a virtual page:

```
    \begingroup%
    \setlength{\unitlength}{1in}%
    \begin{picture}(8,10.5)%
    \ESO@HookIIBG%
```

```
    \end{picture}%
    \endgroup%
    \global\let\ESO@HookIIBG\@empty%
}
}
```

\AM@output Patched to use \LWR@esopic@newpage.

```
\xpatchcmd{\AM@output}
    {\newpage}
    {\LWR@esopic@newpage}
    {}
    {\LWR@patcherror{pdfpages}{AM@output-1}}

\xpatchcmd{\AM@output}
    {\newpage}
    {\LWR@esopic@newpage}
    {}
    {\LWR@patcherror{pdfpages}{AM@output-2}}

\xpatchcmd{\AM@output}
    {\newpage}
    {\LWR@esopic@newpage}
    {}
    {\LWR@patcherror{pdfpages}{AM@output-3}}
```

\includepdf Patched to set a reasonable paper size.

```
\xpretocmd{\includepdf}{%
    \begingroup%
    \setlength{\paperwidth}{8in}%
    \setlength{\paperheight}{10.5in}%
}{}{}

\xapptocmd{\includepdf}{%
    \endgroup%
}{}{}
```

\includepdfmerge Patched to set a reasonable paper size.

```
\xpretocmd{\includepdfmerge}{%
    \begingroup%
    \setlength{\paperwidth}{8in}%
    \setlength{\paperheight}{10.5in}%
}{}{}

\xapptocmd{\includepdfmerge}{%
    \endgroup%
}{}{}
```

\AM@hyper@begin@i Hyper links are created by \LWR@esopic@newpage, so don't create them here:

```
\renewcommand{\AM@hyper@begin@i}{}
```

File 248 **lwarp-pdfprivacy.sty**

§ 345 Package pdfprivacy

Pkg pdfprivacy pdfprivacy is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pdfprivacy}[2017/12/03]
```

File 249 **lwarp-pdfrender.sty**

§ 346 Package pdfrender

Pkg pdfrender pdfrender is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pdfrender}[2016/05/17]

\newcommand*{\pdfrender}[1]{}
\newcommand{\textpdfrender}[2]{#2}
```

File 250 **lwarp-pdfsync.sty**

§ 347 Package pdfsync

(*Emulates or patches code by* J. Laurens.)

Pkg pdfsync Emulated.

for HTML output: Discard all options for lwarp-pdfsync:

```
\LWR@ProvidesPackageDrop{pdfsync}[2008/01/26]

\newcommand*{\pdfsync}{}
\newcommand*{\pdfsyncstart}{}
\newcommand*{\pdfsyncstop}{}
```

File 251 **lwarp-pdftricks.sty**

§ 348 Package pdftricks

(*Emulates or patches code by* C. V. Radhakrishnan, C. V. Rajagopal, Antoine Chambert-Loir.)

Pkg pdftricks 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:

```
\LWR@ProvidesPackagePass{pdftricks}[2003/08/10]
```

Reuse the print-mode images:

```
\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.

```
\PackageWarning{lwarp-pdftricks}{
When the pdftricks images change,
remember to convert PDF images to SVG using 'lwarpmk pdftosvg *-fig.pdf',
}

\AfterEndDocument{\typeout{***}}
\AfterEndDocument{\typeout{*** Note: If pdftricks images are not found, new, or updated,}}
\AfterEndDocument{\typeout{*** \space use 'lwarpmk pdftosvg \BaseJobname-fig*.pdf'}}
\AfterEndDocument{\typeout{***}}
```

File 252 **lwarp-pdfx.sty**

§ 349 Package pdfx

Pkg pdfx pdfx is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pdfx}[2017/05/18]
```

File 253 **lwarp-pfnote.sty**

§ 350 Package pfnote

Pkg pfnote pfnote is emulated.

⚠ 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:

```
\LWR@ProvidesPackageDrop{pfnote}[1999/07/14]
```

File 254 **lwarp-phfqit.sty**

§ 351 Package phfqit

(*Emulates or patches code by* Philippe Faist.)

Pkg phfqit phfqit is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{phfqit}[2017/08/16]

\LetLtxMacro\LWR@origbitstring\bitstring

\renewcommand\bitstring[1]{%
\InlineClass[%
    text-decoration: overline underline ;
]{bitstring}{#1}%
% \phfqit@bitstring{#1}%
}

\appto\LWR@restoreorigformatting{%
\LetLtxMacro\bitstring\LWR@origbitstring%
}
```

File 255 **lwarp-pifont.sty**

§ 352 Package pifont

(*Emulates or patches code by* Walter Schmidt.)

Pkg pifont 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:

```
\LWR@ProvidesPackagePass{pifont}[2005/04/12]

\renewcommand{\Pisymbol}[2]{%
    \begin{lateximage}*[(Pisymbol)][pisymbol#1#2]%
    {\Pifont{#1}\char#2}%
    \end{lateximage}%
}

\newcommand{\LWR@HTML@Pifill}[2]{
    \Pisymbol{#1}{#2} \Pisymbol{#1}{#2} \Pisymbol{#1}{#2}
}
\LWR@formatted{Pifill}

\newcommand{\LWR@HTML@Piline}[2]{%
    \par\noindent\hspace*{0.5in}
```

```
    \Pifill{#1}{#2} \Pifill{#1}{#2} \Pifill{#1}{#2}
}
\LWR@formatted{Piline}
```

File 256 **lwarp-placeins.sty**

§ 353 Package

placeins

(*Emulates or patches code by* Donald Arseneau.)

Pkg placeins placeins is not used during HTML conversion.

Discard all options for lwarp-placeins:

for HTML output:

```
\LWR@ProvidesPackageDrop{placeins}[2005/04/18]
\newcommand*{\FloatBarrier}{}
```

File 257 **lwarp-plarydshln.sty**

§ 354 Package

plarydshln

Pkg plarydshln plarydshln is emulated by lwarp-arydshln.

for HTML output:

```
\LWR@ProvidesPackageDrop{plarydshln}[2018/10/20]
\LWR@origRequirePackage{lwarp-arydshln}
```

File 258 **lwarp-plext.sty**

§ 355 Package

plext

Pkg plext plext is preloaded by jtarticle and related classes.

for HTML output:

```
\LWR@loadbefore{plext}

\LWR@ProvidesPackagePass{plext}[2017/07/21]

\let\tate\relax

\DeclareExpandableDocumentCommand{\rensuji}{s o m}{#3}

% \layoutfloat(width,height)[pos]#4
\DeclareDocumentCommand{\layoutfloat}{d() o m}{}

% \DeclareLayoutCaption{type} <dir>(width)[pos1pos2]
```

```
\DeclareDocumentCommand{\DeclareLayoutCaption}{m d<> d() o}{}

\LetLtxMacro\pcaption\caption

% \layoutcaption<dir>(width)[pos]
\DeclareDocumentCommand{\layoutcaption}{d<> d() o}{}

\let\captiondir\relax
```

Add the optional <t/y> direction:

```
\RenewDocumentEnvironment{LWR@HTML@minipage}{d<> O{t} O{} O{t} m}
    {\LWR@HTML@sub@minipage{#2}{#3}{#4}{#5}}
    {\endLWR@HTML@sub@minipage}

\RenewDocumentCommand{\LWR@HTML@parbox}{d<> O{t} O{} O{t} m +m}
{
\LWR@traceinfo{parbox of width #4}%
\begin{minipage}[#2][#3][#4]{#5}%
#6
\end{minipage}%
}

% \pbox <t/y> [width] [l/r] {contents}
\RenewDocumentCommand{\pbox}{d<> O{0pt} O{c} m}{%
\global\booltrue{LWR@minipagefullwidth}%
\parbox{#2}{#4}%
}
```

picture, as modified by pext, is encapsulated by the lwarp core.

File 259 **lwarp-plextarydshln.sty**

§ 356 Package plextarydshln

Pkg plextarydshln

plextarydshln is emulated by lwarp-arydshln.

for HTML output:

```
\LWR@ProvidesPackageDrop{plextarydshln}[2018/10/20]

\LWR@origRequirePackage{lwarp-arydshln}
```

File 260 **lwarp-plextcolortbl.sty**

§ 357 Package plextcolortbl

Pkg plextcolortbl

plextcolortbl is emulated by lwarp-colortbl.

for HTML output:

```
\LWR@ProvidesPackageDrop{plextcolortbl}[2018/09/19]
```

```
\LWR@origRequirePackage{lwarp-colortbl}
```

File 261 **lwarp-prelim2e.sty**

§ 358 Package prelim2e

(*Emulates or patches code by* Martin Schröder.)

Pkg prelim2e Emulated.

for HTML output: Discard all options for lwarp-prelim2e:

```
\LWR@ProvidesPackageDrop{prelim2e}[2009/05/29]

\newcommand{\PrelimText}{}
\newcommand{\PrelimTextStyle}{}
\newcommand{\PrelimWords}{}
```

File 262 **lwarp-prettyref.sty**

§ 359 Package prettyref

(*Emulates or patches code by* Kevin S. Ruland.)

Pkg prettyref prettyref is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{prettyref}[1998/07/09]

\newrefformat{fig}{Figure \ref{#1}}
\newrefformat{tab}{Table \ref{#1}}
```

File 263 **lwarp-preview.sty**

§ 360 Package preview

Pkg preview preview is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{preview}[2017/04/24]

\newenvironment{preview}{}{}
\newenvironment{nopreview}{}{}
\NewDocumentCommand{\PreviewMacro}{s o o +m}{}
\NewDocumentCommand{\PreviewEnvironment}{s o o +m}{}
\newcommand{\PreviewSnarfEnvironment}[2][]{}
\NewDocumentCommand{\PreviewOpen}{s o}{}
\NewDocumentCommand{\PreviewClose}{s o}{}
\let\ifPreview\iffalse% \fi for syntax highlighting
```

File 264 **lwarp-psfrag.sty**

§ 361 Package psfrag

(*Emulates or patches code by* Michael C. Grant, David Carlisle.)

Pkg psfrag 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:

```
\LWR@ProvidesPackagePass{psfrag}[1998/04/11]
```

A lateximage captures the modified image from the document.

```
\BeforeBeginEnvironment{psfrags}{%
    \begin{lateximage}[(-psfrags-~\packagediagramname)]%
}

\AfterEndEnvironment{psfrags}{\end{lateximage}}
```

File 265 **lwarp-psfragx.sty**

§ 362 Package psfragx

(*Emulates or patches code by* Pascal Kockaert.)

Pkg psfragx psfragx is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{psfragx}[2012/05/02]
```

A lateximage captures the modified image from the document.

```
\def\pfx@includegraphicx#1#2{%
    \begin{lateximage}[(-psfragx-~\packagediagramname)]%
    \mbox{\pfx@overpix{#1}{#2}\endpfx@overpix}%
    \end{lateximage}%
}

\def\@@@overpix[#1]<#2>[#3]#4{%
    \begin{lateximage}[(-psfragx-~\packagediagramname)]%
    \pfx@overpix{#1,ovpfgd={#2},ovpbgd={#3}}{#4}%
}

```

```
\def\endoverpix{%
    \endpfx@overpix%
    \end{lateximage}%
}
```

File 266 **lwarp-pst-eps.sty**

§ 363 Package pst-eps

(*Emulates or patches code by* Herbert Voss.)

Pkg pst-eps pst-eps is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{pst-eps}[2005/05/20]
```

```
\renewenvironment{TeXtoEPS}{}{}
\renewcommand{\PSTtoEPS}[3][]{}
```

File 267 **lwarp-pstool.sty**

§ 364 Package pstool

(*Emulates or patches code by* Zebb Prime, Will Robertson.)

Pkg pstool pstool is patched for use by lwarp.

`\graphicspath` is ignored, and the file directory must be stated.

⚠ path and filename The filename must not have a file extension.

Use

Enter ⇒ **lwarpmk html**

followed by

Enter ⇒ **lwarpmk limages**

.

for HTML output:

```
\LWR@ProvidesPackagePass{pstool}[2018/01/20]
```

Each image is placed inside a `lateximage` to capture the results of psfrag.

```
\renewcommand\pstool@alwaysprocess[3][]{%
    \begin{lateximage}[(-pstool-~\packagediagramname)]%
    \includegraphics{#2.pdf}%
    \end{lateximage}%
}
\LetLtxMacro\pstool@neverprocess\pstool@alwaysprocess
```

```
\LetLtxMacro\pstool@maybeprocess\pstool@alwaysprocess

\renewcommand\pstool@@psfragfig[4]{%
    \begin{lateximage}[(-pstool-~\packagediagramname)]%
    \includegraphics{#2.pdf}%
    \end{lateximage}%
}
```

File 268 **lwarp-pstricks.sty**

§ 365 Package

pstricks

(*Emulates or patches code by* Timothy Van Zandt.)

Pkg pstricks pstricks is patched for use by lwarp.

⚠ use pspicture All pstricks content should be contained inside a pspicture environment.

for HTML output:

```
\LWR@ProvidesPackagePass{pstricks}[2018/01/06]

\BeforeBeginEnvironment{pspicture}{\begin{lateximage}[(pspicture)]}
\AfterEndEnvironment{pspicture}{\end{lateximage}}
```

File 269 **lwarp-pxatbegshi.sty**

§ 366 Package

pxatbegshi

Pkg pxatbegshi pxatbegshi is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pxatbegshi}[2017/11/04]

\LWR@origRequirePackage{lwarp-atbegshi}
```

File 270 **lwarp-pxeveryshi.sty**

§ 367 Package

pxeveryshi

Pkg pxeveryshi pxeveryshi is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pxeveryshi}[2012/05/19]

\LWR@origRequirePackage{lwarp-everyshi}
```

File 271 **lwarp-pxftnright.sty**

§ 368 Package

pxftnright

Pkg pxftnright pxftnright is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pxftnright}[2017/02/28]
\LWR@origRequirePackage{lwarp-ftnright}
```

File 272 **lwarp-pxjahyper.sty**

§ 369 Package

pxjahyper

Pkg pxjahyper pxjahyper is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pxjahyper}[2018/07/15]
```

File 273 **lwarp-quotchap.sty**

§ 370 Package

quotchap

(*Emulates or patches code by* Karsten Tinnefeld, Jan Klever.)

Pkg quotchap quotchap is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{quotchap}[2012/10/20]

\newcommand{\@quotchap}{}
\newlength{\LWR@quotchapwidth}

\let\@printcites\relax

\newcommand*{\@iprintcites}{%
```

Place the quotes inside a of class quotchap, of the maximum selected width:

```
\begin{BlockClass}[max-width: \LWR@printlength{\LWR@quotchapwidth}]{quotchap}
%\begin{minipage}{\LWR@quotchapwidth}
\@quotchap
%\end{minipage}
\end{BlockClass}
```

Deactivate the quote printing:

```
\global\let\@printcites\relax
}

\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

```
\setlength{\LWR@quotchapwidth}{#1*2}%
\global\LWR@quotchapwidth=\LWR@quotchapwidth%
```

Remember the body, and activate the quote printing:

```
\global\let\@quotchap\BODY
\global\let\@printcites\@iprintcites%
}
```

The quotation author is placed inside a of class qauthor:

```
\newcommand{\qauthor}[1]{\begin{BlockClass}{qauthor}{#1}\end{BlockClass}}
```

\qsetcnfont is ignored:

```
\newcommand{\qsetcnfont}[1]{}
```

File 274 **lwarp-quoting.sty**

§ 371 Package

quoting

(*Emulates or patches code by* Thomas Titz.)

Pkg quoting quoting is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{quoting}[2014/01/28]

\xpatchcmd{\quoting}{\quo@begintext}
    {\begin{LWR@blocktextcurrentfont}\quo@begintext}
    {}
    {\LWR@patcherror{quoting}{quoting}}

\xpatchcmd{\endquoting}{\quo@endtext}
    {\quo@endtext\end{LWR@blocktextcurrentfont}}
    {}
    {\LWR@patcherror{imakeidx}{endquoting}}
```

File 275 **lwarp-ragged2e.sty**

§ 372 Package ragged2e

(*Emulates or patches code by* Martin Schröder.)

Pkg ragged2e ragged2e is not used during HTML conversion.

Discard all options for lwarp-ragged2e:

for HTML output:

```
\LWR@ProvidesPackageDrop{ragged2e}[2009/05/21]

\LetLtxMacro\Centering\centering
\LetLtxMacro\RaggedLeft\raggedleft
\LetLtxMacro\RaggedRight\raggedright
\newcommand*{\justifying}{}
\newlength{\CenteringLeftskip}
\newlength{\RaggedLeftLeftskip}
\newlength{\RaggedRightLeftskip}
\newlength{\CenteringRightskip}
\newlength{\RaggedLeftRightskip}
\newlength{\RaggedRightRightskip}
\newlength{\CenteringParfillskip}
\newlength{\RaggedLeftParfillskip}
\newlength{\RaggedRightParfillskip}
\newlength{\JustifyingParfillskip}
\newlength{\CenteringParindent}
\newlength{\RaggedLeftParindent}
\newlength{\RaggedRightParindent}
\newlength{\JustifyingParindent}
\newenvironment*{Center}{\center}{\endcenter}
\newenvironment*{FlushLeft}{\flushleft}{\endflushleft}
\newenvironment*{FlushRight}{\flushright}{\endflushright}
\newenvironment*{justify}{\justifying}{\endjustifying}
```

File 276 **lwarp-realscripts.sty**

§ 373 Package realscripts

(*Emulates or patches code by* Will Robertson.)

Pkg realscripts realscripts is emulated. See `lwarp.css` for the of class supsubscript.

for HTML output:

```
\LWR@ProvidesPackageDrop{realscripts}[2016/02/13]

\let\realsuperscript\textsuperscript
\let\realsubscript\textsubscript

```

```
\let\fakesuperscript\textsuperscript
\let\fakesubscript\textsubscript

\newlength{\subsupersep}

\newcommand*{\LWR@realscriptsalign}{}

\newcommand*{\LWR@setrealscriptsalign}[1]{%
    \renewcommand*{\LWR@realscriptsalign}{}%
    \ifthenelse{\equal{#1}{c}}{%
        \renewcommand{\LWR@realscriptsalign}{%
            \LWR@print@mbox{text-align:center} ; %
        }%
    }{}%
    \ifthenelse{\equal{#1}{r}}{%
        \renewcommand{\LWR@realscriptsalign}{%
            \LWR@print@mbox{text-align:right} ; %
        }%
    }{}%
}

\DeclareDocumentCommand \textsubsuperscript {s O{l} mm} {%
    \LWR@setrealscriptsalign{#2}%
    \InlineClass[\LWR@realscriptsalign]{supsubscript}{%
        \textsuperscript{#4}\textsubscript{#3}%
    }%
}

\DeclareDocumentCommand \textsupersubscript {s O{l} mm} {%
    \LWR@setrealscriptsalign{#2}%
    \InlineClass[\LWR@realscriptsalign]{supsubscript}{%
        \textsubscript{#4}\textsuperscript{#3}%
    }%
}
```

File 277 **lwarp-refcheck.sty**

§ 374 Package

refcheck

Pkg refcheck refcheck is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{refcheck}[2013/02/14]

\def\showrefnames{}
\def\norefnames{}
\def\showcitenames{}
\def\nocitenames{}
\def\setonmsgs{}
\def\setoffmsgs{}
\def\checkunlbld{}
\def\ignoreunlbld{}
\newcommand*{\refcheckxrdoc}[2][]{}
```

File 278 **lwarp-register.sty**

§ 375 Package register

(*Emulates or patches code by* Matthew Lovell.)

Pkg register register is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{register}[2019/01/01]

\xpatchcmd{\register}
    {\centering}
    {\begin{center}\begin{lateximage}[(-register-~\packagediagramname)]}
    {}
    {\LWR@patcherror{register}{register}}

\xpatchcmd{\endregister}
    {\leftskip}
    {%
        \end{lateximage}\end{center}%
        \leftskip%
    }%
    {}
    {\LWR@patcherror{register}{endregister}}

\expandafter\xapptocmd\csname register*\endcsname
    {\begin{center}\begin{lateximage}[(-register-~\packagediagramname)]}
    {}
    {\LWR@patcherror{register}{register*}}

\expandafter\xpatchcmd\csname endregister*\endcsname
    {\leftskip}
    {%
        \end{lateximage}\end{center}%
        \leftskip%
    }%
    {}
    {\LWR@patcherror{register}{endregister*}}

\setlength{\regWidth}{5in}
```

File 279 **lwarp-relsize.sty**

§ 376 Package relsize

(*Emulates or patches code by* Donald Arseneau, Bernie Cosell, Matt Swift.)

Pkg relsize relsize is patched for use by lwarp.

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:

```
\LWR@ProvidesPackagePass{relsize}[2013/03/29]

\let\LWR@origrelsize\relsize
\LetLtxMacro\LWR@origlarger\larger
\LetLtxMacro\LWR@origsmaller\smaller
\let\LWR@relscale\relscale
\LetLtxMacro\LWR@origtextlarger\textlarger
\LetLtxMacro\LWR@origtextsmaller\textsmaller
\let\LWR@textscale\textscale

\appto\LWR@restoreorigformatting{%
\let\relsize\LWR@origrelsize%
\LetLtxMacro\larger\LWR@origlarger%
\LetLtxMacro\smaller\LWR@origsmaller%
\let\relscale\LWR@relscale%
\LetLtxMacro\textlarger\LWR@origtextlarger%
\LetLtxMacro\textsmaller\LWR@origtextsmaller%
\let\textscale\LWR@textscale%
}

\newcounter{LWR@relsizetemp}

\renewcommand*{\relsize}[1]{}
\renewcommand*{\larger}[1][]{}
\renewcommand*{\smaller}[1][]{}
\renewcommand*{\relscale}[1]{}

\renewcommand*{\textlarger}[2][1]{%
\setcounter{LWR@relsizetemp}{100+(#1*20)}%
\InlineClass[font-size:\arabic{LWR@relsizetemp}\%]{textlarger}{#2}%
}

\renewcommand*{\textsmaller}[2][1]{%
\setcounter{LWR@relsizetemp}{100-(#1*20)}%
\InlineClass[font-size:\arabic{LWR@relsizetemp}\%]{textsmaller}{#2}%
}

\renewcommand*{\textscale}[2]{%
\setcounter{LWR@relsizetemp}{100*\real{#1}}%
\InlineClass[font-size:\arabic{LWR@relsizetemp}\%]{textscale}{#2}%
}
```

File 280 **lwarp-repeatindex.sty**

§ 377 Package

repeatindex

Pkg repeatindex 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:

```
\LWR@ProvidesPackageDrop{repeatindex}[2001/10/13]
```

In the lwarp core, `\LWR@indexitem` is modified to accept the optional `\item` argument.

```
\RequirePackage{makeidx}
\def\entryprefix{\itshape}
\def\entrypostfix{\dots}
```

File 281 **lwarp-resizegather.sty**

§ 378 Package resizegather

Pkg resizegather resizegather is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{resizegather}[2016/05/16]
\newcommand*{\resizegathersetup}[1]{}
```

File 282 **lwarp-rmpage.sty**

§ 379 Package rmpage

Pkg rmpage rmpage is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{rmpage}[1997/09/29]
```

File 283 **lwarp-romanbar.sty**

§ 380 Package romanbar

(*Emulates or patches code by* H.-Martin Münch.)

Pkg romanbar romanbar is patched for use by lwarp.

An inline class with an overline and underline is used.

for HTML output:

```
\LWR@ProvidesPackagePass{romanbar}[2012/01/01]
\DeclareRobustCommand{\Roman@bar}[1]{% #1 is in Roman, i.e. MMXII
\InlineClass[%
    text-decoration: overline underline ;
]{romanbar}{#1}%
}
```

File 284 **lwarp-romanbarpagenumber.sty**

§ 381 Package romanbarpagenumber

Pkg romanbarpagenumber romanbarpagenumber is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{romanbarpagenumber}[2015/02/06]
```

File 285 **lwarp-rotating.sty**

§ 382 Package rotating

(*Emulates or patches code by* Robin Fairbairns, Sebastian Rahtz, Leonor Barroca.)

Pkg rotating rotating is emulated.

All rotations are ignored in HTML output.

for HTML output:

```
\LWR@ProvidesPackageDrop{rotating}[2016/08/11]

\LetLtxMacro\sidewaystable\table
\let\endsidewaystable\endtable

\LetLtxMacro\sidewaysfigure\figure
\let\endsidewaysfigure\endfigure

\newenvironment*{sideways}{}{}
\newenvironment*{turn}[1]{}{}
\newenvironment*{rotate}[1]{}{}
\NewDocumentCommand{\turnbox}{m +m}{#2}
\let\rotcaption\caption
\let\@makerotcaption\@makecaption
```

File 286 **lwarp-rotfloat.sty**

§ 383 Package rotfloat

(*Emulates or patches code by* Axel Sommerfeldt.)

Pkg rotfloat rotfloat is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{rotfloat}[2004/01/04]

\RequirePackage{float}
```

\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.

```
\RenewDocumentCommand{\newfloat}{m m m o}{%
\IfValueTF{#4}%
{%
    \DeclareFloatingEnvironment[fileext=#3,within=#4]{#1}%
```

```
}%
{%
    \DeclareFloatingEnvironment[fileext=#3]{#1}%
}%
\csletcs{sideways#1}{#1}%
\csletcs{endsideways#1}{end#1}%
```

Remember the float style:

```
\csedef{LWR@floatstyle@#1}{\LWR@floatstyle}%
\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:

```
\cslet{listof#1s}\relax%
\cslet{listof#1es}\relax%
\cslet{listofsideways#1s}\relax%
\cslet{listofsideways#1es}\relax%
}
```

File 287 **lwarp-rviewport.sty**

§ 384 Package rviewport

Pkg rviewport

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:

```
\LWR@ProvidesPackagePass{rviewport}[2011/08/27]
```

```
\define@key{igraph}{rviewport}{}
```

File 288 **lwarp-savetrees.sty**

§ 385 Package savetrees

Pkg savetrees

Emulated.

for HTML output:

Discard all options for lwarp-savetrees:

```
\LWR@ProvidesPackageDrop{savetrees}[2016/04/13]
```

File 289 **lwarp-scalefnt.sty**

§ 386 Package scalefnt

(*Emulates or patches code by* D. Carlisle.)

Pkg scalefnt scalefnt is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{scalefnt}

\DeclareRobustCommand\scalefont[1]{}
```

File 290 **lwarp-schemata.sty**

§ 387 Package schemata

(*Emulates or patches code by* Charles P. Schaum.)

Pkg schemata schemata is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{schemata}[2016/01/25]

\LetLtxMacro\LWR@schemata@origschema\schema
\LetLtxMacro\LWR@schemata@origSchema\Schema

\renewcommand{\schema}[3][open]{%
\begin{lateximage}%
\LWR@print@normalsize
\LWR@schemata@origschema[#1]{#2}{#3}%
\end{lateximage}%
}

\renewcommand{\Schema}[5][open]{%
\begin{lateximage}%
\LWR@print@normalsize
\LWR@schemata@origSchema[#1]{#2}{#3}{#4}{#5}%
\end{lateximage}%
}
```

File 291 **lwarp-scrextend.sty**

§ 388 Package scrextend

Pkg scrextend 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:

```
\LWR@ProvidesPackageDrop{scrextend}[2018/03/30]

\DeclareDocumentCommand{\setkomafont}{m m}{}
\DeclareDocumentCommand{\addkomafont}{m m}{}
\DeclareDocumentCommand{\usekomafont}{m}{}

\DeclareDocumentCommand{\usefontofkomafont}{m}{}
\DeclareDocumentCommand{\useencodingofkomafont}{m}{}
\DeclareDocumentCommand{\usesizeofkomafont}{m}{}
\DeclareDocumentCommand{\usefamilyofkomafont}{m}{}
\DeclareDocumentCommand{\useseriesofkomafont}{m}{}
\DeclareDocumentCommand{\useshapeofkomafont}{m}{}

\AtBeginDocument{
\let\LWR@maketitle\maketitle
\DeclareDocumentCommand{\maketitle}{o}{\LWR@maketitle}
}

\DeclareDocumentCommand{\extratitle}{m}{}
\DeclareDocumentCommand{\titlehead}{m}{}
\DeclareDocumentCommand{\subject}{m}{}
\DeclareDocumentCommand{\publishers}{m}{\published{#1}}
\DeclareDocumentCommand{\uppertitleback}{m}{}
\DeclareDocumentCommand{\lowertitleback}{m}{}
\DeclareDocumentCommand{\dedication}{m}{}

\DeclareDocumentCommand{\ifthispageodd}{m m}{#1}

\DeclareDocumentCommand{\titlepagestyle}{}{}

\DeclareDocumentCommand{\cleardoublepageusingstyle}{m}{}
\DeclareDocumentCommand{\cleardoubleemptypage}{}{}
\DeclareDocumentCommand{\cleardoubleplainpage}{}{}
\DeclareDocumentCommand{\cleardoublestandardpage}{}{}
\DeclareDocumentCommand{\cleardoubleoddpage}{}{}
\DeclareDocumentCommand{\cleardoubleoddpageusingstyle}{m}{}
\DeclareDocumentCommand{\cleardoubleoddemptypage}{}{}
\DeclareDocumentCommand{\cleardoubleoddplainpage}{}{}
\DeclareDocumentCommand{\cleardoubleoddstandardpage}{}{}
\DeclareDocumentCommand{\cleardoubleevenpage}{}{}
\DeclareDocumentCommand{\cleardoubleevenpageusingstyle}{m}{}
\DeclareDocumentCommand{\cleardoubleevenemptypage}{}{}
\DeclareDocumentCommand{\cleardoubleevenplainpage}{}{}
\DeclareDocumentCommand{\cleardoubleevenstandardpage}{}{}

\DeclareDocumentCommand{\multiplefootnoteseparator}{}{%
  \begingroup\let\thefootnotemark\multfootsep\@makefnmark\endgroup
}

\DeclareDocumentCommand{\multfootsep}{}{,}

```

```
\DeclareDocumentCommand{\footref}{m}{%
  \begingroup
    \unrestored@protected@xdef\@thefnmark{\ref{#1}}%
  \endgroup
  \@footnotemark
}

\DeclareDocumentCommand{\deffootnote}{o m m m}{}
\DeclareDocumentCommand{\deffootnotemark}{m}{}
\DeclareDocumentCommand{\setfootnoterule}{o m}{}
\DeclareDocumentCommand{\raggedfootnote}{}{}

\DeclareDocumentCommand{\dictum}{o m}{
\begin{LWR@BlockClassWP}{\LWR@print@mbox{text-align:right}}{}{dictum}
    #2
    \IfValueT{#1}
    {
        \ifbool{FormatWP}
     {\begin{BlockClass}[\LWR@print@mbox{border-top:} 1px solid gray]{dictumauthor}}
        {\begin{BlockClass}{dictumauthor}}
        \dictumauthorformat{#1}
        \end{BlockClass}
    }
\end{LWR@BlockClassWP}
}

\DeclareDocumentCommand{\dictumwidth}{}{}
\DeclareDocumentCommand{\dictumauthorformat}{m}{(#1)}
\DeclareDocumentCommand{\dictumrule}{}{}
\DeclareDocumentCommand{\raggeddictum}{}{}
\DeclareDocumentCommand{\raggeddictumtext}{}{}
\DeclareDocumentCommand{\raggeddictumauthor}{}{}

\DeclareDocumentEnvironment{labeling}{o m}
{%
\def\sc@septext{#1}%
\list{}{}%
\let\makelabel\labelinglabel%
}
{
\endlist
}

\DeclareDocumentCommand{\labelinglabel}{m}{%
#1 \qquad \sc@septext%
}

\let\addmargin\relax
\let\endaddmargin\relax
\cslet{addmargin*}{\relax}
\cslet{endaddmargin*}{\relax}

\NewDocumentEnvironment{addmargin}{s O{} m}
{
\setlength{\LWR@templengthtwo}{#3}
```

```
\ifblank{#2}
{
    \begin{BlockClass}[
        \LWR@print@mbox{margin-left:\LWR@printlength{\LWR@templengthtwo}} ;
        \LWR@print@mbox{margin-right:\LWR@printlength{\LWR@templengthtwo}}
    ]{addmargin}
}
{
    \setlength{\LWR@templengthone}{#2}
    \begin{BlockClass}[
        \LWR@print@mbox{margin-left:\LWR@printlength{\LWR@templengthone}} ;
        \LWR@print@mbox{margin-right:\LWR@printlength{\LWR@templengthtwo}}
    ]{addmargin}
}
}
{\end{BlockClass}}
```

Ref to create a starred environment:
https://tex.stackexchange.com/questions/45401/use-the-s-star-argument-with-newdocumentenvironment

```
\ExplSyntaxOn
\cs_new:cpn {addmargin*} {\addmargin*}
\cs_new_eq:cN {endaddmargin*} \endaddmargin
\ExplSyntaxOff

\DeclareDocumentCommand{\marginline}{m}{\marginpar{#1}}
```

File 292 **lwarp-scrhack.sty**

§ 389 Package scrhack

Pkg scrhack scrhack is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{scrhack}[2018/03/30]
```

File 293 **lwarp-scrlayer.sty**

§ 390 Package scrlayer

(*Emulates or patches code by* Markus Kohm.)

Pkg scrlayer scrlayer is emulated.

⚠ Not fully tested! Please send bug reports!

for HTML output:

```
\LWR@ProvidesPackageDrop{scrlayer}[2018/03/30]
```

```
\newcommand*{\DeclareSectionNumberDepth}[2]{}
\newcommand*{\DeclareLayer}[2][]{}
\newcommand*{\DeclareNewLayer}[2][]{}
\newcommand*{\ProvideLayer}[2][]{}
\newcommand*{\RedeclareLayer}[2][]{}
\newcommand*{\ModifyLayer}[2][]{}
\newcommand*{\layerhalign}{}
\newcommand*{\layervalign}{}
\newcommand*{\layerxoffset}{}
\newcommand*{\layeryoffset}{}
\newcommand*{\layerwidth}{}
\newcommand*{\layerheight}{}
\providecommand*{\LenToUnit}[1]{\strip@pt\dimexpr#1*\p@/\unitlength}
\newcommand*{\putUL}[1]{}
\newcommand*{\putUR}[1]{}
\newcommand*{\putLL}[1]{}
\newcommand*{\putLR}[1]{}
\newcommand*{\putC}[1]{}
\newcommand*{\GetLayerContents}[1]{}
\newcommand{\IfLayerExists}[3]{#3}
\newcommand*{\DestroyLayer}[1]{}
\newcommand*{\layercontentsmeasure}{}
\newcommand*{\currentpagestyle}{}
\newcommand*{\BeforeSelectAnyPageStyle}[1]{}
\newcommand*{\AfterSelectAnyPageStyle}[1]{}
\newcommand*{\DeclarePageStyleAlias}[2]{}
\newcommand*{\DeclareNewPageStyleAlias}[2]{}
\newcommand*{\ProvidePageStyleAlias}[2]{}
\newcommand*{\RedeclarePageStyleAlias}[2]{}
\newcommand*{\DestroyPageStyleAlias}[1]{}
\newcommand*{\GetRealPageStyle}[1]{}
\newcommand*{\DeclarePageStyleByLayers}[3][]{}
\newcommand*{\DeclareNewPageStyleByLayers}[3][]{}
\newcommand*{\ProvidePageStyleByLayers}[3][]{}
\newcommand*{\RedeclarePageStyleByLayers}[3][]{}
\NewDocumentCommand{\ForEachLayerOfPageStyle}{s m m}{}
\newcommand*{\AddLayersToPageStyle}[2]{}
\newcommand*{\AddLayersAtBeginOfPageStyle}[2]{}
\newcommand*{\AddLayersAtEndOfPageStyle}[2]{}
\newcommand*{\RemoveLayersFromPageStyle}[2]{}
\newcommand*{\AddLayersToPageStyleBeforeLayer}[3]{}
\newcommand*{\AddLayersToPageStyleAfterLayer}[3]{}
\newcommand*{\UnifyLayersAtPageStyle}[1]{}
\newcommand*{\ModifyLayerPageStyleOptions}[2]{}
\newcommand*{\AddToLayerPageStyleOptions}[2]{}
\newcommand{\IfLayerPageStyleExists}[3]{#3}
\newcommand{\IfRealLayerPageStyleExists}[3]{#3}
\newcommand{\IfLayerAtPageStyle}[4]{#4}
\newcommand{\IfSomeLayerAtPageStyle}[4]{#4}
\newcommand{\IfLayersAtPageStyle}[4]{#4}
\newcommand*{\DestroyRealLayerPageStyle}[1]{}
\@ifundefined{footheight}{\newlength\footheight}{}
\DeclareDocumentCommand{\automark}{s o m}{}
\DeclareDocumentCommand{\manualmark}{}{}
\DeclareDocumentCommand{\MakeMarkcase}{m}{#1}
```

```
\newcommand{\partmarkformat}{}
\if@chapter
\newcommand{\chaptermarkformat}{}
\fi
\newcommand{\sectionmarkformat}{}
\DeclareDocumentCommand{\GenericMarkFormat}{m}{}

\newcommand*{\@mkleft}[1]{}
\newcommand*{\@mkright}[1]{}
\newcommand*{\@mkdouble}[1]{}
\newcommand*{\@mkboth}[2]{}
\newcommand*{\scrlayerInitInterface}[1][]{}
\newcommand{\scrlayerAddToInterface}[3][]{}
\newcommand{\scrlayerAddCsToInterface}[3][]{}
\newcommand{\scrlayerOnAutoRemoveInterface}[2][]{}
```

File 294 **lwarp-scrlayer-notecolumn.sty**

§ 391 Package scrlayer-notecolumn

(*Emulates or patches code by* Markus Kohm.)

Pkg scrlayer-notecolumn scrlayer-notecolumn is emulated.

⚠ Not fully tested! Please send bug reports!

for HTML output:

```
\LWR@ProvidesPackageDrop{scrlayer-notecolumn}[2018/02/02]

\newcommand*{\DeclareNoteColumn}[2][]{}
\newcommand*{\DeclareNewNoteColumn}[2][]{}
\newcommand*{\ProvideNoteColumn}[2][]{}
\newcommand*{\RedeclareNoteColumn}[2][]{}
\NewDocumentCommand{\makenote}{s o m}{\marginpar{#3}}
\newcommand*{\syncwithnotecolumn}[1][]{}
\newcommand*{\syncwithnotecolumns}[1][]{}
\newcommand*{\clearnotecolumn}[1][]{}
\newcommand*{\clearnotecolumns}[1][]{}
```

File 295 **lwarp-scrlayer-scrpage.sty**

§ 392 Package scrlayer-scrpage

(*Emulates or patches code by* Markus Kohm.)

Pkg scrlayer-scrpage scrlayer-scrpage is emulated.

⚠ Not fully tested! Please send bug reports!

for HTML output:

```
\LWR@ProvidesPackageDrop{scrlayer-scrpage}[2018/03/30]
```

```
\@ifundefined{footheight}{\newlength\footheight}{}
\NewDocumentCommand{\lehead}{s o m}{}
\NewDocumentCommand{\cehead}{s o m}{}
\NewDocumentCommand{\rehead}{s o m}{}
\NewDocumentCommand{\lohead}{s o m}{}
\NewDocumentCommand{\cohead}{s o m}{}
\NewDocumentCommand{\rohead}{s o m}{}
\NewDocumentCommand{\lefoot}{s o m}{}
\NewDocumentCommand{\cefoot}{s o m}{}
\NewDocumentCommand{\refoot}{s o m}{}
\NewDocumentCommand{\lofoot}{s o m}{}
\NewDocumentCommand{\cofoot}{s o m}{}
\NewDocumentCommand{\rofoot}{s o m}{}
\NewDocumentCommand{\ohead}{s o m}{}
\NewDocumentCommand{\chead}{s o m}{}
\NewDocumentCommand{\ihead}{s o m}{}
\NewDocumentCommand{\ofoot}{s o m}{}
\NewDocumentCommand{\cfoot}{s o m}{}
\NewDocumentCommand{\ifoot}{s o m}{}
\DeclareDocumentCommand{\MakeMarkcase}{m}{#1}
\newcommand*{\defpairofpagestyles}[3][]{}
\newcommand*{\newpairofpagestyles}[3][]{}
\newcommand*{\renewpairofpagestyles}[3][]{}
\newcommand*{\providepairofpagestyles}[3][]{}

\newcommand*{\clearmainofpairofpagestyles}{}
\newcommand*{\clearplainofpairofpagestyles}{}
\newcommand*{\clearpairofpagestyles}{}
\newcommand*{\clearscrheadings}{}
\newcommand*{\clearscrheadfoot}{}
\newcommand*{\clearscrplain}{}

\NewDocumentCommand{\deftriplepagestyle}{m o o m m m m m m}{}
\NewDocumentCommand{\newtriplepagestyle}{m o o m m m m m m}{}
\NewDocumentCommand{\renewtriplepagestyle}{m o o m m m m m m}{}
\NewDocumentCommand{\providetriplepagestyle}{m o o m m m m m m}{}
\newcommand*{\defpagestyle}[3]{}
\newcommand*{\newpagestyle}[3]{}
\newcommand*{\providepagestyle}[3]{}
\newcommand*{\renewpagestyle}[3]{}
```

File 296 **lwarp-scrpage2.sty**

§ 393 Package

scrpage2

(*Emulates or patches code by* Markus Kohm.)

Pkg scrpage2 scrpage2 is emulated.

⚠ Not fully tested! Please send bug reports!

for HTML output:

```
\LWR@ProvidesPackageDrop{scrpage2}[2018/03/30]
```

```
\@ifundefined{footheight}{\newlength\footheight}{}
\NewDocumentCommand{\lehead}{o m}{}
\NewDocumentCommand{\cehead}{o m}{}
\NewDocumentCommand{\rehead}{o m}{}
\NewDocumentCommand{\lohead}{o m}{}
\NewDocumentCommand{\cohead}{o m}{}
\NewDocumentCommand{\rohead}{o m}{}
\NewDocumentCommand{\lefoot}{o m}{}
\NewDocumentCommand{\cefoot}{o m}{}
\NewDocumentCommand{\refoot}{o m}{}
\NewDocumentCommand{\lofoot}{o m}{}
\NewDocumentCommand{\cofoot}{o m}{}
\NewDocumentCommand{\rofoot}{o m}{}
\NewDocumentCommand{\ohead}{o m}{}
\NewDocumentCommand{\chead}{o m}{}
\NewDocumentCommand{\ihead}{o m}{}
\NewDocumentCommand{\ofoot}{o m}{}
\NewDocumentCommand{\cfoot}{o m}{}
\NewDocumentCommand{\ifoot}{o m}{}
\DeclareDocumentCommand{\automark}{o m}{}
\DeclareDocumentCommand{\manualmark}{}{}
\DeclareDocumentCommand{\MakeMarkcase}{m}{#1}
\NewDocumentCommand{\deftripstyle}{m o o m m m m m}{}
\NewDocumentCommand{\defpagestyle}{s m m m}{}
\NewDocumentCommand{\newpagestyle}{s m m m}{}
\NewDocumentCommand{\renewpagestyle}{s m m m}{}
\NewDocumentCommand{\providepagestyle}{s m m m}{}
\newcommand{\partmarkformat}{}
\if@chapter
\newcommand{\chaptermarkformat}{}
\fi
\newcommand{\sectionmarkformat}{}
\newcommand{\subsectionmarkformat}{}
\newcommand{\subsubsectionmarkformat}{}
\newcommand{\paragraphmarkformat}{}
\newcommand{\subparagraphmarkformat}{}

\newcommand*{\clearscrheadings}{}
\newcommand*{\clearscrheadfoot}{}
\newcommand*{\clearscrplain}{}
```

File 297 **lwarp-section.sty**

§ 394 Package

section

Pkg section section is ignored.

(*Emulates or patches code by* Oliver Pretzel.)

for HTML output:

```
\LWR@ProvidesPackageDrop{section}

\ifx\chapter\undefined
 \def\chsize{\Large}\def\hdsize{\huge}\else
```

```
 \def\chsize{\huge}\def\hdsize{\Huge}
\fi
\let\ttsize\LARGE
\let\ausize\large
\let\dasize\large
\let\secsize\Large
\let\subsize\large
\let\hdpos\raggedright
\newcounter{hddepth}
\let\fpind\relax
\def\ttfnt{}
\def\hdfnt{}
\def\fefnt{}
\def\thfnt{}
\def\pgfnt{}
\def\hmkfnt{}
\let\mkcse\uppercase
\def\hddot{}
\def\cpdot{:}
\def\nmdot{}
\ifx\secindent\undefined
 \newdimen\secindent
 \newskip\secpreskp
 \newskip\secpstskp
 \newdimen\subindent
 \newskip\subpreskp
 \newskip\subpstskp
 \newskip\parpstskp
 \newcount\c@hddepth
\fi
```

File 298 **lwarp-sectionbreak.sty**

§ 395 Package sectionbreak

(*Emulates or patches code by* Michal Hoftich.)

Pkg sectionbreak sectionbreak is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{sectionbreak}[2018-01-03]

\renewcommand\asterism{\HTMLunicode{2042}}

\renewcommand\pre@sectionbreak{}
\renewcommand\post@sectionbreak{}

\renewcommand\print@sectionbreak[1]{%
\begin{center}
#1
\end{center}
}

```

File 299 **lwarp-sectsty.sty**

§ 396 Package sectsty

(*Emulates or patches code by* Rowland McDonnell.)

Pkg sectsty sectsty is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{sectsty}[2002/02/25]

\newcommand*{\partfont}          [1] {}
\newcommand*{\partnumberfont}    [1] {}
\newcommand*{\parttitlefont}     [1] {}
\newcommand*{\chapterfont}       [1] {}
\newcommand*{\chapternumberfont} [1] {}
\newcommand*{\chaptertitlefont}  [1] {}
\newcommand*{\sectionfont}       [1] {}
\newcommand*{\subsectionfont}    [1] {}
\newcommand*{\subsubsectionfont} [1] {}
\newcommand*{\paragraphfont}     [1] {}
\newcommand*{\subparagraphfont}  [1] {}
\newcommand*{\minisecfont}  [1] {}
\newcommand*{\allsectionsfont}[1] {}
\newcommand{\nohang}{}
```

`\sectionrule` is only to be used in `*font` commands, thus it is ignored.

```
\newcommand*{\sectionrule}[5]{}

\def\ulemheading#1#2{}
```

File 300 **lwarp-semantic-markup.sty**

§ 397 Package semantic-markup

(*Emulates or patches code by* Andrew A. Cashner.)

Pkg semantic-markup semantic-markup is patched for use by lwarp.

⚠ If using the `endnotes` option, add `\theendnotes` where desired.

for HTML output:

```
\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.

```
\ifendnotes
\RenewDocumentCommand{\SetupEndnotes}{}{%
```

```
    \let\footnote=\endnote
%     \AtEndDocument{\DoBeforeEndnotes{\EndnoteFont\theendnotes}}%
}
\fi
```

HTML unicode characters from musicography are used.

```
\RequirePackage{musicography}

\let\fl\musFlat
\let\sh\musSharp
\let\na\musNatural
```

The `\musfig` is placed inside a hashed image, with a simple `alt` tag.

```
\RequirePackage{amsmath}

\RenewDocumentCommand{\musfig}{ m m }{%
    \LWR@subsingledollar*%
        {#1/#2}% alt tag
        {musfig}% addl' hashing
        {% contents
            \LWR@origensuredmath{%
                \genfrac{}{}{0pt}{1}{\text{#1}}{\text{#2}}%
            }%
        }%
}
```

The `\meter` is taken from musicography, and becomes a hashed image with a simple `alt` tag.

```
\RenewDocumentCommand{\meter}{ m m }{%
    \musMeter{#1}{#2}%
}
```

File 301 **lwarp-setspace.sty**

§ 398 Package

setspace

(*Emulates or patches code by* Robin Fairbairns.)

Pkg setspace setspace is not used during HTML conversion.

Discard all options for lwarp-setspace:

for HTML output:

```
\LWR@ProvidesPackageDrop{setspace}[2011/12/19]

\newcommand*{\setstretch}[1]{}
\newcommand*{\SetSinglespace}[1]{}
\newcommand*{\singlespacing}{}
\newcommand*{\onehalfspacing}{}
\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 302 **lwarp-shadow.sty**

§ 399 Package

shadow

(*Emulates or patches code by* Mauro Orlandini.)

Pkg shadow 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 303 **lwarp-showidx.sty**

§ 400 Package showidx

Pkg showidx showidx is ignored.

for HTML output: Discard all options for lwarp-showidx:

```
1 \LWR@ProvidesPackageDrop{showidx}[2014/09/29]
```

\@wrindex is redefined \AtBeginDocument by the lwarp core.

File 304 **lwarp-showkeys.sty**

§ 401 Package showkeys

(*Emulates or patches code by* David Carlisle, Morten Høgholm.)

Pkg showkeys showkeys is ignored.

for HTML output: Discard all options for lwarp-showkeys:

```
1 \LWR@ProvidesPackageDrop{showkeys}[2014/10/28]
```

```
2 \NewDocumentCommand{\showkeys}{s}{}
```

File 305 **lwarp-sidecap.sty**

§ 402 Package sidecap

(*Emulates or patches code by* Rolf Niepraschk, Hubert Gässlein.)

Pkg sidecap 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}}
```

```
{\endtable}

\ExplSyntaxOn
\cs_new:cpn {SCtable*} {\SCtable*}
\cs_new_eq:cN {endSCtable*} \endSCtable
\ExplSyntaxOff


\NewDocumentEnvironment{SCfigure}{soo}
{\IfValueTF{#3}{\figure[#3]}{\figure}}
{\endfigure}

\ExplSyntaxOn
\cs_new:cpn {SCfigure*} {\SCfigure*}
\cs_new_eq:cN {endSCfigure*} \endSCfigure
\ExplSyntaxOff


\newenvironment*{wide}{}{}
```

File 306 **lwarp-sidenotes.sty**

§ 403 Package

sidenotes

(*Emulates or patches code by* Andy Thomas, Oliver Schebaum.)

Pkg sidenotes Patched for lwarp.

for HTML output: Load the original package:

```
\LWR@ProvidesPackagePass{sidenotes}
```

The following patch sidenotes for use with lwarp:

\sidecaption * [⟨*entry*⟩] [⟨*offset*⟩] {⟨*text*⟩}

```
\RenewDocumentCommand \sidecaption {s o o +m}
{
   \LWR@stoppars
   \begingroup
  \captionsetup{style=sidecaption}
  \IfBooleanTF{#1}
  { % starred
    \begin{BlockClass}[border:none ; box-shadow:none]{marginblock}
    \caption*{#4}
    \end{BlockClass}
  }
  { % unstarred
  \IfNoValueOrEmptyTF{#2}
    {\def\@sidenotes@sidecaption@tof{#4}}
    {\def\@sidenotes@sidecaption@tof{#2}}
    \begin{BlockClass}[border:none ; box-shadow:none]{marginblock}
    \caption[\@sidenotes@sidecaption@tof]{#4}
```

```
    \end{BlockClass}
  }
    \endgroup
    \LWR@startpars
}
```

Borrowed from the lwarp version of keyfloat:

```
\NewDocumentEnvironment{KFLTsidenotes@marginfloat}{O{-1.2ex} m}
{% start
\LWR@BlockClassWP{float:right; width:2in; margin:10pt}{}{marginblock}%
\captionsetup{type=#2}%
}
{%
\endLWR@BlockClassWP%
}

\RenewDocumentEnvironment{marginfigure}{o}
  {\begin{KFLTsidenotes@marginfloat}{figure}}
  {\end{KFLTsidenotes@marginfloat}}

\RenewDocumentEnvironment{margintable}{o}
  {\begin{KFLTsidenotes@marginfloat}{table}}
  {\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 2$_\varepsilon$ `article.cls` source:

```
\renewenvironment{figure*}
               {\@dblfloat{figure}}
               {\end@dblfloat}

\renewenvironment{table*}
               {\@dblfloat{table}}
               {\end@dblfloat}
```

File 307 **lwarp-SIunits.sty**

§ 404 Package

SIunits

(*Emulates or patches code by* Marcel Heldoorn.)

Pkg SIunits SIunits is patched for use by lwarp.

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 to "`(<\mathimagename>)`", which defaults to "`(math image)`".

for HTML output:

```
\LWR@ProvidesPackagePass{SIunits}[2007/12/02]
```

Patched for copy/paste with the HTML `alt` tag:

```
\DeclareRobustCommand{\unit}[2]{%
\@inunitcommandtrue%    original
\LWR@subsingledollar*%  lwarp
{% alt tag
    \textbackslash{}unit\{\LWR@HTMLsanitize{#1}\}%
        \{ \LWR@HTMLsanitize{#2}\}% extra space
}%
{SIunits}% add'l hashing
{%
    \LWR@origensuredmath{%  lwarp modification
        \SI@fstyle{#1\@qsk\period@active{#2}}%  original
    }%
}% contents
\@inunitcommandfalse%   original
}
```

File 308 **lwarp-siunitx.sty**

§ 405 Package

siunitx

(*Emulates or patches code by* Joseph Wright.)

Pkg siunitx

siunitx is patched for use by lwarp.

fractions

Due to *pdftotext* limitations, fraction output is replaced by symbol output for `per-mode` and `quotient-mode`.

⚠ math mode required

Some units will require that the expression be placed inside math mode.

NOTE: As of this writing, the siunitx extension for MathJax is not currently hosted at any public CDN, thus siunitx is not usable with MathJax unless a local copy of this extension is created first.

⚠ tabular

Tabular `S` columns are rendered as simple `c` columns, and tabular `s` columns are not supported. These may be replaced by `c` columns with each cell contained in `\num` or `\si`.

for HTML output:

```
\RequirePackage{xcolor}% for \convertcolorspec

\LWR@ProvidesPackagePass{siunitx}[2018/05/17]


\AtBeginDocument{% in case textcomp was not loaded
\DeclareSIUnit\bohr{\textit{a}\textsubscript{0}}
\DeclareSIUnit\clight{\textit{c}\textsubscript{0}}
\DeclareSIUnit\elementarycharge{\textit{e}}
\DeclareSIUnit\electronmass{\textit{m}\textsubscript{e}}
\DeclareSIUnit\hartree{\textit{E}\textsubscript{h}}
\DeclareSIUnit\planckbar{\LWR@siunitx@textplanckbar}
}% AtBeginDocument
```

\@ensuredmath is not supported inside an \hbox, so it must temporarily be restored to its original. Similar for \mbox. SVG math is created explicitly when necessary, using \LWR@subsingledollar.

```
\ExplSyntaxOn
%
```

Modified to set set HTML \textcolor if not black:

```
\cs_undefine:N \__siunitx_print_aux:
\cs_new_protected:Npn \__siunitx_print_aux:
  {
    \text
      {
        \__siunitx_ensure_ltr:n
          {
            \color@begingroup
            \__siunitx_print_color:
            \__siunitx_font_shape:
            \__siunitx_font_weight:
            \use:c
              {
                @@_ \l__siunitx_print_type_tl _
                text \l__siunitx_font_family_tl :
              }
            \bool_if:NTF \l__siunitx_font_math_mode_bool
              { \__siunitx_print_math: }
              {
                \LWR@findcurrenttextcolor% lwarp
                \ifdefstring{\LWR@tempcolor}{000000}% lwarp
                {\__siunitx_print_text:}% lwarp
                {% lwarp
                    \LWR@textcurrentcolor{% lwarp
                        \__siunitx_print_text:
                    }% lwarp
                }% lwarp
              }
            \color@endgroup
          }
      }
  }


\cs_undefine:N \__siunitx_set_math_fam:n
\cs_new_protected:Npn \__siunitx_set_math_fam:n #1 {
  \int_new:c { c__siunitx_math #1 _int }
  \group_begin:% lwarp
    \LetLtxMacro\@ensuredmath\LWR@origensuredmath% lwarp
    \LetLtxMacro\mbox\LWR@print@mbox% lwarp
    \hbox_set:Nn \l__siunitx_tmp_box
      {
        \ensuremath
          {
            \use:c { math #1 }
```

```
              {
                \int_gset:cn { c__siunitx_math #1 _int } { \fam }
              }
          }
      }
  \group_end:% lwarp
}

\cs_undefine:N \__siunitx_combined_output:n
\cs_new_protected:Npn \__siunitx_combined_output:n #1 {
    \group_begin:% lwarp
    \LetLtxMacro\@ensuredmath\LWR@origensuredmath% lwarp
    \LetLtxMacro\mbox\LWR@print@mbox% lwarp
  \bool_if:NTF \l__siunitx_number_parse_bool
    {
      \tl_clear:N \l__siunitx_number_out_tl
      \bool_set_false:N \l__siunitx_number_compound_bool
      \__siunitx_number_output_parse:n {#1}
    }
    {
      \__siunitx_unit_output_pre_print:
```

For `parse-numbers=false`:

```
%       \__siunitx_print:nn { number } { \ensuremath {#1} }
        \LWR@subsingledollar{% lwarp
            \textbackslash( \LWR@HTMLsanitize{#1} \textbackslash)% lwarp
        }{siunitx}{%
            \__siunitx_print:nn { number } {%
                \LWR@origensuredmath{#1}%
            }%
        }% lwarp

      \__siunitx_unit_output_print:
    }
    \group_end:% lwarp
}
%
```

For quotients, the fraction code is replaced by the symbol code:

```
\cs_undefine:N \__siunitx_number_output_quotient_fraction:
\cs_new_protected:Npn \__siunitx_number_output_quotient_fraction: {
  \bool_set_true:N \l__siunitx_number_compound_bool
  \__siunitx_number_output_quotient_aux_i:
  \tl_set_eq:NN \l__siunitx_number_out_tl
    \l__siunitx_number_numerator_tl
  \tl_put_right:NV \l__siunitx_number_out_tl \l__siunitx_output_quotient_tl
  \tl_put_right:NV \l__siunitx_number_out_tl
    \l__siunitx_number_denominator_tl
  \__siunitx_number_output_single_aux:
}
```

For units, the fraction code is replaced by the symbol code:

```
\cs_undefine:N \__siunitx_unit_format_fraction_fraction:
\cs_new_protected:Npn \__siunitx_unit_format_fraction_fraction: {
  \__siunitx_unit_format_fraction_symbol_aux:
  \int_compare:nNnT { \l__siunitx_unit_denominator_int } > { 1 }
    {
      \bool_if:NT \l__siunitx_unit_denominator_bracket_bool
        {
       \tl_put_left:NV \l__siunitx_unit_denominator_tl \l__siunitx_bracket_open_tl
       \tl_put_right:NV \l__siunitx_unit_denominator_tl \l__siunitx_bracket_close_tl
        }
    }
  \tl_set_eq:NN \l__siunitx_unit_tl \l__siunitx_unit_numerator_tl
  \tl_put_right:NV \l__siunitx_unit_tl \l__siunitx_per_symbol_tl
  \tl_put_right:NV \l__siunitx_unit_tl \l__siunitx_unit_denominator_tl
}

\cs_undefine:N \__siunitx_angle_print_astronomy_aux:
\cs_new_protected:Npn \__siunitx_angle_print_astronomy_aux: {
  \prop_get:NnNT \l__siunitx_number_out_prop { mantissa-integer }
    \l__siunitx_tmpa_tl
    { \__siunitx_print:nV { number } \l__siunitx_tmpa_tl }
 \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}% lwarp
 {% lateximage
  \hbox_set:Nn \l__siunitx_angle_marker_box
    {
      \__siunitx_print:nn { number } { { \l__siunitx_output_decimal_tl } }
    }
  \hbox_set:Nn \l__siunitx_angle_unit_box
    {
      \__siunitx_print:nV { unit } \l__siunitx_unit_tl
      \skip_horizontal:n { -\scriptspace }
    }
  \__siunitx_angle_print_astronomy_aux:n { marker }
  \__siunitx_angle_print_astronomy_aux:n { unit }
  \hbox_set:Nn \l__siunitx_angle_marker_box
    {
      \box_use:N \l__siunitx_angle_marker_box
      \box_use:N \l__siunitx_angle_unit_box
    }
  \dim_compare:nNnTF
    { \l__siunitx_angle_marker_dim } > { \l__siunitx_angle_unit_dim }
    { \__siunitx_angle_print_astronomy_marker: }
    { \__siunitx_angle_print_astronomy_unit: }
 }% lateximage
 {% not a lateximage
      \__siunitx_print:nV { unit } \l__siunitx_unit_tl
      \__siunitx_print:nn { number } { { \l__siunitx_output_decimal_tl } }
 }% not a lateximage
  \prop_get:NnNT \l__siunitx_number_out_prop { mantissa-decimal }
    \l__siunitx_tmpa_tl
    { \__siunitx_print:nV { number } \l__siunitx_tmpa_tl }
}

\RenewDocumentCommand \num { o m } {
  \leavevmode
```

```
  \group_begin:% lwarp
    \LetLtxMacro\@ensuredmath\LWR@origensuredmath% lwarp
    \LetLtxMacro\mbox\LWR@print@mbox% lwarp
    \bool_set_false:N \l__siunitx_font_set_bool
    \IfNoValueF {#1}
      { \keys_set:nn { siunitx } {#1} }
    \__siunitx_number_output:n {#2}
  \group_end:% lwarp
}

\RenewDocumentCommand \numrange { o m m } {
  \leavevmode
  \group_begin:% lwarp
    \LetLtxMacro\@ensuredmath\LWR@origensuredmath% lwarp
    \LetLtxMacro\mbox\LWR@print@mbox% lwarp
    \bool_set_false:N \l__siunitx_font_set_bool
    \IfNoValueF {#1}
      { \keys_set:nn { siunitx } {#1} }
    \__siunitx_range_numbers:nn {#2} {#3}
  \group_end:% lwarp
}

\RenewDocumentCommand \ang { o > { \SplitArgument { 2 } { ; } } m } {
  \group_begin:% lwarp
    \LetLtxMacro\@ensuredmath\LWR@origensuredmath% lwarp
    \LetLtxMacro\mbox\LWR@print@mbox% lwarp
    \IfNoValueF {#1}
      { \keys_set:nn { siunitx } {#1} }
    \__siunitx_angle_output:nnn #2
  \group_end:% lwarp
}

\RenewDocumentCommand \si { o m } {
  \leavevmode
  \group_begin:% lwarp
    \LetLtxMacro\@ensuredmath\LWR@origensuredmath% lwarp
    \LetLtxMacro\mbox\LWR@print@mbox% lwarp
    \bool_set_false:N \l__siunitx_font_set_bool
    \IfNoValueTF {#1}
      { \__siunitx_unit_output:nn {#2} { } }
      {
        \keys_set:nn { siunitx } {#1}
        \__siunitx_unit_output:nn {#2} {#1}
      }
  \group_end:% lwarp
}


\RenewDocumentCommand{\SIrange}{o m m m}
{%
  \leavevmode
  \group_begin:% lwarp
    \LetLtxMacro\@ensuredmath\LWR@origensuredmath% lwarp
    \LetLtxMacro\mbox\LWR@print@mbox% lwarp
    \bool_set_false:N \l__siunitx_font_set_bool
```

```
    \IfNoValueTF {#1}
      { \__siunitx_range_unit:nnnn {#4} { } {#2} {#3} }
      {
        \keys_set:nn { siunitx } {#1}
        \__siunitx_range_unit:nnnn {#4} {#1} {#2} {#3}
      }
  \group_end:% lwarp
}

\ExplSyntaxOff
```

File 309 **lwarp-soul.sty**

§ 406 Package

soul

(*Emulates or patches code by* Melchior FRANZ.)

Pkg soul Emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{soul}[2003/11/17]
\RequirePackage{xcolor}% for \convertcolorspec
```

Storage for the colors to use:

```
\newcommand*{\LWR@soululcolor}{}

\newcommand*{\LWR@soulstcolor}{}

% \definecolor{LWR@soulhlcolordefault}{HTML}{F8E800}
% \newcommand*{\LWR@soulhlcolor}{LWR@soulhlcolordefault}
\newcommand*{\LWR@soulhlcolor}{}
```

\so {⟨*text*⟩}

Basic markup with CSS:

```
\newcommand{\so}[1]{%
\InlineClass(letter-spacing:.2ex){letterspacing}{#1}%
}
```

\caps {⟨*text*⟩}

```
\newcommand{\caps}[1]{%
    \InlineClass%
        (font-variant:small-caps;letter-spacing:.1ex)%
        {capsspacing}{#1}%
}
```

\LWR@soulcolor {⟨*text*⟩} {⟨*color*⟩} {⟨*class*⟩} {⟨*colorstyle*⟩} {⟨*FormatWPstyle*⟩}

Add colors if not empty:

```
\newcommand{\LWR@soulcolor}[5]{%
\ifcsempty{#2}%
{%
    \InlineClass(#5){#3}{#1}%
}%
{%
    \convertcolorspec{named}{\@nameuse{#2}}{HTML}\LWR@tempcolor%
    \LWR@htmlspanclass[#5;#4:\LWR@origpound\LWR@tempcolor]{#3}{#1}%
}%
}

\newcommand{\ul}[1]{%
\LWR@soulcolor{#1}{LWR@soululcolor}{uline}{text-decoration-color}%
    {text-decoration:underline; text-decoration-skip: auto;}%
}

\newcommand{\st}[1]{
\LWR@soulcolor{#1}{LWR@soulstcolor}{sout}{text-decoration-color}%
    {text-decoration:line-through}%
}

\newcommand{\hl}[1]{
\LWR@soulcolor{#1}{LWR@soulhlcolor}{highlight}{background-color}%
    {background:\LWR@origpound{}F8E800}
}
```

Nullified:

```
\newcommand*{\soulaccent}[1]{}
\newcommand*{\soulregister}[2]{}
\newcommand{\sloppyword}[1]{#1}
\newcommand*{\sodef}[5]{\DeclareRobustCommand*#1[1]{\so{##1}}}
\newcommand*{\resetso}{}
\newcommand*{\capsdef}[5]{}
\newcommand*{\capsreset}{}
\newcommand*{\capssave}[1]{}
\newcommand*{\capsselect}[1]{}
\newcommand*{\setul}[2]{}
\newcommand*{\resetul}{}
\newcommand*{\setuldepth}[1]{}
\newcommand*{\setuloverlap}[1]{}
```

Set colors:

```
\newcommand*{\setulcolor}[1]{\renewcommand{\LWR@soululcolor}{#1}}
\newcommand*{\setstcolor}[1]{\renewcommand{\LWR@soulstcolor}{#1}}
\newcommand*{\sethlcolor}[1]{\renewcommand{\LWR@soulhlcolor}{#1}}
```

Long versions of the user-level macros:

```
\let\textso\so
\let\textul\ul
\let\texthl\hl
\let\textcaps\caps
```

File 310 **lwarp-soulpos.sty**

§ 407 Package soulpos

(*Emulates or patches code by* Javier Bezos.)

Pkg soulpos soulpos is emulated.

for HTML output:

```
\RequirePackage{soul}
\RequirePackage{soulutf8}
\LWR@ProvidesPackageDrop{soulpos}[2012/02/25]

\NewDocumentCommand{\ulposdef}{m o m}{}

\newdimen\ulwidth

\newcommand\ifulstarttype[1]{%
\expandafter\@secondoftwo%
}

\newcommand\ifulendtype[1]{%
\expandafter\@secondoftwo%
}

\newcommand{\ulstarttype}{0}
\newcommand{\ulendtype}{0}
\newcommand\ulpostolerance{0}%
```

File 311 **lwarp-soulutf8.sty**

§ 408 Package soulutf8

Pkg soulutf8 soulutf8 is emulated.

lwarp's HTML output naturally supports UTF-8 encoding.

for HTML output:

```
\LWR@ProvidesPackageDrop{soulutf8}[2016/05/16]
```

File 312 **lwarp-splitidx.sty**

§ 409 Package splitidx

(*Emulates or patches code by* Markus Kohm.)

Pkg splitidx 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 494.17 in the lwarp-patch-memoir package for the `\@@wrspindexhyp` macro as an example.

for HTML output:

```
\LWR@ProvidesPackagePass{splitidx}[2016/02/18]

\catcode`\_=12%
\xpatchcmd{\newindex}
    {\jobname-#2.idx}
    {\jobname-#2_html.idx}
    {}
    {\LWR@patcherror{splitidx}{@newindex}}
\catcode`\_=8%
```

Patched to use lwarp's automatic indexing counter instead of `\thepage`:

```
\renewcommand*{\@wrsindex}[2][]{%
  \ifx\relax#1\relax
    \if@splitidx
      \@wrsindex[idx]{#2}%
    \else
      \def\@tempa{#2}%
      \if@verbindex\@onelevel@sanitize\@tempa\fi
      \@wrindex{\@tempa}%
    \fi
  \else
    \def\@tempa{#2}%
    \csname index@#1@hook\endcsname
%     \expandafter\ifx\csname @@wrsindex\endcsname\relax
    \addtocounter{LWR@autoindex}{1}%                    lwarp
    \LWR@new@label{LWRindex-\arabic{LWR@autoindex}}%    lwarp
%       \@@@wrsindex{#1}{{\@tempa}{\thepage}}%
      \@@@wrsindex{#1}{{\@tempa}{\arabic{LWR@autoindex}}}%
%     \else
%       \def\@tempb{\@@wrsindex{#1}}%
%       \expandafter\@tempb\@tempa||\\%
%     \fi
    \endgroup
    \@esphack
  \fi
}
```

lwarp defines sectioning commands with xparse, so the below patches are done as temporary redefinitions instead of being \let.

```
\xpatchcmd{\printsubindex}
    {\let\section\subsection}
    {\renewcommand*{\section}{\subsection}}
    {}
    {\LWR@patcherror{splitidx}{printsubindex-section}}

\xpatchcmd{\printsubindex}
    {\let\chapter\section}
    {\renewcommand*{\chapter}{\section}}
    {}
    {\LWR@patcherror{splitidx}{printsubindex-chapter}}

\xpatchcmd{\printsubindex}
    {\let\@makechapterhead\section}
    {\def\@makechapterhead{\section}}
    {}
    {\LWR@patcherror{splitidx}{printsubindex-chapter}}
```

File 313 **lwarp-srcltx.sty**

§ 410 Package srcltx

Pkg srcltx srcltx is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{srcltx}[2006/11/12]

\newif\ifSRCOK \SRCOKfalse
\newcommand*\srcIncludeHook[1]{}
\newcommand*\srcInputHook[1]{}
\newcommand*\MainFile{}
\def\MainFile{\jobname.tex}
\newcommand*\CurrentInput{}
\gdef\CurrentInput{\MainFile}
\newcommand\Input{}
\let\Input\input
```

File 314 **lwarp-srctex.sty**

§ 411 Package srctex

Pkg srctex srctex is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{srctex}[2006/11/12]
\LWR@origRequirePackage{lwarp-srcltx}
```

File 315 **lwarp-stabular.sty**

§ 412 Package stabular

(*Emulates or patches code by* Sigitas Tolušis.)

Pkg `stabular` stabular is emulated.

for HTML output:
```
\LWR@ProvidesPackageDrop{stabular}[2014/03/20]
```

Env `stabular` `[`⟨*vpos*⟩`]` `{`⟨*colspec*⟩`}`

```
\newenvironment{stabular}[2][c]
{
\begin{tabular}[#1]{#2}
\renewcommand{\noalign}[1]{}
}
{\end{tabular}}
```

Env `stabular` `{`⟨*width*⟩`}` `[`⟨*vpos*⟩`]` `{`⟨*colspec*⟩`}`

```
\NewDocumentEnvironment{stabular*}{m o m}
{
\begin{tabular}[#2]{#3}
\renewcommand{\noalign}[1]{}
}
{\end{tabular}}
```

File 316 **lwarp-stfloats.sty**

§ 413 Package stfloats

Pkg `stfloats` stfloats is ignored.

for HTML output:
```
\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:

```
\providecommand*{\fnbelowfloat}{}
\providecommand*{\fnunderfloat}{}
\providecommand*{\setbaselinefloat}{}
\providecommand*{\setbaselinefixed}{}
```

Nullified for HTML:

```
\renewcommand*{\fnbelowfloat}{}
\renewcommand*{\fnunderfloat}{}
\renewcommand*{\setbaselinefloat}{}
\renewcommand*{\setbaselinefixed}{}
```

File 317 **lwarp-struktex.sty**

§ 414 Package struktex

(*Emulates or patches code by* Jobst Hoffmann.)

Pkg struktex struktex is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{struktex}[2018/06/04]

\BeforeBeginEnvironment{struktogramm}{\begin{lateximage}[(struktogramm)]}
\AfterEndEnvironment{struktogramm}{\end{lateximage}}

\newenvironment{LWR@HTML@centernss}{\begin{center}}{\end{center}}
\LWR@formattedenv{centernss}

\newcommand{\LWR@HTML@CenterNssFile}[1]{%
    \begin{center}
    \input{#1.nss}
    \end{center}
}
\LWR@formatted{CenterNssFile}

\newcommand{\LWR@HTML@centernssfile}{\LWR@HTML@CenterNssFile}
\LWR@formatted{centernssfile}
```

File 318 **lwarp-subcaption.sty**

§ 415 Package subcaption

(*Emulates or patches code by* Axel Sommerfeldt.)

Pkg subcaption subcaption is patched for use by lwarp.

for HTML output:

```
\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.

```
\ifdef{\subcaption@minipage}{
    \xpatchcmd{\subcaption@minipage}
        {\minipage}
        {\minipagefullwidth\minipage}
```

```
        {}
        {\LWR@patcherror{subcaption}{subcaption@minipage}}
}{}
```

Likewise for a \subcaptionbox:

```
\ifdef{\subcaptionbox}{
    \xpretocmd{\subcaptionbox}
        {\minipagefullwidth}
        {}
        {\LWR@patcherror{subcaption}{subcaptionbox}}
}{}
```

File 319 **lwarp-subfig.sty**

§ 416 Package

subfig

(*Emulates or patches code by* Steven Douglas Cochran.)

Pkg subfig subfig is supported and patched by lwarp.

⚠ 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.

In the document source, use \hfill and \hspace* subfig>inline 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.

for HTML output: Accept all options for lwarp-subfig:

```
\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.

```
\long\def\sf@@@subfloat#1[#2][#3]#4{%
\begin{minipage}{\linewidth}% lwarp

\IfValueTF{#2}{%
    \LWR@setlatestname{#2}%
}{%
    \IfValueTF{#3}{%
        \LWR@setlatestname{#3}%
    }{}%
}%
\LWR@stoppars% lwarp
    \@ifundefined{FBsc@max}{}%
        {\FB@readaux{\let\FBsuboheight\relax}}%
    \@tempcnta=\@ne
    \if@minipage
      \@tempcnta=\z@
```

```
    \else\ifdim \lastskip=\z@ \else
      \@tempcnta=\tw@
    \fi\fi
    \ifmaincaptiontop
      \sf@top=\sf@nearskip
      \sf@bottom=\sf@farskip
    \else
      \sf@top=\sf@farskip
      \sf@bottom=\sf@nearskip
    \fi
    \leavevmode

%     \setbox\@tempboxa \hbox{#4}%
%     \@tempdima=\wd\@tempboxa
%     \@ifundefined{FBsc@max}{}%
%         {\global\advance\Xhsize-\wd\@tempboxa
%          \dimen@=\ht\@tempboxa
%          \advance\dimen@\dp\@tempboxa
%          \ifdim\dimen@>\FBso@max
%            \global\FBso@max\dimen@
%          \fi}%
```

Do not use boxes, which interfere with `lateximages`:

```
%     \vtop%
    \bgroup
%       \vbox%
      \bgroup
        \ifcase\@tempcnta
          \@minipagefalse
        \or
%           \vskip\sf@top
        \or
          \ifdim \lastskip=\z@ \else
%             \@tempskipb\sf@top\relax\@xaddvskip
          \fi
        \fi
        \sf@ifpositiontop{%
          \ifx \@empty#3\relax \else
            \sf@subcaption{#1}{#2}{#3}%
%             \vskip\sf@capskip
%             \vskip\sf@captopadj
          \fi\egroup
%           \hrule width0pt height0pt depth0pt
            \LWR@startpars% lwarp
%  \box\@tempboxa
            #4
            \LWR@stoppars% lwarp
        }{%
        \LWR@startpars% lwarp
        \@ifundefined{FBsc@max}%
            {
%  \box\@tempboxa
                #4
            }%
            {\ifx\FBsuboheight\relax
```

```
%                \box\@tempboxa
                 #4
              \else
%                \vbox to \FBsuboheight{\FBafil\box\@tempboxa\FBbfil}%
                 #4
              \fi}%
         \LWR@stoppars% lwarp
           \egroup
           \ifx \@empty#3\relax \else
%              \vskip\sf@capskip
%              \hrule width0pt height0pt depth0pt
             \sf@subcaption{#1}{#2}{#3}%
          \fi
         }%
%        \vskip\sf@bottom
    \egroup
    \@ifundefined{FBsc@max}{}%
        {\addtocounter{FRobj}{-1}%
         \ifnum\c@FRobj=0\else
           \subfloatrowsep
         \fi}%
    \ifmaincaptiontop\else
      \global\advance\@nameuse{c@\@captype}\m@ne
    \fi
\end{minipage}% lwarp
\LWR@startpars% lwarp
  \endgroup\ignorespaces%
}%
```

\sf@subcaption {⟨*1 type*⟩} {⟨*2 LOF entry*⟩} {⟨*3 caption*⟩}

```
\long\def\sf@subcaption#1#2#3{%
\LWR@stoppars% lwarp
  \ifx \relax#2\relax \else
    \bgroup
      \let\label=\@gobble
      \let\protect=\string
      \def\@subcaplabel{%
        \caption@lstfmt{\@nameuse{p@#1}}{\@nameuse{the#1}}}%
      \sf@updatecaptionlist{#1}{#2}{\the\value{\@captype}}{\the\value{#1}}%
    \egroup
  \fi
  \bgroup
    \ifx \relax#3\relax
      \let\captionlabelsep=\relax
    \fi
%     \setbox0\vbox{%
%        \hb@xt@\the\@tempdima{%
%
% %          \hss
% %         \parbox[t]{\the\@tempdima}{%
%           \caption@make
%               {\@nameuse{sub\@captype name}}%
%               {\@nameuse{thesub\@captype}}%
%               {#3}
```

```
% %  }%
% %            \hss
%   }
%  }%
    \@ifundefined{FBsc@max}%
%         {\box0}%
        {
%  \parbox[t]{\the\@tempdima}{%
\LWR@traceinfo{sfsubcap B1}% lwarp
            \LWR@figcaption% lwarp
            \caption@make
                {\@nameuse{sub\@captype name}}%
                {\@nameuse{thesub\@captype}}%
                {\LWR@isolate{#3}}%

            \endLWR@figcaption% lwarp
\LWR@traceinfo{sfsubcap B2}% lwarp
%  }%
        }%
        {\dimen@\ht0%
         \advance\dimen@\dp0%
         \ifdim\dimen@>\FBsc@max
           \global\FBsc@max\dimen@
         \fi
         \FB@readaux{\let\FBsubcheight\relax}%
         \ifx\FBsubcheight\relax
           \def\next{
%  \parbox[t]{\the\@tempdima}
            }%
         \else
           \def\next{
%  \parbox[t][\FBsubcheight][t]{\the\@tempdima}
            }%
         \fi
%          \vbox{%
%            \hb@xt@\the\@tempdima{%

%              \hss
%              \next{%
\LWR@traceinfo{sfsubcap C1}% lwarp
                \caption@make
                    {\@nameuse{sub\@captype name}}%
                    {\@nameuse{thesub\@captype}}%
                    {#3}
\LWR@traceinfo{sfsubcap C1}% lwarp
%  }%
%              \hss

%  }
%           }
        }%
  \egroup
\LWR@startpars% lwarp
}
```

\subfloat@label Patches for \sf@sub@label:

```
\def\subfloat@label{%
\LWR@ensuredoingapar% lwarp
  \@ifnextchar(%   %) match left parenthesis
    {\sf@sub@label}
    {\sf@sub@label(Sub\@captype\space
                   \@ifundefined{thechapter}{}{\@nameuse{thechapter}\space}%
                   \@nameuse{p@sub\@captype}%
                   \@nameuse{thesub\@captype}.)}}
```

Patches for \subref.

\sf@subref {⟨*label*⟩}

The unstarred version uses a \ref link whose printed text comes from the sub@<label>:

```
\renewcommand{\sf@subref}[1]{%
\LWR@subnewref{#1}{sub@#1}%
}
```

\sf@@subref {⟨*label*⟩}

The starred version uses the printed sub@<label> which is stored as if it were a page number:

```
\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*⟩}

```
\LetLtxMacro\LWR@orig@newsubfloat\@newsubfloat

\def\@newsubfloat[#1]#2{%
\LWR@orig@newsubfloat[#1]{#2}%
\renewcommand{\l@sub#2}[2]{\hypertocfloat{2}{sub#2}{\ext@sub#2}{##1}{##2}}%
}
```

Pre-defined for figures and tables:

\l@subfigure {⟨*text*⟩} {⟨*pagenum*⟩}

```
\renewcommand{\l@subfigure}[2]{\hypertocfloat{2}{subfigure}{lof}{#1}{#2}}
```

\l@subtable {⟨*text*⟩} {⟨*pagenum*⟩}

```
\renewcommand{\l@subtable}[2]{\hypertocfloat{2}{subtable}{lot}{#1}{#2}}
```

File 320 **lwarp-subfigure.sty**

§ 417 Package

subfigure

Pkg subfigure subfigure is emulated by subfig.

for HTML output:

```
\LWR@ProvidesPackageDrop{subfigure}[2002/03/15]
\RequirePackage{subfig}

\LetLtxMacro\subfigure\subfloat
\LetLtxMacro\subtable\subfloat
\LetLtxMacro\Subref\subref
\@ifundefined{figuretopcaptrue}{\newif\iffiguretopcap}{}
\newif\ifsubfiguretopcap
\newif\ifsubcaphang
\newif\ifsubcapcenter
\newif\ifsubcapcenterlast
\newif\ifsubcapnooneline
\newif\ifsubcapraggedright
\newskip\subfigtopskip
\newskip\subfigcapskip
\newdimen\subfigcaptopadj
\newskip\subfigbottomskip
\newdimen\subfigcapmargin
\newskip\subfiglabelskip
\newcommand*{\subcapsize}{}
\newcommand*{\subcaplabelfont}{}
\newcommand*{\subcapfont}{}
```

File 321 **lwarp-supertabular.sty**

§ 418 Package

supertabular

(*Emulates or patches code by* Johannes Braams, Theo Jurriens.)

Pkg supertabular supertabular is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{supertabular}[2004/02/20]
```

⚠ Misplaced alignment tab character &

For `\tablefirsthead`, etc., enclose them as follows:

```
\StartDefiningTabulars
\tablefirsthead
. . .
\StopDefiningTabulars
```

See section 9.10.

⚠ lateximage supertabular and xtab are not supported inside a `lateximage`.

```
\newcommand{\LWRST@firsthead}{}

\newcommand{\tablefirsthead}[1]{%
    \long\gdef\LWRST@firsthead{#1}%
}

\newcommand{\tablehead}[1]{}
\newcommand{\tabletail}[1]{}

```

```
\newcommand{\LWRST@lasttail}{}

\newcommand{\tablelasttail}[1]{%
    \long\gdef\LWRST@lasttail{#1}%
}


\newcommand{\tablecaption}[2][]{%
    \long\gdef\LWRST@caption{%
        \ifblank{#1}%
            {\caption{#2}}%
            {\caption[#1]{#2}}%
    }%
}

\let\topcaption\tablecaption
\let\bottomcaption\tablecaption


\newcommand*{\LWRST@caption}{}

\newcommand*{\shrinkheight}[1]{}

\NewDocumentEnvironment{supertabular}{s o m}
{%
\LWR@traceinfo{supertabular}%
\begin{table}%
\LWRST@caption%
\begin{tabular}{#3}%
\TabularMacro\ifdefvoid{\LWRST@firsthead}%
{\LWR@getmynexttoken}%
{\expandafter\LWR@getmynexttoken\LWRST@firsthead}%
}%
{%
\ifdefvoid{\LWRST@lasttail}%
{}%
{%
\TabularMacro\ResumeTabular%
\LWRST@lasttail%
}%
\end{tabular}%
\end{table}%
\LWR@traceinfo{supertabular done}%
}

\NewDocumentEnvironment{mpsupertabular}{s o m}
{\minipage{\linewidth}\supertabular{#3}}
{\endsupertabular\endminipage}
```

File 322 **lwarp-syntonly.sty**

§ 419 Package syntonly

(*Emulates or patches code by* Frank Mittelbach, Rainer Schöpf.)

Pkg syntonly Emulated.

for HTML output: Discard all options for lwarp-syntonly:

```
\LWR@ProvidesPackageDrop{syntonly}[2017/06/30]

\newif\ifsyntax@
\syntax@false

\newcommand*{\syntaxonly}{}

\@onlypreamble\syntaxonly
```

File 323 **lwarp-tabls.sty**

§ 420 Package tabls

(*Emulates or patches code by* Donald Arseneau.)

Pkg tabls tabls is emulated. `\LWR@hline` is used to handle the optional argument when tabls is loaded.

for HTML output:

```
\LWR@ProvidesPackageDrop{tabls}

\newdimen\tablinesep
\newdimen\arraylinesep
\newdimen\extrarulesep
```

File 324 **lwarp-tabularx.sty**

§ 421 Package tabularx

(*Emulates or patches code by* David Carlisle.)

Pkg tabularx tabularx is emulated by lwarp.

for HTML output: Discard all options for lwarp-tabularx:

```
\LWR@ProvidesPackageDrop{tabularx}[2016/02/03]

\DeclareDocumentEnvironment{tabularx}{m o m}
{\tabular{#3}}
{\endtabular}

\DeclareDocumentEnvironment{tabularx*}{m o m}
{\tabular{#3}}
{\endtabular}
```

File 325 **lwarp-tabulary.sty**

§ 422 Package tabulary

(*Emulates or patches code by* David Carlisle.)

Pkg tabulary 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.

```
\LWR@ProvidesPackageDrop{tabulary}[2014/06/11]

\NewDocumentEnvironment{tabulary}{m o m}
{\tabular{#3}}
{\endtabular}

\NewDocumentEnvironment{tabulary*}{m o m}
{\tabular{#3}}
{\endtabular}

\newdimen\tymin
\newdimen\tymax
\def\tyformat{}
```

File 326 **lwarp-tascmac.sty**

§ 423 Package tascmac

Pkg tascmac tascmac is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{tascmac}[2018/03/09]

\newenvironment*{boxnote}
    {
        \BlockClass[
            padding: .5ex ;
            border: 1px solid black ;
            border-top: 1px dashed black ;
        ]{boxnote}
    }
    {\endBlockClass}

\newenvironment*{screen}[1][]
    {
        \BlockClass[
            padding: .5ex ;
```

```
            border: 1px solid gray ;
            border-radius: 8pt
        ]{boxnote}
    }
    {\endBlockClass}

\newenvironment*{itembox}[2][]
    {
        \BlockClass[
            padding: .5ex ;
            border: 1px solid gray ;
            border-radius: 8pt
        ]{boxnote}
        \InlineClass{itemboxtitle}{#2}\par
    }
    {\endBlockClass}

\newenvironment*{shadebox}
    {
        \BlockClass[
            padding: .5ex ;
            border: 1px solid black ;
            box-shadow: 3px 3px 3px \#808080 ;
        ]{boxnote}
    }
    {\endBlockClass}

\newcommand*{\mask}[2]{%
    \InlineClass[background: lightgray]{mask}{#1}%
}

\newcommand*{\maskbox}[5]{%
    \InlineClass[background: lightgray]{mask}{#5}%
}

\newcommand*{\Maskbox}[6]{%
    \InlineClass[
        background: lightgray ;
        border: #5 solid black
    ]{mask}{#6}%
}

\newcommand*{\keytop}[2][]{%
    \InlineClass[%
        padding: .2ex ;
        border: 1px solid black ;
        border-radius: .7ex ;
    ]{keytop}{#2}%
}

\def\yen{\HTMLunicode{00A5}}

\def\return{\HTMLunicode{23CE}}

\def\Return{\HTMLunicode{23CE}}
```

```
71
72 \def\ascii{ASCII Corporation}
73
74 \def\Ascii{ASCII Corporation}
75
76 \def\ASCII{ASCII Corporation}
```

File 327 **lwarp-textarea.sty**

§ 424 Package

textarea

(*Emulates or patches code by* Alexander I. Rozhenko.)

Pkg textarea 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 328 **lwarp-textcomp.sty**

§ 425 Package

textcomp

(*Emulates or patches code by* Frank Mittelbach, Robin Fairbairns, Werner Lemberg.)

Pkg textcomp textcomp is patched for use by lwarp.

§ 425.1 Limitations

Some textcomp symbols do not have Unicode equivalents, and thus are not supported.

⚠ missing symbols Many textcomp symbols are not supported by many fonts. In the css try referencing fonts which are more complete, but expect to see gaps in coverage.

§ 425.2 Package loading

for HTML output:

```
1 \LWR@ProvidesPackagePass{textcomp}[2017/04/05]
```

§ 425.3 HTML symbols

For HTML, use HTML entities or direct Unicode, depending on the engine.

`\AtBeginDocument` improves support for LuaLaTeX and XeLaTeX.

§ 425.3.1 pdfLaTeX symbols

```
\AtBeginDocument{
\ifPDFTeX% pdflatex or dvi latex
\newcommand*{\LWR@HTML@textdegree}{\HTMLentity{deg}}
\newcommand*{\LWR@HTML@textcelsius}{\HTMLunicode{2103}}
\newcommand*{\LWR@HTML@textohm}{\HTMLunicode{2126}}
\newcommand*{\LWR@HTML@textmu}{\HTMLunicode{00B5}}
\newcommand*{\LWR@HTML@textlquill}{\HTMLunicode{2045}}
\newcommand*{\LWR@HTML@textrquill}{\HTMLunicode{2046}}
\newcommand*{\LWR@HTML@textcircledP}{\HTMLunicode{2117}}
\newcommand*{\LWR@HTML@texttwelveudash}{\HTMLunicode{2014}}% emdash
\newcommand*{\LWR@HTML@textthreequartersemdash}{\HTMLunicode{2014}}% emdash
\newcommand*{\LWR@HTML@textmho}{\HTMLunicode{2127}}
\newcommand*{\LWR@HTML@textnaira}{\HTMLunicode{20A6}}
\newcommand*{\LWR@HTML@textpeso}{\HTMLunicode{20B1}}
\newcommand*{\LWR@HTML@textrecipe}{\HTMLunicode{211E}}
\newcommand*{\LWR@HTML@textinterrobangdown}{\HTMLunicode{2E18}}
\newcommand*{\LWR@HTML@textperthousand}{\HTMLunicode{2030}}
\newcommand*{\LWR@HTML@textpertenthousand}{\HTMLunicode{2031}}
\newcommand*{\LWR@HTML@textbaht}{\HTMLunicode{0E3F}}
\newcommand*{\LWR@HTML@textdiscount}{\%}
\newcommand*{\LWR@HTML@textservicemark}{\HTMLunicode{2120}}
\else
```

§ 425.3.2 XeLaTeX 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.

```
\newcommand*{\LWR@HTML@textdegree}{°}
\newcommand*{\LWR@HTML@textcelsius}{}
\newcommand*{\LWR@HTML@textohm}{}
\newcommand*{\LWR@HTML@textmu}{µ}
\newcommand*{\LWR@HTML@textlquill}{}
\newcommand*{\LWR@HTML@textrquill}{}
\newcommand*{\LWR@HTML@textcircledP}{}
\newcommand*{\LWR@HTML@texttwelveudash}{---}% emdash
\newcommand*{\LWR@HTML@textthreequartersemdash}{---}% emdash
\newcommand*{\LWR@HTML@textmho}{}
\newcommand*{\LWR@HTML@textnaira}{}
\newcommand*{\LWR@HTML@textpeso}{}
\newcommand*{\LWR@HTML@textrecipe}{}
\newcommand*{\LWR@HTML@textinterrobangdown}{}
\newcommand*{\LWR@HTML@textperthousand}{%}
\newcommand*{\LWR@HTML@textpertenthousand}{%}
\newcommand*{\LWR@HTML@textbaht}{}
```

```
\newcommand*{\LWR@HTML@textdiscount}{\%}
\newcommand*{\LWR@HTML@textservicemark}{}
\fi

\LWR@formatted{textdegree}
\LWR@formatted{textcelsius}
\LWR@formatted{textohm}
\LWR@formatted{textmu}
\LWR@formatted{textlquill}
\LWR@formatted{textrquill}
\LWR@formatted{textcircledP}
\LWR@formatted{texttwelveudash}
\LWR@formatted{textthreequartersemdash}
\LWR@formatted{textmho}
\LWR@formatted{textnaira}
\LWR@formatted{textpeso}
\LWR@formatted{textrecipe}
\LWR@formatted{textinterrobangdown}
\LWR@formatted{textperthousand}
\LWR@formatted{textpertenthousand}
\LWR@formatted{textbaht}
\LWR@formatted{textdiscount}
\LWR@formatted{textservicemark}
```

§ 425.4 HTML diacritics

For HTML, Unicode diacritical marks are used:

```
\newcommand*{\LWR@HTML@capitalcedilla}[1]{#1\HTMLunicode{0327}}
\newcommand*{\LWR@HTML@capitalogonek}[1]{#1\HTMLunicode{0328}}
\newcommand*{\LWR@HTML@capitalgrave}[1]{#1\HTMLunicode{0300}}
\newcommand*{\LWR@HTML@capitalacute}[1]{#1\HTMLunicode{0301}}
\newcommand*{\LWR@HTML@capitalcircumflex}[1]{#1\HTMLunicode{0302}}
\newcommand*{\LWR@HTML@capitaltilde}[1]{#1\HTMLunicode{0303}}
\newcommand*{\LWR@HTML@capitaldieresis}[1]{#1\HTMLunicode{0308}}
\newcommand*{\LWR@HTML@capitalhungarumlaut}[1]{#1\HTMLunicode{30B}}
\newcommand*{\LWR@HTML@capitalring}[1]{#1\HTMLunicode{30A}}
\newcommand*{\LWR@HTML@capitalcaron}[1]{#1\HTMLunicode{30C}}
\newcommand*{\LWR@HTML@capitalbreve}[1]{#1\HTMLunicode{306}}
\newcommand*{\LWR@HTML@capitalmacron}[1]{#1\HTMLunicode{304}}
\newcommand*{\LWR@HTML@capitaldotaccent}[1]{#1\HTMLunicode{307}}
```

`\textcircled` becomes a span with a rounded border. `\providecommand` is used to avoid conflict with xunicode.

```
\providecommand*{\LWR@HTML@textcircled}[1]{%
\InlineClass[border: 1px solid \LWR@currenttextcolor]{textcircled}{#1}%
}

\LWR@formatted{capitalcedilla}
\LWR@formatted{capitalogonek}
\LWR@formatted{capitalgrave}
\LWR@formatted{capitalacute}
\LWR@formatted{capitalcircumflex}
\LWR@formatted{capitaltilde}
```

```
\LWR@formatted{capitaldieresis}
\LWR@formatted{capitalhungarumlaut}
\LWR@formatted{capitalring}
\LWR@formatted{capitalcaron}
\LWR@formatted{capitalbreve}
\LWR@formatted{capitalmacron}
\LWR@formatted{capitaldotaccent}
\LWR@formatted{textcircled}
```

Nullify textcomp macros when generating filenames:

```
\FilenameNullify{%
    \renewcommand*{\textdegree}{}%
    \renewcommand*{\textcelsius}{}%
    \renewcommand*{\textohm}{}%
    \renewcommand*{\textmu}{}%
    \renewcommand*{\textlquill}{}%
    \renewcommand*{\textrquill}{}%
    \renewcommand*{\textcircledP}{}%
    \renewcommand*{\texttwelveudash}{}%
    \renewcommand*{\textthreequartersemdash}{}%
    \renewcommand*{\textmho}{}%
    \renewcommand*{\textnaira}{}%
    \renewcommand*{\textpeso}{}%
    \renewcommand*{\textrecipe}{}%
    \renewcommand*{\textinterrobangdown}{}%
    \renewcommand*{\textperthousand}{}%
    \renewcommand*{\textpertenthousand}{}%
    \renewcommand*{\textbaht}{}%
    \renewcommand*{\textdiscount}{}%
    \renewcommand*{\textservicemark}{}%
    \renewcommand*{\textcircled}[1]{#1}%
    \renewcommand*{\capitalcedilla}[1]{#1}%
    \renewcommand*{\capitalogonek}[1]{#1}%
    \renewcommand*{\capitalgrave}[1]{#1}%
    \renewcommand*{\capitalacute}[1]{#1}%
    \renewcommand*{\capitalcircumflex}[1]{#1}%
    \renewcommand*{\capitaltilde}[1]{#1}%
    \renewcommand*{\capitaldieresis}[1]{#1}%
    \renewcommand*{\capitalhungarumlaut}[1]{#1}%
    \renewcommand*{\capitalring}[1]{#1}%
    \renewcommand*{\capitalcaron}[1]{#1}%
    \renewcommand*{\capitalbreve}[1]{#1}%
    \renewcommand*{\capitalmacron}[1]{#1}%
    \renewcommand*{\capitaldotaccent}[1]{#1}%
}% FilenameNullify

}% AtBeginDocument
```

File 329 **lwarp-textfit.sty**

§ 426 Package textfit

Pkg textfit textfit is emulated.

Text is placed into a of class `textfit`. Sizes are approximated, and also limited by browser min/max font-size settings.

for HTML output:

```
\LWR@ProvidesPackageDrop{textfit}[1994/04/15]

\newsavebox{\LWR@textfitbox}

\newcommand*{\LWR@textfitscale}[2]{%
\setlength{\LWR@templengthone}{#1}%
\setlength{\LWR@templengthone}{%
    1em*\ratio{\LWR@templengthone}{\LWR@templengththwo}%
}%
\InlineClass[font-size:\LWR@printlength{\LWR@templengthone}]{textfit}{#2}%
}

\newcommand*{\scaletowidth}[2]{%
\sbox{\LWR@textfitbox}{#2}%
\settowidth{\LWR@templengththwo}{\usebox{\LWR@textfitbox}}%
\LWR@textfitscale{#1}{#2}%
}

\newcommand*{\scaletoheight}[2]{%
\sbox{\LWR@textfitbox}{#2}%
\settoheight{\LWR@templengththwo}{\usebox{\LWR@textfitbox}}%
\LWR@textfitscale{#1}{#2}%
}
```

File 330 **lwarp-textpos.sty**

§ 427 Package textpos

(*Emulates or patches code by* Norman Gray.)

Pkg textpos textpos is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{textpos}[2016/06/07]

\NewDocumentEnvironment{textblock}{m r()}{}{}
\NewDocumentEnvironment{textblock*}{m o r()}{}{}
\newcommand*{\TPGrid}[3][]{}
\NewDocumentCommand{\TPMargin}{s o}{}
\newcommand*{\textblockcolour}[1]{}
```

```
\newcommand*{\textblockrulecolour}[1]{}
\newcommand*{\textblockcolor}[1]{}
\newcommand*{\textblockrulecolor}[1]{}
\newcommand*{\tekstblokkulur}[1]{}
\newcommand*{\tekstblokrulekulur}[1]{}
\newlength{\TPHorizModule}
\newlength{\TPVertModule}
\newlength{\TPboxrulesize}
\newcommand{\textblocklabel}[1]{}
\newcommand*{\showtextsize}{}
\newcommand{\textblockorigin}[2]{}
```

File 331 **lwarp-theorem.sty**

§ 428 Package

theorem

(*Emulates or patches code by* Frank Mittelbach.)

Pkg theorem theorem is patched for use by lwarp.

Table 16: Theorem package — css styling of theorems and proofs

Theorem: of class theorembody<theoremstyle>

Theorem Header: of class theoremheader

where <theoremstyle> is plain, break, etc.

for HTML output:

```
\LWR@ProvidesPackagePass{theorem}[2014/10/28]
```

§ 428.1 Remembering the theorem style

Storage for the style being used for new theorems:

```
\newcommand{\LWR@newtheoremstyle}{plain}
```

Patched to remember the style being used for new theorems:

```
\gdef\theoremstyle#1{%
   \@ifundefined{th@#1}{\@warning
          {Unknown theoremstyle `#1'. Using `plain'}%
          \theorem@style{plain}%
           \renewcommand{\LWR@newtheoremstyle}{plain}% lwarp
          }%
      {%
          \theorem@style{#1}%
          \renewcommand{\LWR@newtheoremstyle}{#1}% lwarp
      }%
      \begingroup
        \csname th@\the\theorem@style \endcsname
      \endgroup}
```

Patched to remember the style for this theorem type, and set it later when the environment is started.

```
\gdef\@xnthm#1#2[#3]{%
   \expandafter\@ifdefinable\csname #1\endcsname
   {%
    \csedef{LWR@thmstyle#1}{\LWR@newtheoremstyle}% lwarp
    \@definecounter{#1}\@newctr{#1}[#3]%
    \expandafter\xdef\csname the#1\endcsname
      {\expandafter \noexpand \csname the#3\endcsname
       \@thmcountersep \@thmcounter{#1}}%
    \def\@tempa{\global\@namedef{#1}}%
    \expandafter \@tempa \expandafter{%
      \csname th@\the \theorem@style
            \expandafter \endcsname \the \theorem@bodyfont
     \@thm{#1}{#2}}%
    \global \expandafter \let \csname end#1\endcsname \@endtheorem
   \AtBeginEnvironment{#1}{\edef\LWR@thisthmstyle{\@nameuse{LWR@thmstyle#1}}}% lwarp
   }}

\gdef\@ynthm#1#2{%
    \expandafter\@ifdefinable\csname #1\endcsname
   {
    \csedef{LWR@thmstyle#1}{\LWR@newtheoremstyle}% lwarp
    \@definecounter{#1}%
    \expandafter\xdef\csname the#1\endcsname{\@thmcounter{#1}}%
    \def\@tempa{\global\@namedef{#1}}\expandafter \@tempa
     \expandafter{\csname th@\the \theorem@style \expandafter
     \endcsname \the\theorem@bodyfont \@thm{#1}{#2}}%
    \global \expandafter \let \csname end#1\endcsname \@endtheorem
   \AtBeginEnvironment{#1}{\edef\LWR@thisthmstyle{\@nameuse{LWR@thmstyle#1}}}% lwarp
   }}

\gdef\@othm#1[#2]#3{%
  \expandafter\ifx\csname c@#2\endcsname\relax
   \@nocounterr{#2}%
  \else
   \expandafter\@ifdefinable\csname #1\endcsname
   {
    \csedef{LWR@thmstyle#1}{\LWR@newtheoremstyle}% lwarp
    \expandafter \xdef \csname the#1\endcsname
     {\expandafter \noexpand \csname the#2\endcsname}%
    \def\@tempa{\global\@namedef{#1}}\expandafter \@tempa
     \expandafter{\csname th@\the \theorem@style \expandafter
     \endcsname \the\theorem@bodyfont \@thm{#2}{#3}}%
    \global \expandafter \let \csname end#1\endcsname \@endtheorem
   \AtBeginEnvironment{#1}{\edef\LWR@thisthmstyle{\@nameuse{LWR@thmstyle#1}}}% lwarp
    }%
  \fi}
```

§ 428.2 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`.

```
\gdef\th@plain{%
  \def\@begintheorem##1##2{%
        \item[
            \InlineClass{theoremheader}{##1\ ##2}
        ]
    }%
\def\@opargbegintheorem##1##2##3{%
   \item[
        \InlineClass{theoremheader}{##1\ ##2\ (##3)}
    ]
    }
}

\gdef\th@break{%
  \def\@begintheorem##1##2{%
    \item[
        \InlineClass{theoremheader}{##1\ ##2}\newline%
    ]
    }%
\def\@opargbegintheorem##1##2##3{%
  \item[
        \InlineClass{theoremheader}{##1\ ##2\ (##3)}\newline
    ]
    }
}

\gdef\th@marginbreak{%
  \def\@begintheorem##1##2{
    \item[
        \InlineClass{theoremheader}{##2 \qquad ##1}\newline
    ]
    }%
\def\@opargbegintheorem##1##2##3{%
    \item[
        \InlineClass{theoremheader}{##2 \qquad ##1\ %
        (##3)}\newline
    ]
    }
}

\gdef\th@changebreak{%
  \def\@begintheorem##1##2{
    \item[
        \InlineClass{theoremheader}{##2\ ##1}\newline
    ]
```

```
    }%
\def\@opargbegintheorem##1##2##3{%
    \item[
        \InlineClass{theoremheader}{ ##2\ ##1\ %
        (##3)}\newline
    ]
    }
}

\gdef\th@change{%
  \def\@begintheorem##1##2{
    \item[
        \InlineClass{theoremheader}{##2\ ##1}
    ]
    }%
\def\@opargbegintheorem##1##2##3{%
    \item[
        \InlineClass{theoremheader}{##2\ ##1\ (##3)}
    ]
    }
}

\gdef\th@margin{%
  \def\@begintheorem##1##2{
    \item[
        \InlineClass{theoremheader}{##2 \qquad ##1}
    ]
    }%
\def\@opargbegintheorem##1##2##3{%
    \item[
            \InlineClass{theoremheader}{##2 \qquad ##1\ (##3)}
    ]
    }
}
```

Patched for css:

```
\gdef\@thm#1#2{\refstepcounter{#1}%
\LWR@forcenewpage% lwarp
    \BlockClass{theorembody\LWR@thisthmstyle}% lwarp
   \trivlist
   \@topsep \theorempreskipamount               % used by first \item
   \@topsepadd \theorempostskipamount           % used by \@endparenv
   \@ifnextchar [%
   {\@ythm{#1}{#2}}%
   {\@begintheorem{#2}{\csname the#1\endcsname}\ignorespaces}}

\gdef\@endtheorem{%
\endtrivlist
\endBlockClass
}
```

File 332 **lwarp-thinsp.sty**

§ 429 Package thinsp

Pkg thinsp thinsp is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{thinsp}[2016/10/02]

\AtBeginDocument{
\let\thinthinspace\relax% defined by some packages
\newcommand*{\thinthinspace}{\thinspace}
}

\newcommand*{\stretchthinspace}{\thinspace}
\newcommand*{\stretchthinthinspace}{\thinthinspace}
\newcommand*{\stretchnegthinspace}{\negthinspace}
```

File 333 **lwarp-threadcol.sty**

§ 430 Package threadcol

Pkg threadcol threadcol is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{threadcol}[2013/01/06]

\newcommand{\setthreadname}[1]{}
```

File 334 **lwarp-threeparttable.sty**

§ 431 Package threeparttable

(*Emulates or patches code by* Donald Arseneau.)

Pkg threeparttable threeparttable is emulated.

Table note are contained inside a css of class tnotes. If enumitem is used, the note item labels are also individually highlighted with an additional css of class tnoteitemheader, otherwise they are plain text.

for HTML output:

```
\LWR@ProvidesPackageDrop{threeparttable}[2003/06/13]
```

Env threeparttable [⟨*alignment*⟩] To emulate threeparttable:

```
\newenvironment*{threeparttable}[1][b]{}{}
```

Env tablenotes [⟨*options*⟩]

```
\newenvironment*{tablenotes}[1][]
{%
\LWR@forcenewpage
\BlockClass{tnotes}%
\description%
}
{%
\enddescription%
\endBlockClass%
}
```

\tnote {⟨*text*⟩}

```
\newcommand{\tnote}[1]{\LWR@htmlspan{sup}{#1}}
```

File 335 **lwarp-thumb.sty**

§ 432 Package thumb

Pkg thumb thumb is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{thumb}[1997/12/24]
\newcommand*{\Overviewpage}{}
\newlength{\thumbheight}
\newlength{\thumbwidth}
```

File 336 **lwarp-thumbs.sty**

§ 433 Package thumbs

Pkg thumbs thumbs is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{thumbs}[2014/03/09]
\newcommand{\addthumb}[4]{}
\newcommand{\addtitlethumb}[5]{}
\newcommand{\stopthumb}{}
\newcommand{\continuethumb}{}
\newcommand{\thumbsoverview}[1]{}
\newcommand{\thumbsoverviewback}[1]{}
\newcommand{\thumbsoverviewverso}[1]{}
\newcommand{\thumbsoverviewdouble}[1]{}
\newcommand{\thumbnewcolumn}{}
\newcommand{\addthumbsoverviewtocontents}[2]{}
\newcommand{\thumbsnophantom}{}
```

File 337 **lwarp-tikz.sty**

§ 434 Package

tikz

(*Emulates or patches code by* Till Tantau.)

Pkg tikz tikz is supported.

⚠ 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 \&.

Accept all options for lwarp-tikz:

```
\LWR@ProvidesPackagePass{tikz}[2015/08/07]
```

catcodes lwarp changes the catcode of $ for its own use. The Ti*k*z babel library temporarily changes catcodes back to normal for Ti*k*z's use. tikz v3.0.0 introduced the babel library which handles catcode changes. For older versions, lwarp must change $'s catcode itself.

Also see:
https://tex.stackexchange.com/questions/16199/test-if-a-package-or-package-option-is-loaded

```
\newbool{LWR@tikzbabel}

\@ifpackagelater{tikz}{2013/12/20}% Test for Tikz version v3.0.0
{\usetikzlibrary{babel}\booltrue{LWR@tikzbabel}}
{\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.

```
\preto\pgfpicture{%
    \begin{lateximage}%
    \ifbool{LWR@tikzbabel}% Test for Tikz version v3.0.0
    {}%
    {\catcode`\$=3}% dollar sign is math shift
}

\appto\endpgfpicture{\end{lateximage}}
```

Ti*k*z is placed inside an SVG image, so use the original meanings of the following:

```
\LetLtxMacro\pgfutil@minipage\LWR@print@minipage
\let\pgfutil@endminipage\endLWR@print@minipage

\let\pgfutil@raggedleft\LWR@print@raggedleft
\let\pgfutil@raggedright\LWR@print@raggedright

\def\pgfutil@font@tiny{\LWR@printtiny}
\def\pgfutil@font@scriptsize{\LWR@printscriptsize}
\def\pgfutil@font@footnotesize{\LWR@printfootnotesize}
\def\pgfutil@font@small{\LWR@print@small}
\def\pgfutil@font@normalsize{\LWR@print@normalsize}
\def\pgfutil@font@large{\LWR@printlarge}
\def\pgfutil@font@Large{\LWR@printLarge}
\def\pgfutil@font@huge{\LWR@printhuge}
\def\pgfutil@font@Huge{\LWR@printHuge}

\def\pgfutil@font@itshape{\LWR@origitshape}
\def\pgfutil@font@bfseries{\LWR@origbfseries}

\def\pgfutil@font@normalfont{\LWR@orignormalfont}
```

File 338 **lwarp-titleps.sty**

§ 435 Package

titleps

(*Emulates or patches code by* Javier Bezos.)

Pkg titleps

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:

```
\LWR@ProvidesPackageDrop{titleps}[2016/03/15]
```

\pagestyle and \thispagestyle are already disabled in the lwarp code.

\newpagestyle {⟨*name*⟩} [⟨*style*⟩] {⟨*commands*⟩}

```
\NewDocumentCommand{\newpagestyle}{m o m}{}
```

\renewpagestyle {⟨*name*⟩} [⟨*style*⟩] {⟨*commands*⟩}

```
\NewDocumentCommand{\renewpagestyle}{m o m}{}
```

\sethead [⟨*el*⟩] [⟨*ec*⟩] [⟨*er*⟩] {⟨*ol*⟩} {⟨*oc*⟩} {⟨*or*⟩}

```
\NewDocumentCommand{\sethead}{o o o m m m}{}
```

\setfoot [⟨*el*⟩] [⟨*ec*⟩] [⟨*er*⟩] {⟨*ol*⟩} {⟨*oc*⟩} {⟨*or*⟩}

```
\NewDocumentCommand{\setfoot}{o o o m m m}{}
```

\settitlemarks * {⟨*names*⟩}

```
\NewDocumentCommand{\settitlemarks}{s m}{}
```

\headrule

```
\newcommand*{\headrule}{}
```

\footrule

```
\newcommand*{\footrule}{}
```

\setheadrule {⟨*length*⟩}

```
\newcommand*{\setheadrule}[1]{}
```

\setfootrule {⟨*length*⟩}

```
\newcommand*{\setfootrule}[1]{}
```

\makeheadrule

```
\newcommand*{\makeheadrule}{}
```

\makefootrule

```
\newcommand*{\makefootrule}{}
```

\setmarkboth {⟨*code*⟩}

```
\newcommand{\setmarkboth}[1]{}
```

\widenhead

```
\NewDocumentCommand{\widenhead}{s o o m m}{}
```

\bottitlemarks

```
\newcommand*{\bottitlemarks}{}
```

\toptitlemarks

```
\newcommand*{\toptitlemarks}{}
```

\firsttitlemarks

```
\newcommand*{\firsttitlemarks}{}
```

\nexttitlemarks

```
\newcommand*{\nexttoptitlemarks}{}
```

\outertitlemarks

```
\newcommand*{\outertitlemarks}{}
```

\innertitlemarks

```
\newcommand*{\innertitlemarks}{}
```

\newtitlemark * {⟨*name*⟩}

```
\NewDocumentCommand{\newtitlemark}{s m}{}
```

\pretitlemark * {⟨*section*⟩} {⟨*text*⟩}

```
\NewDocumentCommand{\pretitlemark}{s m m}{}
```

\ifsamemark {⟨*group*⟩} {⟨*command*⟩} {⟨*true*⟩} {⟨*false*⟩}

```
\newcommand{\ifsamemark}[4]{}
```

\setfloathead * [⟨.⟩] [⟨.⟩] [⟨.⟩] {⟨.⟩} {⟨.⟩} {⟨.⟩} {⟨*extra*⟩} [⟨*which*⟩]

```
\NewDocumentCommand{\setfloathead}{s o o o m m m m m}{}
```

\setfloatfoot * [⟨.⟩] [⟨.⟩] [⟨.⟩] {⟨.⟩} {⟨.⟩} {⟨.⟩} {⟨*extra*⟩} [⟨*which*⟩]

```
\NewDocumentCommand{\setfloatfoot}{s o o o m m m m m}{}
```

\nextfloathead * [⟨.⟩] [⟨.⟩] [⟨.⟩] {⟨.⟩} {⟨.⟩} {⟨.⟩} {⟨*extra*⟩} [⟨*which*⟩]

```
\NewDocumentCommand{\nextfloathead}{s o o o m m m m m}{}
```

\nextfloatfoot * [⟨.⟩] [⟨.⟩] [⟨.⟩] {⟨.⟩} {⟨.⟩} {⟨.⟩} {⟨*extra*⟩} [⟨*which*⟩]

```
\NewDocumentCommand{\nextfloatfoot}{s o o o m m m m m}{}
```

\newmarkset {⟨*markset*⟩}

```
\newcommand{\newmarkset}[1]{}
```

\newextramark * {⟨*markset*⟩} {⟨*macro-name*⟩}

```
\NewDocumentCommand{\newextramarkset}{s m m}{}
```

\botextramarks {⟨*markset*⟩}

```
\newcommand{\botextramarks}[1]{}
```

\topextramarks {⟨*markset*⟩}

```
\newcommand{\topextramarks}[1]{}
```

\firstextramarks {⟨*markset*⟩}

```
\newcommand{\firstextramarks}[1]{}
```

\nextextramarks {⟨*markset*⟩}

```
\newcommand{\nexttopextramarks}[1]{}
```

\outerextramarks {⟨*markset*⟩}

```
\newcommand{\outerextramarks}[1]{}
```

\innerextramarks {⟨*markset*⟩}

```
\newcommand{\innerextramarks}[1]{}
```

File 339 **lwarp-titleref.sty**

§ 436 Package titleref

Pkg titleref titleref is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{titleref}[2001/04/05]

\LetLtxMacro\titleref\nameref

\providecounter{LWR@currenttitle}

\newcommand*{\currenttitle}{%
    \addtocounter{LWR@currenttitle}{1}%
    \label{currenttitle\arabic{LWR@currenttitle}}%
    \nameref{currenttitle\arabic{LWR@currenttitle}}%
}

\newcommand*{\theTitleReference}[2]{}
```

File 340 **lwarp-titlesec.sty**

§ 437 Package titlesec

(*Emulates or patches code by* Javier Bezos.)

Pkg titlesec titlesec is emulated. All user options and macros are ignored and disabled.

Discard all options for lwarp-titlesec:

for HTML output:

```
\LWR@ProvidesPackageDrop{titlesec}[2016/03/21]
```

\titlelabel {⟨*label-format*⟩}

```
\newcommand*{\titlelabel}[1]{}
```

\titleformat* {⟨*command*⟩} {⟨*format*⟩}

\titleformat {⟨*command*⟩} [⟨*shape*⟩] {⟨*format*⟩} {⟨*label*⟩} {⟨*sep*⟩} {⟨*begfore*⟩} [⟨*after*⟩]

```
\newcommand\titleformat{%
  \@ifstar{\ttl@format@s}%
          {\ttl@format@i}}
\newcommand{\ttl@format@s}[1]{}
\NewDocumentCommand{\ttl@format@i}{m o m m m m o}{}
```

\chaptertitlename

```
\@ifundefined{@chapapp}{\let\@chapapp\chaptername}{}
\newcommand\chaptertitlename{\@chapapp}
```

\titlespacing * {⟨*command*⟩} {⟨*left*⟩} {⟨*before*⟩} {⟨*after*⟩} [⟨*right*⟩]

```
\NewDocumentCommand{\titlespacing}{s m m m m o}{}
```

\filright

```
\newcommand*{\filright}{}
```

\filcenter

```
\newcommand*{\filcenter}{}
```

\filleft

```
\newcommand*{\filleft}{}
```

\fillast

```
\newcommand*{\fillast}{}
```

\filinner

```
\newcommand*{\filinner}{}
```

\filouter

```
\newcommand*{\filouter}{}
```

\wordsep

```
\newcommand\wordsep{\fontdimen\tw@\font \@plus
  \fontdimen\thr@@\font \@minus \fontdimen4\font}
```

\titleline * [⟨*align*⟩] {⟨*material*⟩}

```
\NewDocumentCommand{\titleline}{s o m}{}
```

\titlerule [⟨*height*⟩]

```
\providecommand*\titlerule{\@ifstar{\ttl@row}{\ttl@rule}}
\newcommand*{\ttl@rule}[1][]{}
\newcommand*{\ttl@row}[2][]{}
```

\iftitlemeasuring {⟨*true*⟩} {⟨*false*⟩}

```
\newcommand{\iftitlemeasuring}[2]{#2}
```

\assignpagestyle {⟨*command*⟩} {⟨*pagestyle*⟩}

```
\newcommand{\assignpagestyle}[2]{#2}
```

\titleclass {⟨*name*⟩} [⟨*startlevel*⟩] {⟨*class*⟩} [⟨*cmd*⟩]

```
\NewDocumentCommand{\titleclass}{m o m o}
```

File 341 **lwarp-titletoc.sty**

§ 438 Package

titletoc

(*Emulates or patches code by* Javier Bezos.)

Pkg titletoc titletoc is emulated. All user options and macros are ignored and disabled.

Discard all options for lwarp-titletoc:

for HTML output:

```
\LWR@ProvidesPackageDrop{titletoc}[2011/12/15]
```

\dottedcontents {⟨*section*⟩} [⟨*left*⟩] {⟨*above*⟩} {⟨*label*⟩} {⟨*leader*⟩}

```
\NewDocumentCommand{\dottedcontents}{m o m m m}{}
```

\titlecontents * {⟨*section*⟩} [⟨*left*⟩] {⟨*above*⟩} {⟨*numbered*⟩} {⟨*numberless*⟩} {⟨*filler*⟩} [⟨*below or begin*⟩] [⟨*separator*⟩] [⟨*end*⟩]

```
\newcommand{\titlecontents}{\@ifstar{\ttl@tcstar}{\ttl@tcnostar}}
\NewDocumentCommand{\ttl@tcstar}{m o m m m m o o o}{}
\NewDocumentCommand{\ttl@tcnostar}{m o m m m m o}{}
```

\contentsmargin [⟨*correction*⟩] {⟨*right*⟩}

```
\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*⟩}

```
\newcommand{\printlist}[4][]{}
```

File 342 **lwarp-titling.sty**

§ 439 Package

titling

(*Emulates or patches code by* Peter Wilson.)

Pkg titling

package support
⚠ load order

lwarp supports the native LaTeX titling commands, and also supports the packages authblk and titling. If both are used, authblk should be loaded before titling.

\published and \subtitle

If using the titling package, additional titlepage fields for \published and \subtitle may be added by using \AddSubtitlePublished in the preamble. See section 65.8.

The various titling footnote restyling commands have no effect.

Pass all options to lwarp-titling:

for HTML output:

```
\LWR@ProvidesPackagePass{titling}[2009/09/04]
```

\@bsmtitlempty

Patch \@bsmtitlempty:

```
\let\LWR@orig@bsmtitlempty\@bsmtitlempty
\renewcommand*{\@bsmtitlempty}{%
\LWR@orig@bsmtitlempty%
}
```

\keepthetitle

Patch \keepthetitle:

```
\let\LWR@origkeepthetitle\keepthetitle
\renewcommand*{\keepthetitle}{%
\LWR@orig@keepthetitle%
}
```

\killtitle

Patch \killtitle:

```
\let\LWR@origkilltitle\killtitle
\renewcommand*{\killtitle}{%
\LWR@orig@killtitle%
}
```

Env titlingpage

```
\renewenvironment*{titlingpage}
{%
```

Start an HTML titlepage div:

```
\LWR@printpendingfootnotes
\begin{titlepage}
```

Prepare for a custom version of \maketitle inside the titlingpage:

```
\LWR@maketitlesetup
\let\maketitle\LWR@titlingmaketitle
}
{
```

At the end of the environment, end the HTML titlepage div:

```
\end{titlepage}
}
```

Patch the pre/post title/author/date to add HTML tags, then initilize:

```

\pretitle{}
\posttitle{}

\preauthor{}
\postauthor{}

\predate{}
\postdate{}
```

\LWR@maketitlesetup Patches \thanks macros.

```
\renewcommand*{\LWR@maketitlesetup}{%
```

Redefine the footnote mark:

```
\def\@makefnmark{\textsuperscript{\@thefnmark}}
```

\thefootnote ⇒ \nameuse{arabic}{footnote}, or
\thefootnote ⇒ \nameuse{fnsymbol}{footnote}

Redefine the footnote text:

```
\long\def\@makefntext##1{%
```

Make the footnote mark and some extra horizontal space for the tags:

```
\makethanksmark~%
```

\makethanksmark ⇒ \thanksfootmark ⇒ \tamark ⇒
\@thefnmark ⇒ \itshape a (or similar)

Print the text:

```
##1%
}% \@makefntext
}
```

\thanksfootmark

```
\renewcommand{\thanksfootmark}{%
%   \hb@xt@\thanksmarkwidth{\hfil\normalfont%
        \thanksscript{%
            \thanksfootpre \tamark \thanksfootpost%
        }%
%   }%
}
```

\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.

```
\renewcommand*{\maketitle}{%
```

An HTML titlepage is used for all classes.

```
\begin{titlepage}
```

Select which kind of footnote marks to use:

```
\@bsmarkseries
```

Set up special patches:

```
\LWR@maketitlesetup
```

Typeset the title, etc:

```
\@maketitle
```

Immediately generate any \thanks footnotes:

```
\@thanks
```

Close the HTML titlepage div:

```
\end{titlepage}
```

Reset the footnote counter:

```
\@bscontmark
}
```

\@maketitle Typesets the title, etc. Patched for HTML.

```
\DeclareDocumentCommand{\@maketitle}{}{%
    \maketitlehooka
    {
        \LWR@stoppars\LWR@htmltag{\LWR@tagtitle}%
```

```
        \@bspretitle \@title \@bsposttitle%
        \LWR@htmltag{\LWR@tagtitleend}\LWR@startpars%
    }
    \maketitlehookb
    {
        \begin{BlockClass}{author}
        \renewcommand{\and}{%
            \end{BlockClass}%
            \begin{BlockClass}{oneauthor}%
        }
        \begin{BlockClass}{oneauthor}%
        \@bspreauthor \@author  \@bspostauthor%
        \end{BlockClass}%
        \end{BlockClass}%
    }
    \maketitlehookc
    {
        \begin{BlockClass}{titledate}%
        \@bspredate \@date \@bspostdate%
        \end{BlockClass}%
    }
    \maketitlehookd
}
```

\LWR@titlingmaketitle `\maketitle` for use inside an HTML `titlingpage` environment.

```
\renewcommand*{\LWR@titlingmaketitle}{%
```

Keep pending footnotes out of the title block:

```
\@thanks
```

Select which kind of footnote marks to use:

```
\@bsmarkseries
```

Set up special patches:

```
\LWR@maketitlesetup
```

Typeset the title, etc:

```
\@maketitle
```

Immediately generate any `\thanks` footnotes:

```
\@thanks
```

Reset the footnote counter:

```
\@bscontmark
}
```

\thanksmarkseries {⟨*series*⟩}

Sets the type of footnote marks used by \thanks, where type is 'arabic', 'roman', 'fnsymbol', etc.

```
\renewcommand{\thanksmarkseries}[1]{%
\def\@bsmarkseries{\renewcommand{\thefootnote}{\@nameuse{#1}{footnote}}}%
}
```

Set default titlepage thanks footnote marks. See section 65.7.

```
\@ifclassloaded{memoir}{
   \thanksmarkseries{arabic}
}{% not memoir
\if@titlepage
   \thanksmarkseries{arabic}
\else
   \thanksmarkseries{fnsymbol}
\fi
}% not memoir
```

File 343 **lwarp-tocbasic.sty**

§ 440 Package

tocbasic

(*Emulates or patches code by* Markus Kohm.)

Pkg tocbasic 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:

```
\LWR@ProvidesPackagePass{tocbasic}[2018/12/30]
```

```
\DeclareDocumentCommand{\usetocbasicnumberline}{o}{}
\DeclareDocumentCommand{\DeclareTOCStyleEntry}{o m m}{}
\DeclareDocumentCommand{\DeclareTOCStyleEntries}{o m m}{}
\DeclareDocumentCommand{\DeclareTOCEntryStyle}{m o m}{}
\DeclareDocumentCommand{\DefineTOCEntryOption}{m o m}{}
\DeclareDocumentCommand{\DefineTOCEntryBooleanOption}{m o m m m}{}
\DeclareDocumentCommand{\DefineTOCEntryCommandOption}{m o m m m}{}
\DeclareDocumentCommand{\DefineTOCEntryIfOption}{m o m m m}{}
\DeclareDocumentCommand{\DefineTOCEntryLengthOption}{m o m m m}{}
\DeclareDocumentCommand{\DefineTOCEntryNumberOption}{m o m m m}{}
\DeclareDocumentCommand{\CloneTOCEntryStyle}{m m}{}
\DeclareDocumentCommand{\TOCEntryStyleInitCode}{m m}{}
\DeclareDocumentCommand{\TOCEntryStyleStartInitCode}{m m}{}
```

File 344 **lwarp-tocbibind.sty**

§ 441 Package

tocbibind

(*Emulates or patches code by* Peter Wilson.)

Pkg tocbibind 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:

Pkg makeidx **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
```

Pkg makeidx **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
```

Pkg tocbibind **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
```

Pkg tocbibind **On its own HTML page, with an automatic TOC entry:**

```
\usepackage[nottoc]{tocbibind}
. . .
\cleardoublepage
\phantomsection % to fix print-version index link
\ForceHTMLPage
\printindex
```

Opt [tocbibind] numindex numbered index section Use the tocbibind numindex option to generate a numbered index. Without this option, the index heading has no number.

Other packages, such as imakeidx, may also have options for including the index in the Table of Contents.

for HTML output:

```
\let\simplechapterdelim\relax

\LWR@ProvidesPackagePass{tocbibind}[2010/10/13]

\renewenvironment{theindex}%
{%
     \if@bibchapter
        \if@donumindex
            \chapter{\indexname}
        \else
          \if@dotocind
            \chapter*{\indexname}
            \addcontentsline{toc}{chapter}{\LWR@isolate{\indexname}}
          \else
            \chapter*{\indexname}
          \fi
        \fi
     \else
        \if@donumindex
            \section{\indexname}
        \else
          \if@dotocind
            \section*{\indexname}
            \addcontentsline{toc}{\@tocextra}{\LWR@isolate{\indexname}}
          \else
            \section*{\indexname}
          \fi
        \fi
     \fi
\let\item\LWR@indexitem%
\let\subitem\LWR@indexsubitem%
\let\subsubitem\LWR@indexsubsubitem%
}{}
```

The following code is shared by anonchap.

```
\DeclareDocumentCommand{\simplechapter}{O{\@empty}}{%
    \def\@chapcntformat##1{%
        #1~\csname the##1\endcsname\simplechapterdelim\quad%
    }%
}

\DeclareDocumentCommand{\restorechapter}{}{%
\let\@chapcntformat\@seccntformat%
}
```

File 345 **lwarp-tocenter.sty**

§ 442 Package

tocenter

Pkg tocenter

tocenter is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{tocenter}[2004/12/09]

\NewDocumentCommand{\ToCenter}{s o m m}{}
\NewDocumentCommand{\FromMargins}{s o m m m m}{}
```

File 346 **lwarp-tocloft.sty**

§ 443 Package

tocloft

(*Emulates or patches code by* Peter Wilson.)

Pkg tocloft

tocloft is emulated. Most user options and macros are ignored and disabled. `\newlistof` and `\cftchapterprecis` are supported.

Pkg tocloft

⚠ tocloft & other packages

If using tocloft with tocbibind, anonchap, fncychap, or other packages which change chapter title formatting, load tocloft with its `titles` option, which tells tocloft to use standard LaTeX commands to create the titles, allowing other packages to work with it.

Discard all options for lwarp-tocloft:

for HTML output:

```
\LWR@ProvidesPackageDrop{tocloftpagestyle}[2017/08/31]
```

`\tocloftpagestyle` {⟨*style*⟩}

```
\newcommand{\tocloftpagestyle}[1]{}
```

`\cftmarktoc`

```
\newcommand*{\cftmarktoc}{}
```

`\cfttoctitlefont`

```
\newcommand*{\cfttoctitlefont}{}
```

`\cftaftertoctitle`

```
\newcommand*{\cftaftertoctitle}{}

\newlength{\cftbeforetoctitleskip}
\newlength{\cftaftertoctitleskip}
```

`\cftmarklof`

```
\newcommand*{\cftmarklof}{}
```

`\cftloftitlefont`

```
\newcommand*{\cftloftitlefont}{}
```

\cftafterloftitle

```
\newcommand*{\cftafterloftitle}{}

\newlength{\cftbeforeloftitleskip}
\newlength{\cftafterloftitleskip}
```

\cftmarklot

```
\newcommand*{\cftmarklot}{}
```

\cftlottitlefont

```
\newcommand*{\cftlottitlefont}{}
```

\cftafterlottitle

```
\newcommand*{\cftafterlottitle}{}

\newlength{\cftbeforelottitleskip}
\newlength{\cftafterlottitleskip}
```

\cftdot

```
\providecommand*{\cftdot}{.}
```

\cftdotsep

```
\providecommand*{\cftdotsep}{1}
```

\cftnodots

```
\providecommand*{\cftnodots}{5000}
```

\cftdotfill {⟨*sep*⟩}

```
\providecommand{\cftdotfill}[1]{}
```

\cftsetpnumwidth {⟨*length*⟩}

```
\DeclareDocumentCommand{\cftsetpnumwidth}{m}{}
```

\cftsetrmarg {⟨*length*⟩}

```
\DeclareDocumentCommand{\cftsetrmarg}{m}{}
```

\cftpnumalign {⟨*alignment*⟩}

```
\DeclareDocumentCommand{\cftpnumalign}{m}{}

\LWR@providelength{\cftparskip}
```

The part-related items are also provided by memoir:

```
\LWR@providelength{\cftbeforepartskip}
\LWR@providelength{\cftpartindent}
\LWR@providelength{\cftpartnumwidth}
\providecommand*{\cftpartfont}{}
\providecommand*{\cftpartpresnum}{}
\providecommand*{\cftpartaftersnum}{}
\providecommand*{\cftpartaftersnumb}{}
\providecommand*{\cftpartleader}{}
\providecommand*{\cftpartdotsep}{1}
\providecommand*{\cftpartpagefont}{}
\providecommand*{\cftpartafterpnum}{}
```

memoir uses the full name “chapter” instead of “chap”:

```
\LWR@providelength{\cftbeforechapskip}
\LWR@providelength{\cftchapindent}
\LWR@providelength{\cftchapnumwidth}
\newcommand*{\cftchapfont}{}
\newcommand*{\cftchappresnum}{}
\newcommand*{\cftchapaftersnum}{}
\newcommand*{\cftchapaftersnumb}{}
\newcommand*{\cftchapleader}{}
\newcommand*{\cftchapdotsep}{1}
\newcommand*{\cftchappagefont}{}
\newcommand*{\cftchapafterpnum}{}
```

The following do not appear in memoir:

```
\LWR@providelength{\cftbeforesecskip}
\LWR@providelength{\cftsecindent}
\LWR@providelength{\cftsecnumwidth}
\newcommand*{\cftsecfont}{}
\newcommand*{\cftsecpresnum}{}
\newcommand*{\cftsecaftersnum}{}
\newcommand*{\cftsecaftersnumb}{}
\newcommand*{\cftsecleader}{}
\newcommand*{\cftsecdotsep}{1}
\newcommand*{\cftsecpagefont}{}
\newcommand*{\cftsecafterpnum}{}

\LWR@providelength{\cftbeforesubsecskip}
\LWR@providelength{\cftsubsecindent}
\LWR@providelength{\cftsubsecnumwidth}
\newcommand*{\cftsubsecfont}{}
\newcommand*{\cftsubsecpresnum}{}
\newcommand*{\cftsubsecaftersnum}{}
\newcommand*{\cftsubsecaftersnumb}{}
\newcommand*{\cftsubsecleader}{}
\newcommand*{\cftsubsecdotsep}{1}
\newcommand*{\cftsubsecpagefont}{}
\newcommand*{\cftsubsecafterpnum}{}

\LWR@providelength{\cftbeforesubsubsecskip}
```

```
\LWR@providelength{\cftsubsubsecindent}
\LWR@providelength{\cftsubsubsecnumwidth}
\newcommand*{\cftsubsubsecfont}{}
\newcommand*{\cftsubsubsecpresnum}{}
\newcommand*{\cftsubsubsecaftersnum}{}
\newcommand*{\cftsubsubsecaftersnumb}{}
\newcommand*{\cftsubsubsecleader}{}
\newcommand*{\cftsubsubsecdotsep}{1}
\newcommand*{\cftsubsubsecpagefont}{}
\newcommand*{\cftsubsubsecafterpnum}{}

\LWR@providelength{\cftbeforeparaskip}
\LWR@providelength{\cftparaindent}
\LWR@providelength{\cftparanumwidth}
\newcommand*{\cftparafont}{}
\newcommand*{\cftparapresnum}{}
\newcommand*{\cftparaaftersnum}{}
\newcommand*{\cftparaaftersnumb}{}
\newcommand*{\cftparaleader}{}
\newcommand*{\cftparadotsep}{1}
\newcommand*{\cftparapagefont}{}
\newcommand*{\cftparaafterpnum}{}

\LWR@providelength{\cftbeforesubparaskip}
\LWR@providelength{\cftsubparaindent}
\LWR@providelength{\cftsubparanumwidth}
\newcommand*{\cftsubparafont}{}
\newcommand*{\cftsubparapresnum}{}
\newcommand*{\cftsubparaaftersnum}{}
\newcommand*{\cftsubparaaftersnumb}{}
\newcommand*{\cftsubparaleader}{}
\newcommand*{\cftsubparadotsep}{1}
\newcommand*{\cftsubparapagefont}{}
\newcommand*{\cftsubparaafterpnum}{}

\LWR@providelength{\cftbeforefigskip}
\LWR@providelength{\cftfigindent}
\LWR@providelength{\cftfignumwidth}
\newcommand*{\cftfigfont}{}
\newcommand*{\cftfigpresnum}{}
\newcommand*{\cftfigaftersnum}{}
\newcommand*{\cftfigaftersnumb}{}
\newcommand*{\cftfigleader}{}
\newcommand*{\cftfigdotsep}{1}
\newcommand*{\cftfigpagefont}{}
\newcommand*{\cftfigafterpnum}{}

\LWR@providelength{\cftbeforesubfigskip}
\LWR@providelength{\cftsubfigindent}
\LWR@providelength{\cftsubfignumwidth}
\newcommand*{\cftsubfigfont}{}
\newcommand*{\cftsubfigpresnum}{}
\newcommand*{\cftsubfigaftersnum}{}
\newcommand*{\cftsubfigaftersnumb}{}
\newcommand*{\cftsubfigleader}{}
```

```
\newcommand*{\cftsubfigdotsep}{1}
\newcommand*{\cftsubfigpagefont}{}
\newcommand*{\cftsubfigafterpnum}{}

\LWR@providelength{\cftbeforetabskip}
\LWR@providelength{\cfttabindent}
\LWR@providelength{\cfttabnumwidth}
\newcommand*{\cfttabfont}{}
\newcommand*{\cfttabpresnum}{}
\newcommand*{\cfttabaftersnum}{}
\newcommand*{\cfttabaftersnumb}{}
\newcommand*{\cfttableader}{}
\newcommand*{\cfttabdotsep}{1}
\newcommand*{\cfttabpagefont}{}
\newcommand*{\cfttabafterpnum}{}

\LWR@providelength{\cftbeforesubtabskip}
\LWR@providelength{\cftsubtabindent}
\LWR@providelength{\cftsubtabnumwidth}
\newcommand*{\cftsubtabfont}{}
\newcommand*{\cftsubtabpresnum}{}
\newcommand*{\cftsubtabaftersnum}{}
\newcommand*{\cftsubtabaftersnumb}{}
\newcommand*{\cftsubtableader}{}
\newcommand*{\cftsubtabdotsep}{1}
\newcommand*{\cftsubtabpagefont}{}
\newcommand*{\cftsubtabafterpnum}{}

\DeclareDocumentCommand{\cftsetindents}{m m m}{}

\newcommand{\pagenumbersoff}[1]{}
\newcommand{\pagenumberson}[1]{}
```

\newlistentry [⟨*within*⟩] {⟨*counter*⟩} {⟨*ext*⟩} {⟨*level-1*⟩}

```
\DeclareDocumentCommand{\newlistentry}{o m m m}
{%
\LWR@traceinfo{newlistentry #2 #3 #4}%
\IfValueTF{#1}%
{%
    \@ifundefined{c@#2}{%
        \newcounter{#2}[#1]%
        \expandafter\edef\csname the#2\endcsname{%
            \expandafter\noexpand\csname the#1\endcsname.\noexpand\arabic{#2}%
        }%
    }{}%
}%
{%
    \@ifundefined{c@#2}{%
        \newcounter{#2}%
    }{}%
}%
\@namedef{l@#2}##1##2{%
    \hypertocfloat{1}{#2}{#3}{##1}{##2}%
```

```
    \def\cftwhatismyname{#2}% from memoir
}%
\expandafter\newlength\csname cftbefore#2skip\endcsname%
\expandafter\newlength\csname cft#2indent\endcsname%
\expandafter\newlength\csname cft#2numwidth\endcsname%
\@namedef{cft#2font}{}%
\@namedef{cft#2presnum}{}%
\@namedef{cft#2aftersnum}{}%
\@namedef{cft#2aftersnumb}{}%
\@namedef{cft#2leader}{}%
\@namedef{cft#2dotsep}{1}%
\@namedef{cft#2pagefont}{}%
\@namedef{cft#2afterpnum}{}%
\@namedef{toclevel@#2}{#4}%
\@namedef{cft#2fillnum}##1{}%
\LWR@traceinfo{newlistentry done}%
}
```

\newlistof [⟨*within*⟩] {⟨*type*⟩} {⟨*ext*⟩} {⟨*listofname*⟩}

Emulated through the \newfloat mechanism.

```
\DeclareDocumentCommand{\newlistof}{o m m m}
{%
\IfValueTF{#1}
{\newlistentry[#1]{#2}{#3}{0}}
{\newlistentry{#2}{#3}{0}}
\@namedef{ext@#2}{#3}
\@ifundefined{c@#3depth}{\newcounter{#3depth}}{}
\setcounter{#3depth}{1}
\@namedef{cftmark#3}{}
\@namedef{listof#2}{\listof{#2}{#4}}
\@namedef{@cftmake#3title}{}
\expandafter\newlength\csname cftbefore#3titleskip\endcsname
\expandafter\newlength\csname cftafter#3titleskip\endcsname
\@namedef{cft#3titlefont}{}
\@namedef{cftafter#3title}{}
\@namedef{cft#3prehook}{}
\@namedef{cft#3posthook}{}
}
```

\cftchapterprecis {⟨*text*⟩}

```
\newcommand{\cftchapterprecis}[1]{%
  \cftchapterprecishere{#1}
  \cftchapterprecistoc{#1}}
\newcommand{\cftchapterprecishere}[1]{%
  \begin{quote}\textit{#1}\end{quote}}
\newcommand{\cftchapterprecistoc}[1]{
  \addtocontents{toc}{%
  {
     \protect\begin{quote}#1\protect\end{quote}}
  }
}
```

File 347 **lwarp-tocstyle.sty**

§ 444 Package

tocstyle

Pkg tocstyle tocstyle is ignored.

⚠ Not fully tested! Please send bug reports!

for HTML output:

```
\LWR@ProvidesPackageDrop{tocstyle}[2017/02/23]

\newcommand*{\usetocstyle}[2][]{}
\newcommand*{\deactivatetocstyle}[1][]{}
\newcommand*{\reactivatetocstyle}[1][]{}
\NewDocumentCommand{\settocfeature}{o o m m}{}
\NewDocumentCommand{\settocstylefeature}{o m m}{}
\NewDocumentCommand{\newtocstyle}{o o m m}{}
\newcommand*{\aliastoc}[2]{}
\newcommand*{\showtoc}[2][]{}
\newcommand{\iftochasdepth}[4]{}
```

File 348 **lwarp-todo.sty**

§ 445 Package

todo

(*Emulates or patches code by* Federico Garcia.)

Pkg todo todo is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{todo}[2010/03/31]

\renewcommand\todoitem[2]{%
    \refstepcounter{todo}%
    \item[%
        \HTMLunicode{2610} \quad
        \ref{todopage:\thetodo}
     ] : {\todoformat\ifx#1\todomark\else\textbf{#1} \fi}#2%
    \label{todolbl:\thetodo}%
}%

\renewcommand\doneitem[2]{%
    \stepcounter{todo}%
    \item[%
        \HTMLunicode{2611} \quad
        \ref{todopage:\thetodo}
    ] \@nameuse{@done\the\c@todo}:
        {\todoformat\ifx#1\todomark\else\textbf{#1} \fi}#2%
}

```

```
\xpatchcmd{\@displaytodo}
    {\todoformat #1}{\todoformat \textbf{#1}}{}
    {\PackageWarning{lwarp-todo}{Unable to patch @displaytodo.}}

\xpatchcmd{\@displayfulltodo}
    {\todoformat #1}{\todoformat \textbf{#1}}{}
    {\PackageWarning{lwarp-todo}{Unable to patch @displayfulltodo.}}

\patchcmd{\todoenv}{\itshape see text.}{\textit{see text.}}{}
    {\PackageWarning{lwarp-todo}{Unable to patch todoenv.}}

\patchcmd{\astodos}{\todoformat #1}{\todoformat \textbf{#1}}{}
    {\PackageWarning{lwarp-todo}{Unable to patch astodos.}}

\AtBeginDocument{
    \crefname{todo}{todo}{todos}
    \Crefname{todo}{Todo}{Todos}
}
```

File 349 **lwarp-todonotes.sty**

§ 446 Package

todonotes

(*Emulates or patches code by* Henrik Skov Midtiby.)

Pkg todonotes 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:

```
\LWR@ProvidesPackagePass{todonotes}[2012/07/25]

\if@todonotes@disabled
\else

\newcommand{\ext@todo}{tdo}

\renewcommand{\l@todo}[2]{\hypertocfloat{1}{todo}{ldo}{#1}{#2}}

\let\LWRTODONOTES@orig@todototoc\todototoc

\renewcommand*{\todototoc}{%
\phantomsection%
\LWRTODONOTES@orig@todototoc%
}

\renewcommand{\@todonotes@drawMarginNoteWithLine}{
\fcolorbox
    {\@todonotes@currentbordercolor}
    {\@todonotes@currentbackgroundcolor}
    {\arabic{@todonotes@numberoftodonotes}}
```

```
\marginpar{\@todonotes@drawMarginNote}
}

\renewcommand{\@todonotes@drawInlineNote}{%
\fcolorboxBlock%
    {\@todonotes@currentbordercolor}%
    {\@todonotes@currentbackgroundcolor}%
    {%
        \if@todonotes@authorgiven%
        {\@todonotes@author:\,}%
        \fi%
        \@todonotes@text%
    }%
}

\renewcommand{\@todonotes@drawMarginNote}{%
    \if@todonotes@authorgiven%
        \@todonotes@author\par%
    \fi%
    \arabic{@todonotes@numberoftodonotes}: %
    \fcolorbox%
    {\@todonotes@currentbordercolor}%
    {\@todonotes@currentbackgroundcolor}%
    {%
        \@todonotes@sizecommand%
        \@todonotes@text %
    }%
}%

\renewcommand{\@todonotes@drawLineToRightMargin}{}

\renewcommand{\@todonotes@drawLineToLeftMargin}{}

\renewcommand{\missingfigure}[2][]{%
\setkeys{todonotes}{#1}%
\addcontentsline{tdo}{todo}{\@todonotes@MissingFigureText: #2}%
\fcolorboxBlock%
    {\@todonotes@currentbordercolor}%
    {\@todonotes@currentfigcolor}%
    {%
        \setlength{\fboxrule}{4pt}%
        \fcolorbox{red}{white}{Missing figure} \quad #2%
    }
}

\LetLtxMacro\LWRTODONOTES@orig@todo\@todo

\RenewDocumentCommand{\@todo}{o m}{%
\begingroup%
\renewcommand*{\phantomsection}{}%
\IfValueTF{#1}{%
    \LWRTODONOTES@orig@todo[#1]{#2}%
}{%
    \LWRTODONOTES@orig@todo{#2}%
}
```

```
\endgroup%
}

\fi% \if@todonotes@disabled
```

File 350 **lwarp-tram.sty**

§ 447 Package tram

Pkg tram tram is emulated.

⚠ block only The HTML emulation uses a , which must not appear inside an HTML 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:

```
\LWR@ProvidesPackageDrop{tram}[2013/04/04]

\newenvironment{tram}[1][]%
    {\BlockClass[background:lightgray]{tram}}
    {\endBlockClass}
```

File 351 **lwarp-transparent.sty**

§ 448 Package transparent

(*Emulates or patches code by* Heiko Oberdiek.)

Pkg transparent Emulated. \texttransparent works for inline objects. \transparent only works for \includegraphics.

⚠ Not XeLaTeX! Note that transparent does not work with XeLaTeX.

for HTML output: Discard all options for lwarp-transparent:

```
\LWR@ProvidesPackageDrop{transparent}[2016/05/16]

\newcommand*{\transparent}[1]{\edef\LWR@opacity{#1}}

\newcommand*{\texttransparent}[2]{%
\begingroup%
\transparent{#1}%
\InlineClass[opacity: #1]{transparent}{#2}%
```

```
\endgroup%
}
```

File 352 **lwarp-trimclip.sty**

§ 449 Package trimclip

Pkg trimclip trimclip is nullified.

for HTML output:

```
\LWR@ProvidesPackageDrop{trimclip}[2018/04/08]
```

The third argument, the text, is not touched. This allows \bgroup / \egroup, and verbatim content.

```
\csdef{trimbox}{\@ifstar\@gobble\@gobble}
\csletcs{trimbox*}{trimbox}
\def\endtrimbox{}
\csletcs{endtrimbox*}{endtrimbox}

\csletcs{clipbox}{trimbox}
\csletcs{clipbox*}{trimbox}
\csletcs{endclipbox}{endtrimbox}
\csletcs{endclipbox*}{endtrimbox}

\csletcs{marginbox}{trimbox}
\csletcs{marginbox*}{trimbox}
\csletcs{endmarginbox}{endtrimbox}
\csletcs{endmarginbox*}{endtrimbox}
```

File 353 **lwarp-trivfloat.sty**

§ 450 Package trivfloat

(*Emulates or patches code by* Joseph Wright.)

Pkg trivfloat 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.

```
\LWR@ProvidesPackageDrop{trivfloat}[2009/04/23]
\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.

for HTML output:

```
\begin{warpHTML}
\AtBeginDocument{\DeclareDocumentCommand{\tfl@chapter@fix}{m m}{}}
\end{warpHTML}
```

§ 450.1 Combining `\newfloat`, `\trivfloat`, and algorithmicx

for HTML & PRINT:

```
\begin{warpall}
```

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
```

```
\end{warpall}
```

File 354 **`lwarp-truncate.sty`**

§ 451 Package truncate

`Pkg truncate` truncate is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{truncate}[2001/08/20]
\providecommand{\TruncateMarker}{}
\newcommand{\truncate}[3][\TruncateMarker]{#3}
```

File 355 **lwarp-turnthepage.sty**

§ 452 Package turnthepage

Pkg turnthepage turnthepage is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{turnthepage}[2011/03/24]
\newcommand{\turnthepage}{}
```

File 356 **lwarp-twoup.sty**

§ 453 Package twoup

Pkg twoup twoup is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{twoup}[2007/02/26]
\newcommand{\cleartolastpage}{}
```

File 357 **lwarp-typearea.sty**

§ 454 Package typearea

(*Emulates or patches code by* Markus Kohm.)

Pkg typearea 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:

```
\LWR@ProvidesPackageDrop{typearea}[2018/03/30]
\DeclareDocumentCommand{\typearea}{o m}{}
\DeclareDocumentCommand{\recalctypearea}{}{}
\@ifundefined{footheight}{\newlength\footheight}{}
\DeclareDocumentCommand{\areaset}{o m m}{}
\DeclareDocumentCommand{\activateareas}{}{}
\DeclareDocumentCommand{\storeareas}{m}{}
\DeclareDocumentCommand{\BeforeRestoreareas}{s m}{}
\DeclareDocumentCommand{\AfterRestoreareas}{s m}{}
\DeclareDocumentCommand{\AfterCalculatingTypearea}{s m}{}
\DeclareDocumentCommand{\AfterSettingArea}{s m}{}
```

File 358 **lwarp-typicons.sty**

§ 455 Package

typicons

(*Emulates or patches code by* Arthur Vigil, Xavier Danaux.)

Pkg typicons 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:

```
\LWR@ProvidesPackagePass{typicons}[2015/05/20]

\LetLtxMacro\LWR@orig@symbol\symbol

\let\LWR@orig@typicon@TI\TI

\newcommand*{\LWR@typicon@symbol}[1]{%
    \begin{lateximage}*[(typicon)][typicon#1]%
    \begingroup%
    \LWR@orig@typicon@TI%
    \LWR@orig@symbol{#1}%
    \endgroup%
    \end{lateximage}%
}

\renewcommand*{\TI}{%
    \LetLtxMacro\symbol\LWR@typicon@symbol%
}

\renewcommand*{\ticon}[1]
{%
    \begin{lateximage}*[(#1 icon)][typicon#1]%
    \TI\csname ticon@#1\endcsname%
    \end{lateximage}%
}
```

File 359 **lwarp-ulem.sty**

§ 456 Package

ulem

(*Emulates or patches code by* Donald Arseneau.)

Pkg ulem Patched for use by lwarp.

for HTML output: Use the original package:

```
\LWR@ProvidesPackagePass{ulem}[2012/05/18]
```

Basic markup commands, using css:

```
\NewDocumentCommand{\LWR@HTML@uline}{+m}{%
    \InlineClass%
        (text-decoration:underline; text-decoration-skip: auto)%
        {uline}{\LWR@isolate{#1}}%
}
\LWR@formatted{uline}

\NewDocumentCommand{\LWR@HTML@uuline}{+m}{%
    \InlineClass%
        (%
            text-decoration:underline; text-decoration-skip: auto;%
            text-decoration-style:double%
        )%
        {uuline}{\LWR@isolate{#1}}%
}
\LWR@formatted{uuline}

\NewDocumentCommand{\LWR@HTML@uwave}{+m}{%
    \InlineClass%
        (%
            text-decoration:underline; text-decoration-skip: auto;%
            text-decoration-style:wavy%
        )%
        {uwave}{\LWR@isolate{#1}}%
}
\LWR@formatted{uwave}

\NewDocumentCommand{\LWR@HTML@sout}{+m}{%
    \InlineClass%
        (text-decoration:line-through)%
        {sout}{\LWR@isolate{#1}}%
}
\LWR@formatted{sout}

\NewDocumentCommand{\LWR@HTML@xout}{+m}{%
    \InlineClass%
        (text-decoration:line-through)%
        {xout}{\LWR@isolate{#1}}%
}

\NewDocumentCommand{\LWR@HTML@dashuline}{+m}{%
    \InlineClass%
        (%
            text-decoration:underline;%
            text-decoration-skip: auto;%
            text-decoration-style:dashed%
        )%
        {dashuline}{\LWR@isolate{#1}}%
}
\LWR@formatted{xout}

\NewDocumentCommand{\LWR@HTML@dotuline}{+m}{%
    \InlineClass%
```

```
        (%
            text-decoration:underline;%
            text-decoration-skip: auto;%
            text-decoration-style: dotted%
        )%
        {dotuline}{\LWR@isolate{#1}}%
}
\LWR@formatted{dotuline}
```

Nullified/emulated macros:

```
\NewDocumentCommand{\LWR@HTML@markoverwith}{m}{}
\LWR@formatted{markoverwith}

\NewDocumentCommand{\LWR@HTML@ULon}{+m}{\uline{#1}\egroup}
\LWR@formatted{ULon}
```

File 360 **lwarp-umoline.sty**

§ 457 Package

umoline

(*Emulates or patches code by* Hiroshi Nakashima.)

Pkg umoline umoline is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{umoline}[2000/07/11]

\newcommand*{\LWR@HTML@Underline}[1]{%
    \InlineClass{uline}{#1}%
}
\LWR@formatted{Underline}

\newcommand*{\LWR@HTML@Midline}[1]{%
    \InlineClass{sout}{#1}%
}
\LWR@formatted{Midline}

\newcommand*{\LWR@HTML@Overline}[1]{%
    \InlineClass{oline}{#1}%
}
\LWR@formatted{Overline}

\newcommand*{\LWR@HTML@UMOline}[2]{%
    \InlineClass{uline}{#2}%
}
\LWR@formatted{UMOline}

\NewDocumentCommand{\LWR@HTML@UMOspace}{s m o}{\hspace*{#2}}
\LWR@formatted{UMOspace}

\NewDocumentCommand{\LWR@HTML@UMOnewline}{s}{\newline}
\LWR@formatted{UMOnewline}
```

File 361 **lwarp-underscore.sty**

§ 458 Package underscore

Pkg underscore underscore is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{underscore}[2006/09/13]
```

File 362 **lwarp-units.sty**

§ 459 Package units

(*Emulates or patches code by* Axel Reichert.)

Pkg units 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:

```
\LWR@ProvidesPackagePass{units}[1998/08/04]

\DeclareRobustCommand*{\LWR@HTML@unit}[2][]{%
\ifblank{#1}%
    {\LWR@textcurrentfont{#2}}%
    {%
        #1%
        \ifthenelse{\boolean{B@UnitsLoose}}{~}{\,}%
        \LWR@textcurrentfont{#2}%
    }%
}
\LWR@formatted{unit}

\DeclareRobustCommand*{\LWR@HTML@unitfrac}[3][]{%
\ifblank{#1}%
    {%
            \nicefrac{#2}{#3}%
    }%
    {%
            #1%
            \ifthenelse{\boolean{B@UnitsLoose}}{~}{\,}%
            \nicefrac{#2}{#3}%
    }%
}

\LWR@formatted{unitfrac}
```

For Mathjax:

```
\CustomizeMathJax{\newcommand{\unit}[2][]{#1 #2}}
\CustomizeMathJax{\newcommand{\unitfrac}[3][]{#1 #2/#3}}
```

File 363 **lwarp-unitsdef.sty**

§ 460 Package

unitsdef

(*Emulates or patches code by* Patrick Happel.)

Pkg unitsdef unitsdef is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{unitsdef}[2005/01/04]

\renewcommand{\unitvaluesep}{\,}

\renewcommand{\unittimes}{\@@setunitsepfalse\HTMLunicode{22c5}}% \cdot

\renewunit{\arcmin}{%
    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
        {\ensuremath{{}^\prime}}%
        {\HTMLunicode{2032}}% prime
}

\renewunit{\arcsec}{%
    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
        {\ensuremath{{}^{\prime\prime}}}%
        {\HTMLunicode{2033}}% dbl prime
}

\renewrobustcmd{\SI}[2]{%
  \begingroup%
    \let\unit@@xspace\relax%
    \unitSIdef\selectfont%
      \LWR@textcurrentfont{#1#2}% lwarp
  \endgroup%
}
```

File 364 **lwarp-upref.sty**

§ 461 Package

upref

Pkg upref Ignored.

for HTML output: Discard all options for lwarp-upref:

```
\LWR@ProvidesPackageDrop{upref}[2007/03/14]
```

File 365 **lwarp-url.sty**

§ 462 Package url

(*Emulates or patches code by* Donald Arseneau.)

Pkg url url is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{url}[2013/09/16]
```

url uses math mode to print its string inside a group, so the original meaning of math is restored first.

```
\LetLtxMacro\LWR@url@origUrl@FormatString\Url@FormatString

\renewcommand*{\Url@FormatString}{%
    \InlineClass{verbatim}{%
        \LWR@restoreorigformatting%
        \LWR@url@origUrl@FormatString%
    }%
}
```

File 366 **lwarp-uspace.sty**

§ 463 Package uspace

Pkg uspace uspace is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{uspace}[2016/11/06]
```

File 367 **lwarp-verse.sty**

§ 464 Package verse

(*Emulates or patches code by* Peter Wilson.)

Pkg verse verse is supported and patched by lwarp.

for HTML output: Pass all options for lwarp-verse:

```
\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}
```

Len \vleftskip
Len \vleftmargini
Len \HTMLvleftskip
Len \HTMLleftmargini

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 and \HTMLleftmargini are provided to control the margins in HTML output. These 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 case a label may be shifted by one space.

Env verse

The verse environment will be placed inside a HTML <pre>.

```
\AfterEndPreamble{
\LWR@traceinfo{Patching verse.}
```

At the beginning of the verse environment:

```
\AtBeginEnvironment{verse}
{%
```

Use the original list environment inside a <pre> to attempt to preserve formatting.

```
\LWR@restoreoriglists%
```

Pkg verse
Cls memoir
\flagverse
Len \vleftskip

The verse or memoir packages can place stanza numbers to the left with their \flagverse command. Do not allow them to go into the left margin, which would cause *pdfcrop* to crop the entire page further to the left:

```
\ifdef{\vleftskip}{%
\setlength{\vleftskip}{\HTMLvleftskip}
\setlength{\leftmargini}{\HTMLleftmargini}
}{}
\LWR@forcenewpage
\LWR@atbeginverbatim{3}{verse}%
}
```

After the end of the verse environment, which places the <pre> tag at the regular left margin:

```
\AtEndEnvironment{verse}{%
\leavevmode%
\LWR@afterendverbatim{1}%
}
```

Patch to place poemtitle inside an HTML of class poemtitle:

```
\ifdef{\poemtitle}{
\DeclareDocumentCommand{\@vstypeptitle}{m}{%
   \vspace{\beforepoemtitleskip}%
   {\InlineClass{poemtitle}{\poemtitlefont #1}\par}%
   \vspace{\afterpoemtitleskip}%
 }
}{}

\LWR@traceinfo{Finished patching verse.}
}% AfterEndPreamble
```

File 368 **lwarp-versonotes.sty**

§ 465 Package versonotes

(*Emulates or patches code by* Norman Gray.)

Pkg versonotes versonotes is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{versonotes}[2015/12/08]

\newcommand{\versonote}[1]{\marginpar{#1}}
\newdimen\versotextwidth
\newdimen\versoleftmargin
\newcommand*{\versolayout}{}
```

File 369 **lwarp-vertbars.sty**

§ 466 Package vertbars

(*Emulates or patches code by* Peter Wilson.)

Pkg vertbars vertbars is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{vertbars}[2010/11/27]

\newlength{\barwidth}
\setlength{\barwidth}{0.4pt}
\newlength{\barspace}
\setlength{\barspace}{1em}

\newenvironment{vertbar}{
```

```
    \LWR@forcenewpage
    \LWR@forceminwidth{\barwidth}
    \begin{BlockClass}[%
        border-left: \LWR@printlength{\LWR@atleastonept} solid black ; %
        padding-left: \LWR@printlength{\barspace}%
    ]{vertbar}
}{
    \end{BlockClass}
}
```

File 370 **lwarp-vmargin.sty**

§ 467 Package vmargin

Pkg vmargin vmargin is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{vmargin}[2004/07/15]
```

```
\newcommand*{\LWRVM@customsize}[2]{}
\newcommand*{\setpapersize}[2][]{\ifstrequal{#2}{custom}{\LWRVM@customsize}{}}
\newcommand*{\setmargins}[8]{}
\newcommand*{\setmarginsrb}[8]{}
\newcommand*{\setmargnohf}[4]{}
\newcommand*{\setmargnohfrb}[4]{}
\newcommand*{\setmarg}[4]{}
\newcommand*{\setmargrb}[4]{}
\newlength{\PaperWidth}
\setlength{\PaperWidth}{8.5in}
\newlength{\PaperHeight}
\setlength{\PaperHeight}{11in}
\newif\ifLandscape
```

File 371 **lwarp-vowel.sty**

§ 468 Package vowel

(*Emulates or patches code by* FUKUI REI.)

Pkg vowel 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:

```
\LWR@ProvidesPackagePass{vowel}[2002/08/08]
```

```
\renewenvironment{vowel}[1][]
    {%
        \begin{lateximage}[(-vowel-~\packagediagramname)]%
        \@vowel[#1]%
    }
    {%
        \@@vowel%
        \end{lateximage}%
    }
```

File 372 **lwarp-vpe.sty**

§ 469 Package vpe

Pkg vpe vpe is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{vpe}[2012/04/18]
```

File 373 **lwarp-vwcol.sty**

§ 470 Package vwcol

(*Emulates or patches code by* Will Robertson.)

Pkg vwcol 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:

```
\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*⟩}

```
\newcommand*{\LWR@vwcol@addrule}[1]{%
    \appto{\LWR@vwcolstyle}{%
        #1: %
     \LWR@printlength{\vwcol@rule} solid \LWR@origpound\LWR@vwcol@rulecolor ; %
    }%
}
```

`\LWR@vwcol@addrule` {⟨*style tag*⟩}

```
\newcommand*{\LWR@vwcol@addgap}[1]{%
    \appto{\LWR@vwcolstyle}{%
        #1: %
```

```
        \LWR@printlength{\vwcol@sep} ; %
    }%
}
```

Env vwcol {⟨*key/values*⟩}

Redefine the environment to add a HTML style. The style is built depending on the required options.

```
\renewenvironment*{vwcol}[1][]{%
```

New paragraph, and process the options:

```
\par\noindent%
\vwcolsetup{#1}%
```

Begin with no style:

```
\newcommand*{\LWR@vwcolstyle}{}
```

presep and postsep are created with HTML margins:

```
\if@vwcol@presep
    \appto{\LWR@vwcolstyle}{margin-left: 1em ; padding-left: .5em ; }
\fi
\if@vwcol@postsep
    \appto{\LWR@vwcolstyle}{margin-right: 1em ; padding-right: .5em ; }
\fi
```

sep becomes column-gap:

```
\ifdimgreater{\vwcol@sep}{1sp}{
    \LWR@vwcol@addgap{column-gap}
    \LWR@vwcol@addgap{-moz-column-gap}
    \LWR@vwcol@addgap{-webkit-column-gap}
}{}
```

rule become column-rule, while prerule and postrule become HTML borders:

```
\convertcolorspec{named}{\vwcol@rulecol}{HTML}\LWR@vwcol@rulecolor%
\ifdimgreater{\vwcol@rule}{0pt}{
    \ifdimless{\vwcol@rule}{1pt}{
        \setlength{\vwcol@rule}{1pt}
    }{}
    \LWR@vwcol@addrule{column-rule}
    \LWR@vwcol@addrule{-moz-column-rule}
    \LWR@vwcol@addrule{-webkit-column-rule}
    \if@vwcol@prerule\LWR@vwcol@addrule{border-left}\fi
    \if@vwcol@postrule\LWR@vwcol@addrule{border-right}\fi
}{}
```

Each of the justify options becomes a text-align. Indentation is added where appropriate.

```
\ifdefequal{\vwcol@justify}{\RaggedRight}{
    \appto{\LWR@vwcolstyle}{text-align: left ; }
    \ifdimgreater{\vwcol@parindent}{0pt}{
        \appto{\LWR@vwcolstyle}{%
            text-indent: \LWR@printlength{\vwcol@parindent} ; %
        }
    }{}
```

```
}{}
\ifdefequal{\vwcol@justify}{\RaggedLeft}{
    \appto{\LWR@vwcolstyle}{text-align: right ; }
}{}
\ifdefequal{\vwcol@justify}{\Centering}{
    \appto{\LWR@vwcolstyle}{text-align: center ; }
}{}
\ifdefequal{\vwcol@justify}{\justifying}{
    \appto{\LWR@vwcolstyle}{text-align: justify ; }
    \ifdimgreater{\vwcol@parindent}{0pt}{
        \appto{\LWR@vwcolstyle}{%
            text-indent: \LWR@printlength{\vwcol@parindent} ; %
        }
    }{}
}{}
```

Create the with the assembled style:

```
\BlockClass[\LWR@vwcolstyle]{multicols}
}
```

When the environment ends:

```
{
\endBlockClass
}
```

File 374 **lwarp-wallpaper.sty**

§ 471 Package

wallpaper

(*Emulates or patches code by* Michael H.F. Wilkinson.)

Pkg wallpaper wallpaper is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{wallpaper}[2005/01/18]

\newcommand*{\CenterWallPaper}[2]{}
\newcommand*{\ThisCenterWallPaper}[2]{}
\newcommand*{\TileWallPaper}[3]{}
\newcommand*{\ThisTileWallPaper}[3]{}
\newcommand*{\TileSquareWallPaper}[2]{}
\newcommand*{\ThisTileSquareWallPaper}[2]{}
\newcommand*{\ULCornerWallPaper}[2]{}
\newcommand*{\ThisULCornerWallPaper}[2]{}
\newcommand*{\LLCornerWallPaper}[2]{}
\newcommand*{\ThisLLCornerWallPaper}[2]{}
\newcommand*{\URCornerWallPaper}[2]{}
\newcommand*{\ThisURCornerWallPaper}[2]{}
\newcommand*{\LRCornerWallPaper}[2]{}
\newcommand*{\ThisLRCornerWallPaper}[2]{}
\newcommand*{\ClearWallPaper}{}
\newlength{\wpXoffset}
```

```
\newlength{\wpYoffset}
```

File 375 **lwarp-watermark.sty**

§ 472 Package watermark

(*Emulates or patches code by* Alexander I. Rozhenko.)

Pkg watermark watermark is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{watermark}[2004/12/09]

\newcommand{\watermark}[1]{}
\newcommand{\leftwatermark}[1]{}
\newcommand{\rightwatermark}[1]{}
\newcommand{\thiswatermark}[1]{}
\newcommand{\thispageheading}[1]{}
```

File 376 **lwarp-widows-and-orphans.sty**

§ 473 Package widows-and-orphans

Pkg widows-and-orphans widows-and-orphans is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{widows-and-orphans}[2018/09/01]

\NewDocumentCommand\WaOsetup{m}{}
\NewDocumentCommand\WaOparameters{}{}
\NewDocumentCommand\WaOignorenext{}{}
```

File 377 **lwarp-wrapfig.sty**

§ 474 Package wrapfig

(*Emulates or patches code by* Donald Arseneau.)

Pkg wrapfig wrapfig is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{wrapfig}[2003/01/31]

\newcommand*{\LWR@wrapposition}{}

\newcommand*{\LWR@subwrapfigure}[2]{%
\renewcommand*{\LWR@wrapposition}{}%
\ifthenelse{%
    \equal{#1}{r}\OR\equal{#1}{R}\OR%
```

```
    \equal{#1}{o}\OR\equal{#1}{O}%
}%
    {\renewcommand*{\LWR@wrapposition}{float:right}}%
    {\renewcommand*{\LWR@wrapposition}{float:left}}%
\setlength{\LWR@templengthone}{#2}%
\LWR@BlockClassWP{%
    width:\LWR@printlength{\LWR@templengthone}; \LWR@wrapposition; %
    margin:10pt%
}%
{%
    width:\LWR@printlength{\LWR@templengthone}; \LWR@wrapposition%
}%
{marginblock}%
}


\NewDocumentEnvironment{wrapfigure}{o m o m}
{%
\LWR@subwrapfigure{#2}{#4}%
\captionsetup{type=figure}%
}
{%
\endLWR@BlockClassWP%
}


\NewDocumentEnvironment{wraptable}{o m o m}
{%
\LWR@subwrapfigure{#2}{#4}%
\captionsetup{type=table}%
}
{%
\endLWR@BlockClassWP%
}


\NewDocumentEnvironment{wrapfloat}{m o m o m}
{%
\LWR@subwrapfigure{#3}{#5}%
\captionsetup{type=#1}%
}
{%
\endLWR@BlockClassWP%
}

\newlength{\wrapoverhang}
```

File 378 **lwarp-xbmks.sty**

§ 475 Package xbmks

Pkg xbmks xbmks is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{xbmks}[2018/07/04]

\newcommand{\xbmksetup}[1]{}
\NewDocumentCommand{\pdfbookmarkx}{o m o m}{}
\NewDocumentCommand{\currentpdfbookmarkx}{m o m}{}
\NewDocumentCommand{\subpdfbookmarkx}{m o m}{}
\NewDocumentCommand{\belowpdfbookmarkx}{m o m}{}
```

File 379 **lwarp-xcolor.sty**

§ 476 Package

xcolor

(*Emulates or patches code by* Dr. Uwe Kern.)

Pkg xcolor xcolor is supported by lwarp.

§ 476.1 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 instead of a . These 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.

§ 476.2 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 84. 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 480.

\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 930.

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 931.

HTML lateximage: Remembers and reuses the print version.

\colorboxBlock:

Print: Becomes \colorbox.

HTML: Newly defined by lwarp-xcolor to use a , page 931.

HTML lateximage: Remembers and reuses the print version \colorbox.

\fcolorbox:

Print: Modified to allow a background of none.
\LWR@print@fcolorbox at section 84

HTML: Redefined by lwarp-xcolor, page 932.

HTML lateximage: Remembers and reuses the print version.

\fcolorboxBlock:

Print: Becomes \fcolorbox. Section 84

HTML: Newly defined by lwarp-xcolor to use a , page 932.

HTML lateximage: Remembers and reuses the print version \fcolorbox.

fcolorminipage:

Print: Newly defined in the lwarp core.
LWR@print@fcolorminipage at section 84

HTML: Newly defined by lwarp-xcolor, page 933.

HTML lateximage: Uses the print version.

\boxframe:

Print: Used as-is.

HTML: Redefined by lwarp-xcolor, page 934.

HTML lateximage: Remembers and reuses the print version.

§ 476.3 Package loading

for HTML output:

```
\LWR@ProvidesPackagePass{xcolor}[2016/05/11]
\begin{warpHTML}
```

§ 476.4 Remembering and restoring original definitions

Remember the following print-mode actions to be restored when inside a lateximage environment:

```
\LetLtxMacro\LWR@print@pagecolor\pagecolor
\LetLtxMacro\LWR@print@nopagecolor\nopagecolor
```

\LWR@restoreorigformatting Inside a lateximage the following gets restored to their print-mode actions:

```
\appto\LWR@restoreorigformatting{%
\LetLtxMacro\pagecolor\LWR@print@pagecolor%
\LetLtxMacro\nopagecolor\LWR@print@nopagecolor%
}
```

§ 476.5 HTML color style

\LWR@findcurrenttextcolor Sets \LWR@tempcolor to the current color.

```
\renewcommand*{\LWR@findcurrenttextcolor}{%
\protect\colorlet{LWR@current@color}{.}%
\protect\convertcolorspec{named}{LWR@current@color}{HTML}\LWR@tempcolor%
}
```

\LWR@currenttextcolorstyle Prints a color style for the current color.

```
\newcommand*{\LWR@currenttextcolorstyle}{%
\LWR@findcurrenttextcolor%
\ifdefstring{\LWR@tempcolor}{000000}%
{}%
{color: \LWR@origpound\LWR@tempcolor ; }%
}
```

\LWR@textcurrentcolor {⟨*text*⟩} Like \textcolor but uses the current \color instead.

```
\DeclareDocumentCommand{\LWR@textcurrentcolor}{m}{%
\begingroup%
\LWR@FBcancel%
\LWR@findcurrenttextcolor%
\InlineClass[color:\LWR@origpound\LWR@tempcolor]{textcolor}{%
    \renewcommand*{\LWR@currenttextcolor}{\LWR@origpound\LWR@tempcolor}%
    #1%
}%
\endgroup%
}
```

\LWR@colorstyle {⟨*2: model*⟩} {⟨*3: color*⟩}

For a color style, prints the color converted to HTML colors.

```
\NewDocumentCommand{\LWR@colorstyle}{m m}{%
\begingroup%
\LWR@FBcancel%
```

Use the xcolor package to convert to an HTML color space:

```
\convertcolorspec{#1}{#2}{HTML}\LWR@tempcolor%
```

Print the converted color:

```
\LWR@origpound\LWR@tempcolor%
\endgroup%
}
```

\LWR@backgroundcolor [⟨*model*⟩] {⟨*color*⟩} {⟨*text*⟩}

Similar to \textcolor, but prints black text against a color background.

Converted into an HTML hex color span.

```
\NewDocumentCommand{\LWR@backgroundcolor}{O{named} m m}{%
\begingroup%
\LWR@FBcancel%
\InlineClass[background:\LWR@colorstyle{#1}{#2}]{backgroundcolor}{%
#3%
}%
\endgroup%
}
```

§ 476.6 HTML border

\LWR@borderpadding {⟨*colorstyle*⟩} {⟨*color*⟩} Prints the HTML attributes for a black border and padding. \LWR@forceminwidth must be used first in order to set the border width.

```
\newcommand*{\LWR@borderpadding}[2]{%
border:\LWR@printlength{\LWR@atleastonept} solid \LWR@colorstyle{#1}{#2} ; %
padding:\LWR@printlength{\fboxsep}%
}
```

§ 476.7 High-level macros

\textcolor [⟨*model*⟩] {⟨*color*⟩} {⟨*text*⟩}

Converted into an HTML hex color span.

```
\NewDocumentCommand{\LWR@HTML@textcolor}{o m m}{%
\begingroup%
```

Set the PDF color, to be picked up by SVG math if possible.

The print-mode \color command cannot accept the named option with color mixing, but it works with no option at all.

```
\IfValueTF{#1}{%
    \color[#1]{#2}%
}{%
    \color{#2}%
}%

\LWR@FBcancel%
\IfValueTF{#1}{%
\InlineClass[color:\LWR@colorstyle{#1}{#2}]{textcolor}{%
\renewcommand*{\LWR@currenttextcolor}{\LWR@origpound\LWR@tempcolor}%
#3%
}%
}{%
\InlineClass[color:\LWR@colorstyle{named}{#2}]{textcolor}{%
\renewcommand*{\LWR@currenttextcolor}{\LWR@origpound\LWR@tempcolor}%
#3%
}%
}%
\endgroup%
}

```

```
\LWR@formatted{textcolor}
```

\pagecolor [⟨*model*⟩] {⟨*color*⟩}

Ignored. Use css instead.

```
\renewcommand*{\pagecolor}[2][named]{}
```

\nopagecolor Ignored.

```
\renewcommand*{\nopagecolor}{}
```

\colorbox [⟨*model*⟩] {⟨*color*⟩} {⟨*text*⟩}

Converted into an HTML hex background color .

```
\NewDocumentCommand{\LWR@HTML@colorbox}{O{named} m +m}{%
\begingroup%
\LWR@FBcancel%
\InlineClass[%
background:\LWR@colorstyle{#1}{#2} ; %
padding:\LWR@printlength{\fboxsep}%
]{colorbox}{#3}%
\endgroup%
}

\AtBeginDocument{
\LWR@formatted{colorbox}
}
```

\colorboxBlock [⟨*model*⟩] {⟨*color*⟩} {⟨*text*⟩}

Converted into an HTML hex background color .

```
\NewDocumentCommand{\LWR@HTML@colorboxBlock}{O{named} m +m}{%
\begingroup%
\LWR@FBcancel%

\LWR@stoppars%

\begin{BlockClass}[%
background:\LWR@colorstyle{#1}{#2} ; %
padding:\LWR@printlength{\fboxsep}%
]{colorboxBlock}
#3
\end{BlockClass}%
\endgroup%
```

Prevent paragraph tags around horizontal white space until the start of the next paragraph:

```
\global\booltrue{LWR@minipagethispar}%
}

\AtBeginDocument{
\LWR@formatted{colorboxBlock}
}
```

\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.

```
\NewDocumentCommand{\LWR@HTML@fcolorbox}{O{named} m O{named} m +m}{%
\LWR@traceinfo{HTML fcolorbox #2 #4}%
\begingroup%
\LWR@FBcancel%
\LWR@forceminwidth{\fboxrule}%
\ifthenelse{\equal{#4}{none}}%
    {% no background color
        \InlineClass[%
        \LWR@borderpadding{#1}{#2}%
        ]{fcolorbox}{#5}%
    }%
    {% yes background color
        \InlineClass[%
        \LWR@borderpadding{#1}{#2} ; %
        background:\LWR@colorstyle{#3}{#4}%
        ]{fcolorbox}{#5}%
    }%
\endgroup%
}

\AtBeginDocument{
\LWR@formatted{fcolorbox}
}
```

\fcolorboxBlock `[`⟨*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.

```
\NewDocumentCommand{\LWR@HTML@fcolorboxBlock}{O{named} m O{named} m +m}{%
\LWR@traceinfo{HTML fcolorboxBlock #2 #4}%
\begingroup%
\LWR@FBcancel%
\LWR@forceminwidth{\fboxrule}%

\LWR@stoppars%
```

```
\ifthenelse{\equal{#4}{none}}%
    {% no background color
        \begin{BlockClass}[%
            \LWR@borderpadding{#1}{#2}%
        ]{fcolorboxBlock}
        #5
        \end{BlockClass}%
    }%
    {% yes background color
        \convertcolorspec{#3}{#4}{HTML}\LWR@tempcolortwo%
        \begin{BlockClass}[%
            background:\LWR@origpound\LWR@tempcolortwo\ ; %
            \LWR@borderpadding{#1}{#2}%
        ]{fcolorboxBlock}
        #5
        \end{BlockClass}%
    }%
\endgroup%
```

Prevent paragraph tags around horizontal white space until the start of the next paragraph:

```
\global\booltrue{LWR@minipagethispar}%
\LWR@traceinfo{HTML fcolorboxBlock done}%
}

\AtBeginDocument{
\LWR@formatted{fcolorboxBlock}
}
```

Creates a framed HTML around its contents.

A print-output version is defined in the lwarp core: section 84

\LWR@subfcolorminipage {⟨*framemodel*⟩} {⟨*framecolor*⟩} {⟨*background tag*⟩} {⟨*height*⟩}

```
\NewDocumentCommand{\LWR@subfcolorminipage}{m m m m}{%

\LWR@stoppars%

\begin{BlockClass}[%
#3%
\LWR@borderpadding{#1}{#2} ; %
\IfValueT{#4}{height:\LWR@printlength{\LWR@tempheight} ; }%
width:\LWR@printlength{\LWR@tempwidth}%
]{fcolorminipage}%
}
```

Env fcolorminipage [⟨*1:framemodel*⟩] {⟨*2:framecolor*⟩} [⟨*3:boxmodel*⟩] {⟨*4:boxcolor*⟩} [⟨*5:align*⟩] [⟨*6:height*⟩] [⟨*7:inner-align*⟩] {⟨*8:width*⟩}

```
\NewDocumentEnvironment{LWR@HTML@fcolorminipage}{O{named} m O{named} m O{c} o o m}
{%
```

```
\LWR@FBcancel%
\setlength{\LWR@tempwidth}{#8}%
\IfValueT{#6}{\setlength{\LWR@tempheight}{#6}}%
\LWR@forceminwidth{\fboxrule}%
\convertcolorspec{#1}{#2}{HTML}\LWR@tempcolor%
\ifthenelse{\equal{#4}{none}}%
    {\LWR@subfcolorminipage{#1}{#2}{}{#6}}%
    {%
        \convertcolorspec{#3}{#4}{HTML}\LWR@tempcolortwo%
        \LWR@subfcolorminipage{#1}{#2}%
            {background:\LWR@origpound\LWR@tempcolortwo\ ; }%
            {#6}%
    }%
}%
{%
    \end{BlockClass}%
```

Prevent paragraph tags around horizontal white space until the start of the next paragraph:

```
    \global\booltrue{LWR@minipagethispar}%
}

\AtBeginDocument{
\LWR@formattedenv{fcolorminipage}
}
```

`\boxframe` {⟨*width*⟩} {⟨*height*⟩} {⟨*depth*⟩}

The depth is added to the height, but the box is not decended below by the depth. `\textcolor` is honored.

```
\newcommand*{\LWR@HTML@boxframe}[3]{%
{%
\setlength{\LWR@tempwidth}{#1}%
\setlength{\LWR@tempheight}{#2}%
\addtolength{\LWR@tempheight}{#3}%
\LWR@forceminwidth{\fboxrule}%
\LWR@findcurrenttextcolor%
\InlineClass[%
display:inline-block ; %
border:\LWR@printlength{\LWR@atleastonept} solid \LWR@currenttextcolor{} ; %
width:\LWR@printlength{\LWR@tempwidth} ; %
height:\LWR@printlength{\LWR@tempheight}%
]{boxframe}{}%
}%
}

\LWR@formatted{boxframe}
```

§ 476.8 Row colors

\rowc@l@rs [⟨*cmds*⟩] {⟨*startrow*⟩} {⟨*odd color*⟩} {⟨*even color*⟩}

```
\newcommand*{\LWR@xcolortempcolor}{}

\def\rowc@l@rs[#1]#2#3#4%
{
\global\rownum=1
   \global\@rowcolorstrue
   \@ifxempty{#3}%
     {\def\@oddrowcolor{\@norowcolor}}%
     {%
        \convertcolorspec{named}{#3}{HTML}\LWR@xcolortempcolor%
        \edef\@oddrowcolor{%
            \csdef{LWR@xcolorrowHTMLcolor}{\LWR@xcolortempcolor}%
        }%
     }%
   \@ifxempty{#4}%
     {\def\@evenrowcolor{\@norowcolor}}%
     {%
        \convertcolorspec{named}{#4}{HTML}\LWR@xcolortempcolor%
        \edef\@evenrowcolor{%
            \csdef{LWR@xcolorrowHTMLcolor}{\LWR@xcolortempcolor}%
        }%
     }%
   \if@rowcmd
     \def\@rowcolors
     {%
%          #1%
        \if@rowcolors
%         \noalign{%
            \relax\ifnum\rownum<#2\@norowcolor\else
            \ifodd\rownum\@oddrowcolor\else\@evenrowcolor\fi\fi%
%          }%
        \fi%
     }%
   \else
     \def\@rowcolors
     {%
        \if@rowcolors
            \ifnum\rownum<#2%
%            \noalign{%
                \@norowcolor
%             }
            \else
%            #1%
%            \noalign{%
                \ifodd\rownum\@oddrowcolor\else\@evenrowcolor\fi%
%             }%
            \fi
        \fi%
     }%
   \fi
   \ignorespaces%
}
```

\@norowcolor Turns off color for this row.

```
\def\@norowcolor{%
\renewcommand{\LWR@xcolorrowHTMLcolor}{}%
}
```

\@rowc@lors Executed at the end of each row.

```
\def\@rowc@lors{%
%   \noalign{%
        \global\advance\rownum\@ne%
%   }%
    \@rowcolors%
}

\end{warpHTML}
```

File 380 **lwarp-xechangebar.sty**

§ 477 Package xechangebar

Pkg xechangebar xechangebar is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{xechangebar}[2017/08/03]
\LWR@origRequirePackage{lwarp-changebar}
```

File 381 **lwarp-xellipsis.sty**

§ 478 Package xellipsis

(*Emulates or patches code by* Donald P. Goodman III.)

Pkg xellipsis 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:

```
\LWR@ProvidesPackagePass{xellipsis}[2015/11/01]

\newcommand*{\LWR@xellipsespace}[1]{%
\ifdim#1=0pt\else%
    \ifdim#1<\fontdimen2\font%
        \,%
    \else%
        ~%
    \fi%
\fi%
}

```

```
\def\xelip{%
\mbox{%
    \LWR@xellipsespace{\xelipprebef}%
    \xelipprechar%
    \LWR@xellipsespace{\xelippreaft}%
    \LWR@xellipsespace{\xelipbef}%
    \xelipchar%
    \xel@loopi = 1%
    \loop\ifnum\xelipnum>\xel@loopi%
        \advance\xel@loopi by1%
        \LWR@xellipsespace{\xelipgap}%
        \xelipchar%
    \repeat%
    \LWR@xellipsespace{\xelipaft}%
    \LWR@xellipsespace{\xelippostbef}%
    \xelippostchar%
    \LWR@xellipsespace{\xelippostaft}%
}%
}%
```

File 382 **lwarp-xetexko-vertical.sty**

§ 479 Package xetexko-vertical

(*Emulates or patches code by* Dohyun Kim.)

Pkg xetexko-vertical xetexko-vertical is patched for use by lwarp.

for HTML output:

```
\LWR@loadbefore{xetexko-vertical}

\LWR@ProvidesPackagePass{xetexko-vertical}[2018/04/06]

\renewcommand{\verticaltypesetting}{}
\renewenvironment{vertical}[1]{\BlockClass{verticalrl}}{\endBlockClass}
\renewenvironment{horizontal}[1]{\BlockClass{horizontaltb}}{\endBlockClass}
\renewcommand{\vertlatin}[1]{#1}
```

File 383 **lwarp-xfakebold.sty**

§ 480 Package xfakebold

(*Emulates or patches code by* Herbert Voss.)

Pkg xfakebold xfakebold is patched for use by lwarp, and additional underlying support is found in the lwarp core.

⚠ page breaks Note that the print version resets to unbold at each page break, whereas the HTML version maintains the bold state until it is undone.

for HTML output:

```
\LWR@ProvidesPackagePass{xfakebold}[2018/07/25]
```

```
\let\LWR@orig@setBold\setBold
\let\LWR@orig@unsetBold\unsetBold
\renewcommand*{\setBold}{\booltrue{LWR@xfakebold}}
\renewcommand*{\unsetBold}{\boolfalse{LWR@xfakebold}}

\renewcommand*{\LWR@applyxfakebold}{%
    \ifbool{LWR@xfakebold}{\LWR@orig@setBold}{\LWR@orig@unsetBold}%
}
```

File 384 **lwarp-xfrac.sty**

§ 481 Package

xfrac

(*Emulates or patches code by* The LaTeX3 Project.)

Pkg xfrac Supported by adding xfrac instances.

for HTML output:

```
\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.

for HTML & PRINT:

```
\begin{warpall}
```

\xfracHTMLfontsize User-redefinable macro which controls the font size of the fraction.

```
\newcommand*{\xfracHTMLfontsize}{.6em}
```

```
\end{warpall}
```

for HTML output:

```
\begin{warpHTML}
```

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.

```
\DeclareInstance{xfrac}{default}{text}{
    numerator-format = {%
        \begingroup%
        \RenewDocumentCommand{\scalebox}{m o m}{##3}%
        \InlineClass{numerator}{#1}\,%
        \endgroup%
    },
    denominator-format = {%
        \begingroup%
        \RenewDocumentCommand{\scalebox}{m o m}{##3}%
        \InlineClass{denominator}{#1}%
        \endgroup%
    },
```

For *pdftotext*, do not scale the text:

```
    scaling = false
}

\DeclareInstance{xfrac}{lmr}{text}{
    numerator-format = {%
        \begingroup%
        \RenewDocumentCommand{\scalebox}{m o m}{##3}%
        \InlineClass{numerator}{#1}\,%
        \endgroup%
    },
    denominator-format = {%
        \begingroup%
        \RenewDocumentCommand{\scalebox}{m o m}{##3}%
        \InlineClass{denominator}{#1}%
        \endgroup%
    },
```

For *pdftotext*, do not scale the text:

```
    scaling = false
}

\DeclareInstance{xfrac}{lmss}{text}{
    numerator-format = {%
        \begingroup%
        \RenewDocumentCommand{\scalebox}{m o m}{##3}%
        \InlineClass{numerator}{#1}\,%
        \endgroup%
    },
    denominator-format = {%
        \begingroup%
        \RenewDocumentCommand{\scalebox}{m o m}{##3}%
        \InlineClass{denominator}{#1}%
        \endgroup%
    },
```

For *pdftotext*, do not scale the text:

```
    scaling = false
```

```
}

\DeclareInstance{xfrac}{lmtt}{text}{
    numerator-format = {%
        \begingroup%
        \RenewDocumentCommand{\scalebox}{m o m}{##3}%
        \InlineClass{numerator}{#1}\,%
        \endgroup%
    },
    denominator-format = {%
        \begingroup%
        \RenewDocumentCommand{\scalebox}{m o m}{##3}%
        \InlineClass{denominator}{#1}%
        \endgroup%
    },
```

For *pdftotext*, do not scale the text:

```
    scaling = false
}


\end{warpHTML}
```

File 385 **lwarp-xltabular.sty**

§ 482 Package

xltabular

(*Emulates or patches code by* Rolf Niepraschk, Herbert Voss.)

Pkg xltabular 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}}
```

```
\RequirePackage{tabularx}

\LWR@ProvidesPackageDrop{xltabular}[2018/05/23]

\DeclareDocumentEnvironment{xltabular}{o m m}
{\longtable{#3}}
{\endlongtable}
```

File 386 **lwarp-xltxtra.sty**

§ 483 Package xltxtra

(*Emulates or patches code by* Will Robertson, Jonathan Kew.)

Pkg xltxtra xltxtra is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{xltxtra}[2016/01/21]

\RequirePackage{realscripts}
\RequirePackage{metalogo}
\newcommand*\TeX@logo@spacing[6]{}

\newcommand*{\vfrac}[2]{%
\textsuperscript{#1}/\textsubscript{#2}%
}

\newcommand\namedglyph[1]{%
  \@tempcnta=\XeTeXglyphindex "#1"\relax
  \ifnum\@tempcnta>0
    \XeTeXglyph\@tempcnta
  \else
    \xxt@namedglyph@fallback{#1}%
  \fi}

\newcommand\xxt@namedglyph@fallback[1]{[#1]}

\DeclareDocumentCommand{\showhyphens}{m}{}
```

File 387 **lwarp-xmpincl.sty**

§ 484 Package xmpincl

(*Emulates or patches code by* Maarten Sneep.)

Pkg xmpincl Emulated.

for HTML output: Discard all options for lwarp-xmpincl:

```
\LWR@ProvidesPackageDrop{xmpincl}[2008/05/10]

\newcommand*{\includexmp}[1]{}
```

File 388 **lwarp-xpiano.sty**

§ 485 Package xpiano

(*Emulates or patches code by* Enrico Gregorio.)

Pkg xpiano xpiano is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{xpiano}

\ExplSyntaxOn
\NewDocumentCommand{\LWR@print@keyboard}{ O{}m }
{
\xpiano_keyboard:nn { #1 } { #2 }
}

\NewDocumentCommand{\LWR@HTML@keyboard}{ O{}m }
{
\begin{lateximage}*
    [(-xpiano-~\packagediagramname{}: \detokenize\expandafter{#2})]
    [\detokenize\expandafter{#1}]
\xpiano_keyboard:nn { #1 } { #2 }
\end{lateximage}
}
\ExplSyntaxOff

\LWR@formatted{keyboard}
```

File 389 **lwarp-xpinyin.sty**

§ 486 Package xpinyin

(*Emulates or patches code by* Soben Lee.)

Pkg xpinyin xpinyin is partly supported. `\xpinyin` and `pinyinscope` are nullified, but `\pinyin` works.

for HTML output:

```
\LWR@ProvidesPackagePass{xpinyin}[2018/01/28]

\RenewDocumentEnvironment{pinyinscope}{O{}}{}{}

\RenewDocumentCommand{\xpinyin}{s O{} m}{%
    \IfBooleanTF{#1}{#3}{\@firstoftwo#3}%
}

\RenewDocumentCommand{\enablepinyin}{}{}
```

File 390 **lwarp-xtab.sty**

§ 487 Package **xtab**

(*Emulates or patches code by* Peter Wilson.)

Pkg xtab xtab is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{xtab}[2011/07/31]
```

⚠ Misplaced alignment tab character &

For `\tablefirsthead`, etc., enclose them as follows:

```
\StartDefiningTabulars
\tablefirsthead
. . .
\StopDefiningTabulars
```

See section 9.10.

⚠ lateximage supertabular and xtab are not supported inside a `lateximage`.

```
\newcommand{\LWRXT@firsthead}{}

\newcommand{\tablefirsthead}[1]{%
    \long\gdef\LWRXT@firsthead{#1}%
}

\newcommand{\tablehead}[1]{}

\newcommand{\tablelasthead}[1]{}

\newcommand{\notablelasthead}{}

\newcommand{\tabletail}[1]{}

\newcommand{\LWRXT@lasttail}{}

\newcommand{\tablelasttail}[1]{%
    \long\gdef\LWRXT@lasttail{#1}%
}


\newcommand{\tablecaption}[2][]{%
    \long\gdef\LWRXT@caption{%
        \ifblank{#1}%
            {\caption{#2}}%
            {\caption[#1]{#2}}%
    }%
}

\let\topcaption\tablecaption
\let\bottomcaption\tablecaption
```

```
\newcommand*{\LWRXT@caption}{}

\newcommand*{\shrinkheight}[1]{}

\newcommand*{\xentrystretch}[1]{}

\NewDocumentEnvironment{xtabular}{s o m}
{%
\LWR@traceinfo{xtabular}%
\table%
\LWRXT@caption%
\begin{tabular}{#3}%
\TabularMacro\ifdefvoid{\LWRXT@firsthead}%
{\LWR@getmynexttoken}%
{\expandafter\LWR@getmynexttoken\LWRXT@firsthead}%
}%
{%
\ifdefvoid{\LWRXT@lasttail}%
{}%
{%
\TabularMacro\ResumeTabular%
\LWRXT@lasttail%
}%
\end{tabular}%
\endtable%
\LWR@traceinfo{xtabular done}%
}

\NewDocumentEnvironment{mpxtabular}{s o m}
{\minipage{\linewidth}\xtabular{#3}}
{\endxtabular\endminipage}
```

File 391 **lwarp-xunicode.sty**

§ 488 Package

xunicode

Pkg xunicode

Error if xunicode is loaded after lwarp.

Patch lwarp-xunicode, but also verify that is was loaded before lwarp:

for HTML output:

```
\LWR@loadbefore{xunicode}%

\LWR@ProvidesPackagePass{xunicode}[2011/09/09]
```

\textcircled becomes a span with a rounded border. \providecommand is used to avoid conflict with textcomp.

```
\providecommand*{\LWR@HTML@textcircled}[1]{%
    \InlineClass[border: 1px solid \LWR@currenttextcolor]{textcircled}{#1}%
}

\LWR@formatted{textcircled}
```

Nullify textcomp macros when generating filenames:

```
\FilenameNullify{%
    \renewcommand*{\textdegree}{}%
    \renewcommand*{\textcelsius}{}%
    \renewcommand*{\textohm}{}%
    \renewcommand*{\textmu}{}%
    \renewcommand*{\textlquill}{}%
    \renewcommand*{\textrquill}{}%
    \renewcommand*{\textcircledP}{}%
    \renewcommand*{\texttwelveudash}{}%
    \renewcommand*{\textthreequartersemdash}{}%
    \renewcommand*{\textmho}{}%
    \renewcommand*{\textnaira}{}%
    \renewcommand*{\textpeso}{}%
    \renewcommand*{\textrecipe}{}%
    \renewcommand*{\textinterrobangdown}{}%
    \renewcommand*{\textperthousand}{}%
    \renewcommand*{\textpertenthousand}{}%
    \renewcommand*{\textbaht}{}%
    \renewcommand*{\textdiscount}{}%
    \renewcommand*{\textservicemark}{}%
    \renewcommand*{\textcircled}[1]{#1}%
    \renewcommand*{\capitalcedilla}[1]{#1}%
    \renewcommand*{\capitalogonek}[1]{#1}%
    \renewcommand*{\capitalgrave}[1]{#1}%
    \renewcommand*{\capitalacute}[1]{#1}%
    \renewcommand*{\capitalcircumflex}[1]{#1}%
    \renewcommand*{\capitaltilde}[1]{#1}%
    \renewcommand*{\capitaldieresis}[1]{#1}%
    \renewcommand*{\capitalhungarumlaut}[1]{#1}%
    \renewcommand*{\capitalring}[1]{#1}%
    \renewcommand*{\capitalcaron}[1]{#1}%
    \renewcommand*{\capitalbreve}[1]{#1}%
    \renewcommand*{\capitalmacron}[1]{#1}%
    \renewcommand*{\capitaldotaccent}[1]{#1}%
}% FilenameNullify
```

File 392 **lwarp-xurl.sty**

§ 489 Package xurl

Pkg xurl xurl is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{xurl}[2018/06/02]
```

File 393 **lwarp-xy.sty**

§ 490 Package xy

(*Emulates or patches code by* Kristoffer H. Rose, Ross Moore.)

Pkg xy xy is patched for use by lwarp.

⚠ \xypolygon \xypolygon must be used inside the xy environment, or inside \xy ... \endxy.

for HTML output:

```
\LWR@ProvidesPackagePass{xy}[2013/10/06]

\AtBeginDocument{

\preto{\xy}{\begin{lateximage}[(-xy-~\packagediagramname)]}
\appto{\endxy}{\end{lateximage}}

\@ifundefined{xymatrix}{}{
\LetLtxMacro\LWR@origxymatrix\xymatrix

\renewcommand{\xymatrix}[1]{%
\begin{lateximage}[(-xy- xymatrix \packagediagramname)]
\LWR@origxymatrix{#1}
\end{lateximage}
}
}

\@ifundefined{xygraph}{}{
\LetLtxMacro\LWR@origxygraph\xygraph

\renewcommand{\xygraph}[1]{%
\begin{lateximage}[(-xy- xygraph \packagediagramname)]
\LWR@origxygraph{#1}
\end{lateximage}
}
}

}
```

File 394 **lwarp-zhlineskip.sty**

§ 491 Package zhlineskip

Pkg zhlineskip zhlineskip is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{zhlineskip}[2018/11/30]

\newcommand*\SetTextEnvironmentSinglespace[1]{}
```

```
\newcommand*\RestoreTextEnvironmentLeading{}
\newcommand*\SetMathEnvironmentSinglespace[1]{}
\newcommand*\RestoreMathEnvironmentLeading{}
```

File 395 **lwarp-zwpagelayout.sty**

§ 492 Package zwpagelayout

(*Emulates or patches code by* Zdeněk Wagner.)

Pkg zwpagelayout zwpagelayout is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{zwpagelayout}[2013/01/13]

\def\noBboxes{}
\@onlypreamble\noBboxes

\expandafter\ifx\csname definecolor\endcsname\relax \else
  \definecolor{cmykblack}{cmyk}{0,0,0,1}
  \definecolor{grblack}{gray}{0}
%   \ifzwpl@redefineblack
%     \definecolor{black}{cmyk}{0,0,0,1}\color{black}
%   \fi
  \definecolor{cmykred}{cmyk}{0,1,1,0}
  \definecolor{cmykgreen}{cmyk}{1,0,1,0}
  \definecolor{cmykblue}{cmyk}{1,1,0,0}
  \definecolor{rgbred}{rgb}{1,0,0}
  \definecolor{rgbgreen}{rgb}{0,1,0}
  \definecolor{rgbblue}{rgb}{0,0,1}
%   \ifzwpl@redefinetocmyk
%     \definecolor{red}{cmyk}{0,1,1,0}
%     \definecolor{green}{cmyk}{1,0,1,0}
%     \definecolor{blue}{cmyk}{1,1,0,0}
%   \fi
\fi

\let\OverprintXeTeXExtGState\relax

\DeclareRobustCommand\SetOverprint{\ignorespaces}
\DeclareRobustCommand\SetKnockout{\ignorespaces}
\DeclareRobustCommand\textoverprint[1]{{\SetOverprint#1}}
\DeclareRobustCommand\textknockout[1]{{\SetKnockout#1}}

\def\SetPDFminorversion#1{}
\@onlypreamble\SetPDFminorversion

\newcommand*\Vcorr{}

\DeclareRobustCommand\vb[1][]{}
\NewDocumentCommand{\NewOddPage}{* o}{}
\NewDocumentCommand{\NewEvenPage}{* o}{}
\def\SetOddPageMessage#{\gdef\ZW@oddwarning}
\def\SetEvenPageMessage#{\gdef\Z@@evenwarning}
```

```
\def\ZW@oddwarning{Empty page inserted}\let\ZW@evenwarning\ZW@oddwarning

\def\clap#1{#1}

\def\CropFlap{2in}
\def\CropSpine{1in}
\def\CropXSpine{1in}
\def\CropXtrim{.25in}
\def\CropYtrim{.25in}
\def\UserWidth{5in}
\def\UserLeftMargin{1in}
\def\UserRightMargin{1in}
\def\UserTopMargin{1in}
\def\UserBotMargin{1in}
\def\thePageNumber{\LWR@origpound\,\arabic{page}}
\ifXeTeX
\def\ifcaseZWdriver{\ifcase2}
\else
\def\ifcaseZWdriver{\ifcase1}
\fi
\DeclareRobustCommand\ZWifdriver[2]{}
```

File 396 **lwarp-patch-komascript.sty**

§ 493 Package

patch-komascript

Pkg lwarp-patch-komascript

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.

\titlehead, \subject, \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:

```
\ProvidesPackage{lwarp-patch-komascript}
```

typearea is emulated.

```
\RequirePackage{lwarp-typearea}
```

tocbasic is emulated.

```
\RequirePackage{lwarp-tocbasic}
```

scrextend patches most of the new macros.

```
\RequirePackage{lwarp-scrextend}
```

Indexing macros, simplified for lwarp:

```
\AtBeginDocument{

\renewcommand*{\idx@heading}{%
  \idx@@heading{\indexname}%
}

\renewenvironment{theindex}{%
  \idx@heading%
  \index@preamble\par\nobreak
    \let\item\LWR@indexitem%
    \let\subitem\LWR@indexsubitem%
    \let\subsubitem\LWR@indexsubsubitem%
}
{}

\renewcommand*\indexspace{}

}% AtBeginDocument
```

The \minisec is placed inside a of class minisec.

```
\renewcommand*{\minisec}[1]{
\begin{BlockClass}{minisec}
#1
\end{BlockClass}
}
```

The part and chapter preambles are placed as plain text just after each heading.

```
\@ifundefined{setpartpreamble}{}{
\RenewDocumentCommand{\setpartpreamble}{o o +m}{%
\renewcommand{\part@preamble}{#3}%
}
}

\@ifundefined{setchapterpreamble}{}{
\RenewDocumentCommand{\setchapterpreamble}{o o +m}{%
\renewcommand{\chapter@preamble}{#3}%
}
}
```

Simple captions are used in all cases.

```
\LetLtxMacro\captionbelow\caption
\LetLtxMacro\captionabove\caption

\LetLtxMacro\captionofbelow\captionof
\LetLtxMacro\captionofabove\captionof

\RenewDocumentEnvironment{captionbeside}{o m o o o s}
{}
```

```
{%
\IfValueTF{#1}%
{\caption[#1]{#2}}%
{\caption{#2}}%
}

\RenewDocumentEnvironment{captionofbeside}{m o m o o o s}
{}
{%
\IfValueTF{#2}%
{\captionof{#1}[#2]{#3}}%
{\captionof{#1}{#3}}%
}

\RenewDocumentCommand{\setcapindent}{s m}{}
\renewcommand*{\setcaphanging}{}
\renewcommand*{\setcapwidth}[2][]{}
\renewcommand*{\setcapdynwidth}[2][]{}
\RenewDocumentCommand{\setcapmargin}{s o m}{}
```

File 397 **lwarp-patch-memoir.sty**

§ 494 Package

patch-memoir

(*Emulates or patches code by* Peter Wilson.)

Pkg lwarp-patch-memoir

Patches for memoir class.

⚠ Not fully tested!

Please send bug reports!

lwarp loads this package when the memoir class is detected.

⚠ options clash

While emulating memoir, lwarp pre-loads a number of packages (section 494.1). This can cause an options clash when the user's document later loads the same packages with options. To fix this problem, specify the options before loading lwarp:

```
\documentclass{memoir}
. . .
\PassOptionsToPackage{options_list}{package_name}
. . .
\usepackage{lwarp}
. . .
\usepackage{package_name}
```

⚠ version numbers

memoir emulates a number of packages, and declares a version date for each which often does not match the date of the corresponding freestanding package. This can cause warnings about incorrect version numbers. Since lwarp is intended to support the freestanding packages, which are often newer than the date declared by memoir, it is hoped that memoir will update and change its emulated version numbers to match.

`\verbfootnote` is not supported.

`\newfootnoteseries`, etc. are not supported.

lwarp loads pagenote to perform memoir's pagenote functions, but there are minor differences in \pagenotesubhead and related macros.

Poem numbering is not supported.

The verbatim environment does not yet support the memoir enhancements. It is currently recommended to load and use fancyvrb instead.

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:

```
\ProvidesPackage{lwarp-patch-memoir}
```

§ 494.1 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.

```
\RequirePackage{lwarp-abstract}% req'd
\RequirePackage{lwarp-array}% req'd
\RequirePackage{lwarp-booktabs}% req'd
% \RequirePackage{lwarp-ccaption}% emualated below
\RequirePackage{lwarp-changepage}% req'd
\RequirePackage{lwarp-crop}
\RequirePackage{lwarp-dcolumn}% req'd
\RequirePackage{lwarp-enumerate}% req'd
\RequirePackage{lwarp-epigraph}% req'd
\RequirePackage{lwarp-fancyvrb}% req'd
\RequirePackage{lwarp-footmisc}% req'd
\RequirePackage{lwarp-framed}% req'd
\RequirePackage{lwarp-hanging}% req'd
\RequirePackage{lwarp-makeidx}% req'd
\DisemulatePackage{moreverb}
\RequirePackage{lwarp-moreverb}
\RequirePackage{lwarp-mparhack}
\RequirePackage{lwarp-needspace}% req'd
\RequirePackage{lwarp-nextpage}% req'd
\RequirePackage{lwarp-pagenote}% req'd
\RequirePackage{lwarp-parskip}
\RequirePackage{lwarp-setspace}% req'd
\RequirePackage{lwarp-showidx}
\RequirePackage{lwarp-subfigure}% red'q
\makeindex
```

subfigure is emulated via subfig, which pre-defines subfigure and subtable, but memoir does not, so they must be tested for here:

```
\LetLtxMacro\LWR@memorignewsubfloat\newsubfloat
\RenewDocumentCommand{\newsubfloat}{O{} m}{%
    \@ifundefined{c@sub#2}{%
        \LWR@memorignewsubfloat[#1]{#2}%
```

```
    }{}%
}

\RequirePackage{lwarp-tabularx}% req'd
\RequirePackage{lwarp-titling}% req'd
% \RequirePackage{lwarp-tocbibind}% not emulated by memoir
\RequirePackage{lwarp-tocloft}% req'd
\RequirePackage{lwarp-verse}% req'd
```

§ 494.2 Preliminary setup

Bypass the memoir package mechanism:

```
\LetLtxMacro\LWR@orig@label\@mem@old@label
```

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.

```
\setstocksize{190in}{20in}
\setlrmarginsandblock{2in}{2in}{*}
\setulmarginsandblock{1in}{1in}{*}
```

§ 494.3 Laying out the page

```
\renewcommand*{\stockavi}{}
\renewcommand*{\stockav}{}
\renewcommand*{\stockaiv}{}
\renewcommand*{\stockaiii}{}
\renewcommand*{\stockbvi}{}
\renewcommand*{\stockbv}{}
\renewcommand*{\stockbiv}{}
\renewcommand*{\stockbiii}{}
% \renewcommand*{\stockmetriccrownvo}{}% in docs but not in the package
\renewcommand*{\stockmlargecrownvo}{}
\renewcommand*{\stockmdemyvo}{}
\renewcommand*{\stockmsmallroyalvo}{}
\renewcommand*{\pageavi}{}
\renewcommand*{\pageav}{}
\renewcommand*{\pageaiv}{}
\renewcommand*{\pageaiii}{}
\renewcommand*{\pagebvi}{}
\renewcommand*{\pagebv}{}
\renewcommand*{\pagebiv}{}
\renewcommand*{\pagebiii}{}
% \renewcommand*{\pagemetriccrownvo}{}% in docs but not in the package
\renewcommand*{\pagemlargecrownvo}{}
\renewcommand*{\pagemdemyvo}{}
\renewcommand*{\pagemsmallroyalvo}{}

\renewcommand*{\stockdbill}{}
\renewcommand*{\stockstatement}{}
\renewcommand*{\stockexecutive}{}
\renewcommand*{\stockletter}{}
\renewcommand*{\stockold}{}
\renewcommand*{\stocklegal}{}
```

```
\renewcommand*{\stockledger}{}
\renewcommand*{\stockbroadsheet}{}
\renewcommand*{\pagedbill}{}
\renewcommand*{\pagestatement}{}
\renewcommand*{\pageexecutive}{}
\renewcommand*{\pageletter}{}
\renewcommand*{\pageold}{}
\renewcommand*{\pagelegal}{}
\renewcommand*{\pageledger}{}
\renewcommand*{\pagebroadsheet}{}

\renewcommand*{\stockpottvo}{}
\renewcommand*{\stockfoolscapvo}{}
\renewcommand*{\stockcrownvo}{}
\renewcommand*{\stockpostvo}{}
\renewcommand*{\stocklargecrownvo}{}
\renewcommand*{\stocklargepostvo}{}
\renewcommand*{\stocksmalldemyvo}{}
\renewcommand*{\stockdemyvo}{}
\renewcommand*{\stockmediumvo}{}
\renewcommand*{\stocksmallroyalvo}{}
\renewcommand*{\stockroyalvo}{}
\renewcommand*{\stocksuperroyalvo}{}
\renewcommand*{\stockimperialvo}{}
\renewcommand*{\pagepottvo}{}
\renewcommand*{\pagefoolscapvo}{}
\renewcommand*{\pagecrownvo}{}
\renewcommand*{\pagepostvo}{}
\renewcommand*{\pagelargecrownvo}{}
\renewcommand*{\pagelargepostvo}{}
\renewcommand*{\pagesmalldemyvo}{}
\renewcommand*{\pagedemyvo}{}
\renewcommand*{\pagemediumvo}{}
\renewcommand*{\pagesmallroyalvo}{}
\renewcommand*{\pageroyalvo}{}
\renewcommand*{\pagesuperroyalvo}{}
\renewcommand*{\pageimperialvo}{}

\renewcommand*{\memfontfamily}{}
\renewcommand*{\memfontenc}{}
\renewcommand*{\memfontpack}{}

\renewcommand*{\anyptfilebase}{}
\renewcommand*{\anyptsize}{10}

\renewcommand*{\setstocksize}[2]{}
\renewcommand*{\settrimmedsize}[3]{}
\renewcommand*{\settrims}[2]{}

% \newlength{\lxvchars}
% \setlength{\lxvchars}{305pt}
% \newlength{\xlvchars}
% \setlength{\xlvchars}{190pt}
\renewcommand*{\setxlvchars}[1]{}
\renewcommand*{\setlxvchars}[1]{}
```

```
129
130 \renewcommand*{\settypeblocksize}[3]{}
131 \renewcommand*{\setlrmargins}[3]{}
132 \renewcommand*{\setlrmarginsandblock}[3]{}
133 \renewcommand*{\setbinding}[1]{}
134 \renewcommand*{\setulmargins}[3]{}
135 \renewcommand*{\setulmarginsandblock}[3]{}
136 \renewcommand*{\setcolsepandrule}[2]{}
137
138 \renewcommand*{\setheadfoot}[2]{}
139 \renewcommand*{\setheaderspaces}[3]{}
140 \renewcommand*{\setmarginnotes}[3]{}
141 \renewcommand*{\setfootins}[2]{}
142 \renewcommand*{\checkandfixthelayout}[1][]{}
143 \renewcommand*{\checkthelayout}[1]{}
144 \renewcommand*{\fixthelayout}{}
145 %
146 % \newlength{\stockheight}
147 % \newlength{\trimtop}
148 % \newlength{\trimedge}
149 % \newlength{\stockwidth}
150 % \newlength{\spinemargin}
151 % \newlength{\foremargin}
152 % \newlength{\uppermargin}
153 % \newlength{\headmargin}
154 %
155 \renewcommand*{\typeoutlayout}{}
156 \renewcommand*{\typeoutstandardlayout}{}
157 \renewcommand*{\settypeoutlayoutunit}[1]{}
158 \renewcommand*{\fixpdflayout}{}
159 \renewcommand*{\fixdvipslayout}{}
160
161 \renewcommand*{\medievalpage}[1][]{}
162 \renewcommand*{\isopage}[1][]{}
163 \renewcommand*{\semiisopage}[1][]{}
164
165 \renewcommand{\setpagebl}[3]{}
166 \renewcommand{\setpageml}[3]{}
167 \renewcommand{\setpagetl}[3]{}
168 \renewcommand{\setpagetm}[3]{}
169 \renewcommand{\setpagetr}[3]{}
170 \renewcommand{\setpagemr}[3]{}
171 \renewcommand{\setpagebr}[3]{}
172 \renewcommand{\setpagebm}[3]{}
173 \renewcommand{\setpagecc}[3]{}
```

§ 494.4 Text and fonts

```
174 \let\miniscule\tiny
175 \let\HUGE\Huge
176
177 \renewcommand*{\abnormalparskip}[1]{}
178 \renewcommand*{\nonzeroparskip}{}
179 \renewcommand*{\traditionalparskip}{}
180
```

```
\let\onelineskip\baselineskip

\let\OnehalfSpacing\onehalfspacing
\let\DoubleSpacing\doublespacing
\renewcommand*{\setPagenoteSpacing}[1]{}
\renewcommand*{\setFloatSpacing}[1]{}
\let\SingleSpacing\singlespacing
\let\setSingleSpace\SetSinglespace
\let\SingleSpace\singlespace
\let\endSingleSpace\endsinglespace
\let\Spacing\spacing
\let\endSpacing\endspacing
\let\OnehalfSpace\onehalfspace
\let\endOnehalfSpace\endonehalfspace
\csletcs{OnehalfSpace*}{onehalfspace}
\csletcs{endOnehalfSpace*}{endonehalfspace}
\let\DoubleSpace\doublespace
\let\endDoubleSpace\enddoublespace
\csletcs{DoubleSpace*}{doublespace}
\csletcs{endDoubleSpace*}{enddoublespace}
\renewcommand*{\setDisplayskipStretch}[1]{}
\renewcommand*{\memdskipstretch}{}
\renewcommand*{\noDisplayskipStretch}{}
\renewcommand*{\memdskips}{}

\renewcommand*{\midsloppy}{}
\renewenvironment*{midsloppypar}{}{}

\renewcommand*{\sloppybottom}{}
```

§ 494.5 Titles

```
\csletcs{titlingpage*}{titlingpage}
\csletcs{endtitlingpage*}{endtitlingpage}
\let\titlingpageend\relax
\newcommand{\titlingpageend}[2]{}
\let\andnext\and
\renewcommand*{\thanksmarkstyle}[1]{}
\renewcommand{\thanksfootmark}{%
\thanksscript{\tamark}%
}

% \newlength{\thanksmarksep}
```

§ 494.6 Abstracts

```
\renewcommand*{\abstractcol}{}
\renewcommand*{\abstractintoc}{}
\renewcommand*{\abstractnum}{}
\renewcommand*{\abstractrunin}{}
```

§ 494.7 Docment divisions

```

\def\@apppage{%
```

```
    \part*{\appendixpagename}
}
\renewcommand\mempreaddapppagetotochook{}
\renewcommand\mempostaddapppagetotochook{}

\def\@sapppage{%
    \part*{\appendixpagename}
}

\csletcs{frontmatter*}{frontmatter}
\csletcs{mainmatter*}{mainmatter}
\renewcommand*{\raggedbottomsection}{}
\renewcommand*{\normalbottomsection}{}
\renewcommand*{\bottomsectionskip}{}
\renewcommand*{\bottomsectionpenalty}{}
\csletcs{appendixpage*}{appendixpage}
\renewcommand*{\namedsubappendices}{}
\renewcommand*{\unnamedsubappendices}{}
\renewcommand*{\setsecnumdepth}[1]{}% todo tocvsec2
\renewcommand*{\maxsecnumdepth}[1]{}% todo tocvsec2
\renewcommand*{\beforebookskip}{}
\renewcommand*{\afterbookskip}{}
\renewcommand*{\beforepartskip}{}
\renewcommand*{\afterpartskip}{}
\renewcommand*{\midbookskip}{}
\renewcommand*{\midpartskip}{}
\renewcommand*{\printbookname}{}
\renewcommand*{\booknamefont}{}
\renewcommand*{\booknamenum}{}
\renewcommand*{\printbooknum}{}
\renewcommand*{\booknumfont}{}
\renewcommand*{\printpartname}{}
\renewcommand*{\partnamefont}{}
\renewcommand*{\partnamenum}{}
\renewcommand*{\printpartnum}{}
\renewcommand*{\partnumfont}{}
\renewcommand*{\printbooktitle}[1]{}
\renewcommand*{\booktitlefont}{}
\renewcommand{\printparttitle}[1]{}
\renewcommand*{\parttitlefont}{}
\renewcommand*{\bookpageend}{}
\renewcommand*{\bookblankpage}{}
\renewcommand*{\nobookblankpage}{}
\renewcommand*{\partpageend}{}
\renewcommand*{\partblankpage}{}
\renewcommand*{\nopartblankpage}{}
\RenewDocumentCommand{\newleadpage}{s o m m}{}% todo
\RenewDocumentCommand{\renewleadpage}{s o m m}{}% todo
\renewcommand*{\leadpagetoclevel}{chapter}

\renewcommand*{\openright}{}
\renewcommand*{\openleft}{}
\renewcommand*{\openany}{}
\renewcommand*{\clearforchapter}{}
\renewcommand*{\memendofchapterhook}{}
```

```
\renewcommand*{\chapterheadstart}{}
% \newlength{\beforechapskip}
\renewcommand*{\afterchapternum}{}
% \newlength{\midchapskip}
\renewcommand*{\afterchaptertitle}{}
% \newlength{\afterchapskip}
\renewcommand*{\printchaptername}{}
\renewcommand*{\chapnamefont}{}
\renewcommand*{\chapternamenum}{}
\renewcommand*{\printchapternum}{}
\renewcommand*{\chapnumfont}{}
\renewcommand{\printchaptertitle}[1]{}
\renewcommand*{\chaptitlefont}{}
\renewcommand*{\printchapternonum}{}
\renewcommand*{\indentafterchapter}{}
\renewcommand*{\noindentafterchapter}{}
\renewcommand*{\insertchapterspace}{}

\renewcommand*{\chapterstyle}[1]{}
\renewcommand{\makechapterstyle}[2]{}
\renewcommand*{\chapindent}{}
\let\chapterprecis\cftchapterprecis
\let\chapterprecishere\cftchapterprecishere
\let\chapterprecistoc\cftchapterprecistoc
\renewcommand*{\precisfont}{}
\renewcommand*{\prechapterprecis}{}
\renewcommand*{\postchapterprecis}{}
\renewcommand{\precistoctext}[1]{}
\renewcommand*{\precistocfont}{}
\renewcommand*{\precistocformat}{}
% \newlength{\prechapterprecisshift}

\renewcommand*{\setbeforesecskip}[1]{}
\renewcommand*{\setaftersecskip}[1]{}
\renewcommand*{\setsecindent}[1]{}
\renewcommand*{\setsecheadstyle}[1]{}
\renewcommand*{\setbeforesubsecskip}[1]{}
\renewcommand*{\setaftersubsecskip}[1]{}
\renewcommand*{\setsubsecindent}[1]{}
\renewcommand*{\setsubsecheadstyle}[1]{}
\renewcommand*{\setbeforesubsubsecskip}[1]{}
\renewcommand*{\setaftersubsubsecskip}[1]{}
\renewcommand*{\setsubsubsecindent}[1]{}
\renewcommand*{\setsubsubsecheadstyle}[1]{}
\renewcommand*{\setbeforeparaskip}[1]{}
\renewcommand*{\setafterparaskip}[1]{}
\renewcommand*{\setparaindent}[1]{}
\renewcommand*{\setparaheadstyle}[1]{}
\renewcommand*{\setbeforesubparaskip}[1]{}
\renewcommand*{\setaftersubparaskip}[1]{}
\renewcommand*{\setsubparaindent}[1]{}
\renewcommand*{\setsubparaheadstyle}[1]{}
\renewcommand{\@hangfrom}[1]{#1}
\renewcommand{\sethangfrom}[1]{}
\renewcommand{\setsecnumformat}[1]{}
```

```
337
338 \renewcommand*{\hangsecnum}{}
339 \renewcommand*{\defaultsecnum}{}
340
341 \renewcommand*{\sechook}{}
342 \renewcommand{\setsechook}[1]{}
343 \renewcommand*{\subsechook}{}
344 \renewcommand{\setsubsechook}[1]{}
345 \renewcommand*{\subsubsechook}{}
346 \renewcommand{\setsubsubsechook}[1]{}
347 \renewcommand*{\parahook}{}
348 \renewcommand{\setparahook}[1]{}
349 \renewcommand*{\subparahook}{}
350 \renewcommand{\setsubparahook}[1]{}
351
352 \RenewDocumentCommand{\plainbreak}{s m}{\begin{center}~\end{center}}
353
354 \RenewDocumentCommand{\fancybreak}{s +m}{%
355 \begin{center}#2\end{center}%
356 }
357
358 \RenewDocumentCommand{\plainfancybreak}{s m m +m}{%
359 \begin{center}#4\end{center}%
360 }
361
362 \RenewDocumentCommand{\pfbreak}{s}{%
363 \begin{center}
364 \pfbreakdisplay
365 \end{center}
366 }
367
368 % \newlength{\pfbreakskip}
369 \renewcommand{\pfbreakdisplay}{*\quad*\quad*}
370
371 \renewcommand{\makeheadstyles}[2]{}
372 \renewcommand*{\headstyles}[1]{}
```

§ 494.8 Pagination and headers

```
373 \renewcommand*{\savepagenumber}{}
374 \renewcommand*{\restorepagenumber}{}
375 \renewcommand*{\uppercaseheads}{}
376 \renewcommand*{\nouppercaseheads}{}
377
378 \renewcommand*{\bookpagemark}[1]{}
379 \renewcommand*{\partmark}[1]{}
380 \renewcommand*{\bibmark}{}
381 \renewcommand*{\indexmark}{}
382 \renewcommand*{\glossarymark}{}
383
384 \LWR@origpagestyle{empty}
385 \renewcommand*{\ps@empty}{}
386 \renewcommand*{\makepagestyle}[1]{}
387 \renewcommand*{\emptypshook}{}%
388 % \renewcommand*{\empty@oddhead}{}
```

```
% \renewcommand*{\empty@oddfoot}{}
% \renewcommand*{\empty@evenhead}{}
% \renewcommand*{\empty@evenfoot}{}
\renewcommand*{\@oddhead}{}
\renewcommand*{\@oddfoot}{}
\renewcommand*{\@evenhead}{}
\renewcommand*{\@evenfoot}{}
\renewcommand*{\aliaspagestyle}[2]{}
\renewcommand*{\copypagestyle}[2]{}

\renewcommand*{\makeevenhead}[4]{}
\renewcommand*{\makeoddhead}[4]{}
\renewcommand*{\makeevenfoot}[4]{}
\renewcommand*{\makeoddfoot}[4]{}
\renewcommand*{\makerunningwidth}[3]{}
% \newlength{\headwidth}
\renewcommand*{\makeheadrule}[3]{}
\renewcommand*{\makefootrule}[3]{}
\renewcommand*{\makeheadfootruleprefix}[3]{}
% \newlength{\normalrulethickness}
% \setlength{\normalrulethickness}{.4pt}
% \newlength{\footruleheight}
% \newlength{\footruleskip}
\renewcommand*{\makeheadposition}[5]{}
\renewcommand{\makepsmarks}[2]{}
\renewcommand*{\makeheadfootstrut}[3]{}

\renewcommand{\createplainmark}[3]{}
\renewcommand{\memUChead}[1]{}
\renewcommand{\createmark}[5]{}
\renewcommand*{\clearplainmark}[1]{}
\renewcommand*{\clearmark}[1]{}
\renewcommand{\addtopsmarks}[3]{}
\renewcommand{\ifonlyfloats}[2]{#2}
\renewcommand*{\mergepagefloatstyle}[3]{}

\renewcommand*{\framepichead}{}
\renewcommand*{\framepictextfoot}{}
\renewcommand*{\framepichook}{}
\renewcommand*{\showheadfootlocoff}{}
\renewcommand*{\showtextblocklocoff}{}
```

§ 494.9 Paragraphs and lists

```
\renewcommand{\hangfrom}[1]{#1}
\let\centerfloat\centering
\renewcommand*{\raggedyright}[1][]{}
% \newlength{\ragrparindent}
\renewcommand{\sourceatright}[2][]{\attribution{#2}}
\let\memorigdbs\LWR@endofline
\let\memorigpar\par
\let\atcentercr\LWR@endofline

\renewcommand*{\linenottooshort}[1][]{}
\renewcommand*{\russianpar}{}
```

```
\renewcommand*{\lastlinerulefill}{}
\renewcommand*{\lastlineparrule}{}
\renewcommand*{\justlastraggedleft}{}
\renewcommand*{\raggedrightthenleft}{}
\renewcommand*{\leftcenterright}{}

\renewcommand{\leftspringright}[4]{%
\begin{minipage}{#1\linewidth}#3\end{minipage}\qquad%
\begin{minipage}{#2\linewidth}\begin{flushright}#4\end{flushright}\end{minipage}%
}

\renewenvironment*{blockdescription}
{\LWR@descriptionstart\LWR@origdescription}
{\enddescription}
\renewcommand*{\blockdescriptionlabel}[1]{\textbf{#1}}
\renewenvironment*{labelled}[1]{\begin{description}}{\end{description}}
\renewenvironment*{flexlabelled}[6]{\begin{description}}{\end{description}}
\renewcommand*{\tightlists}{}
\renewcommand*{\defaultlists}{}
\RenewDocumentCommand{\firmlists}{s}{}
\renewcommand*{\firmlist}{}
\renewcommand*{\tightlist}{}
\renewcommand*{\zerotrivseps}{}
\renewcommand*{\savetrivseps}{}
\renewcommand*{\restoretrivseps}{}
```

§ 494.10 Contents lists

```
\csletcs{tableofcontents*}{tableofcontents}
\csletcs{listoffigures*}{listoffigures}
\csletcs{listoftables*}{listoftables}
\renewenvironment{KeepFromToc}{}{}
\renewcommand*{\onecoltocetc}{}
\renewcommand*{\twocoltocetc}{}
\renewcommand*{\ensureonecol}{}
\renewcommand*{\restorefromonecol}{}
\renewcommand*{\doccoltocetc}{}
\renewcommand*{\maxtocdepth}[1]{}% tocvsec2
\renewcommand*{\settocdepth}[1]{}% tocvsec2

\renewcommand{\tocheadstart}{}
\renewcommand{\printtoctitle}[1]{}
\renewcommand{\tocmark}{}
\renewcommand{\aftertoctitle}{}
\renewcommand{\lofheadstart}{}
\renewcommand{\printloftitle}[1]{}
\renewcommand{\lofmark}{}
\renewcommand{\afterloftitle}{}
\renewcommand{\lotheadstart}{}
\renewcommand{\printlottitle}[1]{}
\renewcommand{\lotmark}{}
\renewcommand{\afterlottitle}{}

\renewcommand*{\setpnumwidth}[1]{}
\renewcommand*{\setrmarg}[1]{}
```

```
\renewcommand*{\cftbookbreak}{}
\renewcommand*{\cftpartbreak}{}
\renewcommand*{\cftchapterbreak}{}

% \newlength{\cftbeforebookskip}
% \newlength{\cftbookindent}
% \newlength{\cftbooknumwidth}
\renewcommand*{\cftbookfont}{}
\renewcommand*{\cftbookname}{}
\renewcommand*{\cftbookpresnum}{}
\renewcommand*{\cftbookaftersnum}{}
\renewcommand*{\cftbookaftersnumb}{}
\renewcommand*{\cftbookleader}{}
\renewcommand*{\cftbookdotsep}{1}
\renewcommand*{\cftbookpagefont}{}
\renewcommand*{\cftbookafterpnum}{}
\renewcommand*{\cftbookformatpnum}[1]{}
\renewcommand*{\cftbookformatpnumhook}[1]{}
```

Part is already defined by tocloft.

```
% \newlength{\cftbeforechapterskip}
% \newlength{\cftchapterindent}
% \newlength{\cftchapternumwidth}
\renewcommand*{\cftchapterfont}{}
\renewcommand*{\cftchaptername}{}
\renewcommand*{\cftchapterpresnum}{}
\renewcommand*{\cftchapteraftersnum}{}
\renewcommand*{\cftchapteraftersnumb}{}
\renewcommand*{\cftchapterleader}{}
\renewcommand*{\cftchapterdotsep}{1}
\renewcommand*{\cftchapterpagefont}{}
\renewcommand*{\cftchapterafterpnum}{}
\renewcommand*{\cftchapterformatpnum}[1]{}
\renewcommand*{\cftchapterformatpnumhook}[1]{}

% \newlength{\cftbeforesectionskip}
% \newlength{\cftsectionindent}
% \newlength{\cftsectionnumwidth}
\renewcommand*{\cftsectionfont}{}
\renewcommand*{\cftsectionname}{}
\renewcommand*{\cftsectionpresnum}{}
\renewcommand*{\cftsectionaftersnum}{}
\renewcommand*{\cftsectionaftersnumb}{}
\renewcommand*{\cftsectionleader}{}
\renewcommand*{\cftsectiondotsep}{1}
\renewcommand*{\cftsectionpagefont}{}
\renewcommand*{\cftsectionafterpnum}{}
\renewcommand*{\cftsectionformatpnum}[1]{}
\renewcommand*{\cftsectionformatpnumhook}[1]{}

% \newlength{\cftbeforesubsectionskip}
% \newlength{\cftsubsectionindent}
% \newlength{\cftsubsectionnumwidth}
\renewcommand*{\cftsubsectionfont}{}
\renewcommand*{\cftsubsectionname}{}
```

```
\renewcommand*{\cftsubsectionpresnum}{}
\renewcommand*{\cftsubsectionaftersnum}{}
\renewcommand*{\cftsubsectionaftersnumb}{}
\renewcommand*{\cftsubsectionleader}{}
\renewcommand*{\cftsubsectiondotsep}{1}
\renewcommand*{\cftsubsectionpagefont}{}
\renewcommand*{\cftsubsectionafterpnum}{}
\renewcommand*{\cftsubsectionformatpnum}[1]{}
\renewcommand*{\cftsubsectionformatpnumhook}[1]{}

% \newlength{\cftbeforesubsubsectionskip}
% \newlength{\cftsubsubsectionindent}
% \newlength{\cftsubsubsectionnumwidth}
\renewcommand*{\cftsubsubsectionfont}{}
\renewcommand*{\cftsubsubsectionname}{}
\renewcommand*{\cftsubsubsectionpresnum}{}
\renewcommand*{\cftsubsubsectionaftersnum}{}
\renewcommand*{\cftsubsubsectionaftersnumb}{}
\renewcommand*{\cftsubsubsectionleader}{}
\renewcommand*{\cftsubsubsectiondotsep}{1}
\renewcommand*{\cftsubsubsectionpagefont}{}
\renewcommand*{\cftsubsubsectionafterpnum}{}
\renewcommand*{\cftsubsubsectionformatpnum}[1]{}
\renewcommand*{\cftsubsubsectionformatpnumhook}[1]{}

% \newlength{\cftbeforeparagraphskip}
% \newlength{\cftparagraphindent}
% \newlength{\cftparagraphnumwidth}
\renewcommand*{\cftparagraphfont}{}
\renewcommand*{\cftparagraphname}{}
\renewcommand*{\cftparagraphpresnum}{}
\renewcommand*{\cftparagraphaftersnum}{}
\renewcommand*{\cftparagraphaftersnumb}{}
\renewcommand*{\cftparagraphleader}{}
\renewcommand*{\cftparagraphdotsep}{1}
\renewcommand*{\cftparagraphpagefont}{}
\renewcommand*{\cftparagraphafterpnum}{}
\renewcommand*{\cftparagraphformatpnum}[1]{}
\renewcommand*{\cftparagraphformatpnumhook}[1]{}

% \newlength{\cftbeforesubparagraphskip}
% \newlength{\cftsubparagraphindent}
% \newlength{\cftsubparagraphnumwidth}
\renewcommand*{\cftsubparagraphfont}{}
\renewcommand*{\cftsubparagraphname}{}
\renewcommand*{\cftsubparagraphpresnum}{}
\renewcommand*{\cftsubparagraphaftersnum}{}
\renewcommand*{\cftsubparagraphaftersnumb}{}
\renewcommand*{\cftsubparagraphleader}{}
\renewcommand*{\cftsubparagraphdotsep}{1}
\renewcommand*{\cftsubparagraphpagefont}{}
\renewcommand*{\cftsubparagraphafterpnum}{}
\renewcommand*{\cftsubparagraphformatpnum}[1]{}
\renewcommand*{\cftsubparagraphformatpnumhook}[1]{}

% \newlength{\cftbeforefigureskip}
```

```
% \newlength{\cftfigureindent}
% \newlength{\cftfigurenumwidth}
\renewcommand*{\cftfigurefont}{}
\renewcommand*{\cftfigurename}{}
\renewcommand*{\cftfigurepresnum}{}
\renewcommand*{\cftfigureaftersnum}{}
\renewcommand*{\cftfigureaftersnumb}{}
\renewcommand*{\cftfigureleader}{}
\renewcommand*{\cftfiguredotsep}{1}
\renewcommand*{\cftfigurepagefont}{}
\renewcommand*{\cftfigureafterpnum}{}
\renewcommand*{\cftfigureformatpnum}[1]{}
\renewcommand*{\cftfigureformatpnumhook}[1]{}

% \newlength{\cftbeforesubfigureskip}
% \newlength{\cftsubfigureindent}
% \newlength{\cftsubfigurenumwidth}
\newcommand*{\cftsubfigurefont}{}
\newcommand*{\cftsubfigurename}{}
\newcommand*{\cftsubfigurepresnum}{}
\newcommand*{\cftsubfigureaftersnum}{}
\newcommand*{\cftsubfigureaftersnumb}{}
\newcommand*{\cftsubfigureleader}{}
\newcommand*{\cftsubfiguredotsep}{1}
\newcommand*{\cftsubfigurepagefont}{}
\newcommand*{\cftsubfigureafterpnum}{}
\newcommand*{\cftsubfigureformatpnum}[1]{}
\newcommand*{\cftsubfigureformatpnumhook}[1]{}

% \newlength{\cftbeforetableskip}
% \newlength{\cfttableindent}
% \newlength{\cfttablenumwidth}
\renewcommand*{\cfttablefont}{}
\renewcommand*{\cfttablename}{}
\renewcommand*{\cfttablepresnum}{}
\renewcommand*{\cfttableaftersnum}{}
\renewcommand*{\cfttableaftersnumb}{}
\renewcommand*{\cfttableleader}{}
\renewcommand*{\cfttabledotsep}{1}
\renewcommand*{\cfttablepagefont}{}
\renewcommand*{\cfttableafterpnum}{}
\renewcommand*{\cfttableformatpnum}[1]{}
\renewcommand*{\cfttableformatpnumhook}[1]{}

% \newlength{\cftbeforesubtableskip}
% \newlength{\cftsubtableindent}
% \newlength{\cftsubtablenumwidth}
\newcommand*{\cftsubtablefont}{}
\newcommand*{\cftsubtablename}{}
\newcommand*{\cftsubtablepresnum}{}
\newcommand*{\cftsubtableaftersnum}{}
\newcommand*{\cftsubtableaftersnumb}{}
\newcommand*{\cftsubtableleader}{}
\newcommand*{\cftsubtabledotsep}{1}
\newcommand*{\cftsubtablepagefont}{}
\newcommand*{\cftsubtableafterpnum}{}
```

```
\newcommand*{\cftsubtableformatpnum}[1]{}
\newcommand*{\cftsubtableformatpnumhook}[1]{}

\renewcommand*{\booknumberline}[1]{}
\renewcommand*{\partnumberline}[1]{}
\renewcommand*{\chapternumberline}[1]{}
\renewcommand*{\numberlinehook}[1]{}
% \renewcommand*{\cftwhatismyname}{}%
\renewcommand*{\booknumberlinehook}[1]{}
\renewcommand*{\partnumberlinehook}[1]{}
\renewcommand*{\chapternumberlinehook}[1]{}
\renewcommand{\numberlinebox}[2]{}
\renewcommand{\booknumberlinebox}[2]{}
\renewcommand{\partnumberlinebox}[2]{}
\renewcommand{\chapternumberlinebox}[2]{}
%
% \newlength{\cftparfillskip}
\renewcommand*{\cftpagenumbersoff}[1]{}
\renewcommand*{\cftpagenumberson}[1]{}
\renewcommand*{\cftlocalchange}[3]{}
\renewcommand*{\cftaddtitleline}[4]{}
\renewcommand*{\cftaddnumtitleline}[4]{}
\renewcommand{\cftinsertcode}[2]{}
\renewcommand{\cftinserthook}[2]{}
\renewcommand{\settocpreprocessor}[2]{}
\DeclareRobustCommand{\cftpagenumbersoff}[1]{}
\DeclareRobustCommand{\cftpagenumberson}[1]{}
```

§ 494.11 Floats and captions

`\newfloat` `[`⟨*1: within*⟩`]` {⟨*2: type*⟩} {⟨*3: ext*⟩} {⟨*4: capname*⟩}

```
\RenewDocumentCommand{\newfloat}{o m m m}{%
\IfValueTF{#1}%
{\DeclareFloatingEnvironment[fileext=#3,within=#1,name={#4}]{#2}}%
{\DeclareFloatingEnvironment[fileext=#3,name={#4}]{#2}}%
```

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.

```
\cslet{listof#2s}\relax%
\cslet{listof#2es}\relax%
}
```

`\newlistof` `[`⟨*within*⟩`]` {⟨*type*⟩} {⟨*ext*⟩} {⟨*listofname*⟩}

Emulated through the `\newfloat` mechanism. Note that memoir uses a different syntax than tocloft for the name.

```
\RenewDocumentCommand{\newlistof}{o m m m}
{%
\IfValueTF{#1}
{\newlistentry[#1]{#2}{#3}{0}}
{\newlistentry{#2}{#3}{0}}
\@namedef{ext@#2}{#3}%
\@ifundefined{c@#3depth}{\newcounter{#3depth}}{}%
\setcounter{#3depth}{1}%
```

```
\@namedef{#3mark}{}%
\@namedef{#2}{\listof{#2}{#4}}
\@namedef{@cftmake#3title}{}
\@ifundefined{cftbefore#3titleskip}{
    \expandafter\newlength\csname cftbefore#3titleskip\endcsname
    \expandafter\newlength\csname cftafter#3titleskip\endcsname
}{}
\@namedef{cft#3titlefont}{}
\@namedef{cftafter#3title}{}
\@namedef{cft#3prehook}{}
\@namedef{cft#3posthook}{}
}

\renewcommand{\setfloatadjustment}[2]{}
```

Borrowed from the lwarp version of keyfloat:

```
\NewDocumentEnvironment{KFLTmemoir@marginfloat}{O{-1.2ex} m}
{% start
\LWR@BlockClassWP{float:right; width:2in; margin:10pt}{}{marginblock}%
\captionsetup{type=#2}%
}
{%
\endLWR@BlockClassWP%
}

\DeclareDocumentEnvironment{marginfigure}{o}
  {\begin{KFLTmemoir@marginfloat}{figure}}
  {\end{KFLTmemoir@marginfloat}}

\DeclareDocumentEnvironment{margintable}{o}
  {\begin{KFLTmemoir@marginfloat}{table}}
  {\end{KFLTmemoir@marginfloat}}

\renewcommand{\setmarginfloatcaptionadjustment}[2]{}
\renewcommand{\setmpjustification}[2]{}
\renewcommand*{\mpjustification}{}
\renewcommand*{\setfloatlocations}[2]{}
\DeclareDocumentCommand{\suppressfloats}{o}{}
\renewcommand*{\FloatBlock}{}
\renewcommand*{\FloatBlockAllowAbove}{}
\renewcommand*{\FloatBlockAllowBelow}{}
\renewcommand*{\setFloatBlockFor}{}

\renewcommand{\captiontitlefinal}[1]{}

\renewcommand{\flegtable}{\tablename}
\renewcommand{\flegfigure}{\figurename}
\renewcommand{\flegtoctable}{}
\renewcommand{\flegtocfigure}{}


\renewcommand{\subcaption}[2][]{%
\ifblank{#1}{\subfloat[#2]{}}{\subfloat[#1][#2]{}}%
}
```

```
\renewcommand{\contsubcaption}{\ContinuedFloat\subcaption}

\LetLtxMacro\subcaptionref\subref

\renewcommand*{\tightsubcaptions}{}
\renewcommand*{\loosesubcaptions}{}

\renewcommand*{\subcaptionsize}[1]{}
\renewcommand*{\subcaptionlabelfont}[1]{}
\renewcommand*{\subcaptionfont}[1]{}
\renewcommand*{\subcaptionstyle}[1]{}

\renewcommand*{\hangsubcaption}{}
\renewcommand*{\shortsubcaption}{}
\renewcommand*{\normalsubcaption}{}

\RenewDocumentEnvironment{sidecaption}{o m o}
{}
{
\IfValueTF{#1}{\caption[#1]{#2}}{\caption{#2}}%
\IfValueT{#3}{\label{#3}}%
}

% \newlength{\sidecapwidth}
% \newlength{\sidecapsep}
\renewcommand*{\setsidecaps}[2]{}
\renewcommand*{\sidecapmargin}[1]{}
% \newif\ifscapmargleft
\scapmargleftfalse
\renewcommand*{\setsidecappos}[1]{}

\RenewDocumentEnvironment{sidecontcaption}{m o}
{}
{%
\ContinuedFloat%
\caption{#1}%
```

Without \@captype, the section is referred to instead.

```
\IfValueT{#2}{\label[\@captype]{#2}}%
}
```

\sidenamedlegend does not appear to use the TOC argument.

```
\renewenvironment{sidenamedlegend}[2][]{
\begin{center}
\@nameuse{\@captype name}\CaptionSeparator#2
\end{center}
}
{}

\renewenvironment{sidelegend}[1]
{\begin{center}
#1
```

```
788
789 }
790 {\end{center}}
791
792 \renewcommand*{\sidecapstyle}{}
793 \renewcommand*{\overridescapmargin}[1]{}
794 % \newlength{\sidecapraise}
795 \renewcommand*{\sidecapfloatwidth}{\linewidth}
796
797 \LetLtxMacro\ctabular\tabular
798 \LetLtxMacro\endctabular\endtabular
799
800 \renewcommand{\autorows}[5][]{%
801 #5
802 }
803
804 \renewcommand{\autocols}[5][]{%
805 #5
806 }
```

§ 494.12 Page notes

```
807 \renewcommand*{\feetabovefloat}{}
808 \renewcommand*{\feetbelowfloat}{}
809 \renewcommand*{\feetatbottom}{}
810
811 \renewcommand*{\verbfootnote}[2][]{
812 \PackageError{lwarp,memoir}
813 {Verbatim footnotes are not yet supported by lwarp.}
814 {This may be improved some day.}
815 }
816
817 \renewcommand*{\plainfootnotes}{}
818 \renewcommand*{\twocolumnfootnotes}{}
819 \renewcommand*{\threecolumnfootnotes}{}
820 \renewcommand*{\paragraphfootnotes}{}
821 \renewcommand*{\footfudgefiddle}{}
822
823 \renewcommand*{\newfootnoteseries}[1]{
824 \PackageError{lwarp,memoir}
825 {Memoir footnote series are not yet supported by lwarp.}
826 {This may be improved some day.}
827 }
828
829 \renewcommand*{\plainfootstyle}[1]{}
830 \renewcommand*{\twocolumnfootstyle}[1]{}
831 \renewcommand*{\threecolumnfootstyle}[1]{}
832 \renewcommand*{\paragraphfootstyle}[1]{}
833
834 \renewcommand*{\footfootmark}{}
835 \renewcommand*{\footmarkstyle}[1]{}
836
837 % \newlength{\footmarkwidth}
838 % \newlength{\footmarksep}
839 % \newlength{\footparindent}
```

```
840
841 \renewcommand*{\foottextfont}{}
842
843 \renewcommand*{\marginparmargin}[1]{}
844 \renewcommand*{\sideparmargin}[1]{}
845
846 \LetLtxMacro\sidepar\marginpar
847 \renewcommand*{\sideparfont}{}
848 \renewcommand*{\sideparform}{}
849 \LWR@providelength{\sideparvshift}
850
851 \renewcommand*{\parnopar}{}
852
853 \renewcommand{\sidebar}[1]{\begin{quote}#1\end{quote}}
854 \renewcommand*{\sidebarmargin}[1]{}
855 \renewcommand*{\sidebarfont}{}
856 \renewcommand*{\sidebarform}{}
857 % \newlength{\sidebarhsep}
858 % \newlength{\sidebarvsep}
859 % \newlength{\sidebarwidth}
860 % \newlength{\sidebartopsep}
861 \renewcommand{\setsidebarheight}[1]{}
862 \renewcommand*{\setsidebars}[6]{}
863 \renewcommand*{\footnotesatfoot}{}
864 \renewcommand*{\footnotesinmargin}{}
865
866 \LetLtxMacro\sidefootnote\footnote
867 \LetLtxMacro\sidefootnotemark\footnotemark
868 \LetLtxMacro\sidefootnotetext\footnotetext
869
870 \renewcommand*{\sidefootmargin}[1]{}
871 % \newlength{\sidefoothsep}
872 % \newlength{\sidefootvsep}
873 % \newlength{\sidefootwidth}
874 % \newlength{\sidefootadjust}
875 % \newlength{\sidefootheight}
876 \renewcommand*{\setsidefootheight}[1]{}
877 % \renewcommand*{\sidefootfont}{}% in docs but not in the package
878 \renewcommand*{\setsidefeet}[6]{}
879 \renewcommand*{\sidefootmarkstyle}[1]{}
880 \renewcommand*{\sidefoottextfont}{}
881 \renewcommand*{\sidefootform}{}
882
883 \renewcommand*{\continuousnotenums}{\pncontopttrue}% from pagenote
884 \renewcommand*{\notepageref}{}
885 \renewcommand*{\prenotetext}{}
886 \renewcommand*{\postnotetext}{}
887 \renewcommand*{\idtextinnotes}[1]{}
888 \renewcommand*{\printpageinnotes}[1]{}
889 \renewcommand*{\printpageinnoteshyperref}[1]{}
890 \renewcommand*{\foottopagenote}{}
891 \renewcommand*{\pagetofootnote}{}
```

§ 494.13 Decorative text

```
\renewcommand*{\epigraphposition}[1]{}
\renewcommand*{\epigraphtextposition}[1]{}
\renewcommand*{\epigraphsourceposition}[1]{}
\renewcommand*{\epigraphfontsize}[1]{}
\renewcommand*{\epigraphforheader}[2][]{}
\renewcommand*{\epigraphpicture}{}
```

§ 494.14 Poetry

```
\renewcommand*{\vinphantom}{}
\renewcommand*{\vleftofline}[1]{#1}
% \let\linenumberfrequency\poemlines
% \renewcommand*{\linenumberfont}[1]{}

\DeclareDocumentCommand{\PoemTitle}{s o o m}{%
\IfValueTF{#2}%
{\poemtitle[#2]{#4}}%
{\poemtitle{#4}}%
}

\renewcommand*{\NumberPoemTitle}{}
\renewcommand*{\PlainPoemTitle}{}
\renewcommand*{\poemtitlepstyle}{}
\renewcommand*{\poemtitlestarmark}[1]{}
\renewcommand*{\poemtitlestarpstyle}{}
\renewcommand*{\PoemTitleheadstart}{}
\renewcommand*{\printPoemTitlenonum}{}
\renewcommand*{\printPoemTitlenum}{}
\renewcommand*{\afterPoemTitlenum}{}
\renewcommand*{\printPoemTitletitle}[1]{}
\renewcommand*{\afterPoemTitle}{}
\newlength{\midpoemtitleskip}
\renewcommand*{\PoemTitlenumfont}{}
\renewcommand*{\PoemTitlefont}{}
```

§ 494.15 Boxes, verbatims and files

```
\renewenvironment{qframe}{\framed}{\endframed}
\renewenvironment{qshade}{\shaded}{\endshaded}
```

Use the comment package:

```
\renewcommand*{\commentsoff}[1]{\includecomment{#1}}
\renewcommand*{\commentson}[1]{\excludecomment{#1}}
\LetLtxMacro\renewcomment\commentson

\renewcommand*{\setverbatimfont}[1]{}
\renewcommand*{\tabson}[1]{}
\renewcommand*{\tabsoff}{}
\renewcommand*{\wrappingon}{}
\renewcommand*{\wrappingoff}{}
\renewcommand*{\verbatimindent}{}
\renewcommand*{\verbatimbreakchar}[1]{}

\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.

```
\renewcommand*{\bvbox}{}
\renewcommand*{\bvtopandtail}{}
\renewcommand*{\bvsides}{}
\renewcommand*{\nobvbox}{}
% \newlength\bvboxsep
\renewcommand*{\bvtoprulehook}{}
\renewcommand*{\bvtopmidhook}{}
\renewcommand*{\bvendrulehook}{}
\renewcommand*{\bvleftsidehook}{}
\renewcommand*{\bvrightsidehook}{}
\renewcommand*{\bvperpagetrue}{}
\renewcommand*{\bvperpagefalse}{}
\renewcommand{\bvtopofpage}[1]{}
\renewcommand{\bvendofpage}[1]{}
\renewcommand*{\linenumberfrequency}[1]{}
\renewcommand*{\resetbvlinenumber}{}
\renewcommand*{\setbvlinenums}[2]{}
\renewcommand*{\linenumberfont}[1]{}
\renewcommand*{\bvnumbersinside}{}
\renewcommand*{\bvnumbersoutside}{}
```

§ 494.16 Cross referencing

```
\renewcommand*{\fref}[1]{\cref{#1}}
\renewcommand*{\tref}[1]{\cref{#1}}
\renewcommand*{\pref}[1]{\cpageref{#1}}
\renewcommand*{\Aref}[1]{\cref{#1}}
\renewcommand*{\Bref}[1]{\cref{#1}}
\renewcommand*{\Pref}[1]{\cref{#1}}
\renewcommand*{\Sref}[1]{\cref{#1}}
\renewcommand*{\figurerefname}{Figure}
\renewcommand*{\tablerefname}{Table}
\renewcommand*{\pagerefname}{page}
\renewcommand*{\bookrefname}{Book~}
\renewcommand*{\partrefname}{Part~}
\renewcommand*{\chapterrefname}{Chapter~}
\renewcommand*{\sectionrefname}{\S}
\renewcommand*{\appendixrefname}{Appendix~}
\LetLtxMacro\titleref\nameref
\renewcommand*{\headnameref}{}
\renewcommand*{\tocnameref}{}

\providecounter{LWR@currenttitle}

\renewcommand*{\currenttitle}{%
    \addtocounter{LWR@currenttitle}{1}%
    \label{currenttitle\arabic{LWR@currenttitle}}%
    \nameref{currenttitle\arabic{LWR@currenttitle}}%
}

\renewcommand*{\theTitleReference}[2]{}
\renewcommand*{\namerefon}{}
```

```
\renewcommand*{\namerefoff}{}
```

§ 494.17 Back matter

Redefined to write the LWR@autoindex counter instead of page. Note that memoir has two versions, depending on the use of hyperref.

```
\AtBeginDocument{

\def\@@wrindexhyp#1||\\{%
    \addtocounter{LWR@autoindex}{1}%
    \LWR@new@label{LWRindex-\arabic{LWR@autoindex}}%
%     \ifshowindexmark\@showidx{#1}\fi
    \protected@write\@auxout{}%
    {\string\@@wrindexm@m{\@idxfile}{#1}{\arabic{LWR@autoindex}}}%
    \endgroup
    \@esphack}%
```

\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.

```
\def\@@wrspindexhyp#1||\\{%
    \addtocounter{LWR@autoindex}{1}%
    \LWR@new@label{LWRindex-\arabic{LWR@autoindex}}%
%     \ifshowindexmark\@showidx{#1}\fi
    \protected@write\@auxout{}%
    {%
%         \string\@@wrindexm@m{\@idxfile}{#1}{\@nameuse{the\@sptheidx}}%
        \string\@@wrindexm@m{\@idxfile}{#1}{\arabic{LWR@autoindex}}%
    }%
    \endgroup
    \@esphack}%

}% \AtBeginDocument
```

Patched to use _html filename and \BaseJobname:

```
\catcode`\_=12%
\renewcommand*{\makeindex}[1][\BaseJobname]{%
  \if@filesw
    \def\gindex{\@bsphack%
      \@ifnextchar [{\@index}{\@index[\BaseJobname]}}
    \def\specialindex{\@bsphack\@spindex}%
    \makememindexhook
    \expandafter\newwrite\csname #1@idxfile\endcsname
   \expandafter\immediate\openout \csname #1@idxfile\endcsname #1_html.idx\relax
    \typeout{Writing index file #1_html.idx }%
  \fi}
\catcode`\_=8%
```

Patched to use _html filename and \BaseJobname. This will later be patched by the lwarp core.

```
\catcode`\_=12%
\renewcommand{\printindex}[1][\BaseJobname]{\@input@{#1_html.ind}}
\catcode`\_=8%

\DeclareDocumentCommand{\newblock}{}{}
%
\renewcommand*{\showindexmarks}{}
\renewcommand*{\hideindexmarks}{}

\renewcommand*{\xindyindex}{}
```

§ 494.18 Miscellaneous

```
\renewcommand*{\changemarks}{}
\renewcommand*{\nochangemarks}{}
\renewcommand*{\added}[1]{}
\renewcommand*{\deleted}[1]{}
\renewcommand*{\changed}[1]{}

\renewcommand*{\showtrimsoff}{}
\renewcommand*{\showtrimson}{}
\renewcommand*{\trimXmarks}{}
\renewcommand*{\trimLmarks}{}
\renewcommand*{\trimFrame}{}
\renewcommand*{\trimNone}{}
\renewcommand*\trimmarkscolor{}
\renewcommand*{\trimmarks}{}
\renewcommand*{\tmarktl}{}
\renewcommand*{\tmarktr}{}
\renewcommand*{\tmarkbr}{}
\renewcommand*{\tmarkbl}{}
\renewcommand*{\tmarktm}{}
\renewcommand*{\tmarkmr}{}
\renewcommand*{\tmarkbm}{}
\renewcommand*{\tmarkml}{}
\renewcommand*{\trimmark}{}
\renewcommand*{\quarkmarks}{}
\renewcommand*{\registrationColour}[1]{}

\renewcommand*{\leavespergathering}[1]{}

\renewcommand*{\noprelistbreak}{}

\renewcommand*{\cleartorecto}{}
\renewcommand*{\cleartoverso}{}

\renewenvironment{vplace}[1][]{}{}
```

§ 494.19 ccaption emulation

```
\renewcommand*{\captiondelim}[1]{\renewcommand*{\CaptionSeparator}{#1}}
\renewcommand*{\captionnamefont}[1]{}
\renewcommand*{\captiontitlefont}[1]{}
\renewcommand*{\flushleftright}{}
\renewcommand*{\centerlastline}{}
```

```
\renewcommand*{\captionstyle}[2][]{}
\DeclareDocumentCommand{\captionwidth}{m}{}
\renewcommand*{\changecaptionwidth}{}
\renewcommand*{\normalcaptionwidth}{}
\renewcommand*{\hangcaption}{}
\renewcommand*{\indentcaption}[1]{}
\renewcommand*{\normalcaption}{}
\renewcommand{\precaption}[1]{}
\renewcommand{\postcaption}[1]{}
\renewcommand{\midbicaption}[1]{}
\renewcommand{\contcaption}[1]{%
% \ContinuedFloat%
% \caption{#1}%
\begin{LWR@figcaption}% later becomes \caption*
\LWR@isolate{\@nameuse{\@captype name}}~%
\thechapter.\the\value{\@captype}\CaptionSeparator\LWR@isolate{#1}%
\end{LWR@figcaption}
}

\newlength{\abovelegendskip}
\setlength{\abovelegendskip}{0.5\baselineskip}
\newlength{\belowlegendskip}
\setlength{\belowlegendskip}{\abovelegendskip}
```

The extra \\ here forces a <br> in HTML when \legend is used in a \marginpar.

```
\renewcommand{\legend}[1]{\begin{center}#1\\\end{center}}

\renewcommand{\namedlegend}[2][]{
\begin{center}
\@nameuse{fleg\@captype}\CaptionSeparator#2\\
\end{center}
\@nameuse{flegtoc\@captype}{#1}
}

\renewcommand{\newfixedcaption}[3][\caption]{%
  \renewcommand{#2}{\def\@captype{#3}#1}}
\renewcommand{\renewfixedcaption}[3][\caption]{%
  \renewcommand{#2}{\def\@captype{#3}#1}}
\renewcommand{\providefixedcaption}[3][\caption]{%
  \providecommand{#2}{\def\@captype{#3}#1}}

\renewcommand{\bitwonumcaption}[6][]{%
\ifblank{#2}{\caption{#3}}{\caption[#2]{#3}}%
\addtocounter{\@captype}{-1}%
\begingroup%
\csdef{\@captype name}{#4}%
\ifblank{#5}{\caption{#6}}{\caption[#5]{#6}}%
\endgroup%
\ifblank{#1}{}{\label{#1}}%
}

\LetLtxMacro\bionenumcaption\bitwonumcaption% todo

\renewcommand{\bicaption}[5][]{%
\ifblank{#2}{\caption{#3}}{\caption[#2]{#3}}%
```

```
\begin{LWR@figcaption}% later becomes \caption*
\LWR@isolate{#4} \thechapter.\the\value{\@captype}\CaptionSeparator\LWR@isolate{#5}%
\end{LWR@figcaption}
\ifblank{#1}{}{\label{#1}}%
}

\renewcommand{\bicontcaption}[3]{%
\contcaption{#1}%
\begingroup%
\csdef{\@captype name}{#2}%
\contcaption{#3}%
\endgroup%
}

% only in ccaption, not in memoir:
% \LetLtxMacro\longbitwonumcaption\bitwonumcaption%
% \LetLtxMacro\longbionenumcaption\bitwonumcaption%
% \LetLtxMacro\longbicaption\bicaption%

\RenewDocumentCommand{\subtop}{O{} O{} m}{%
\subfloat[#1][#2]{#3}%
}

\RenewDocumentCommand{\subbottom}{O{} O{} m}{%
\subfloat[#1][#2]{#3}%
}

\renewcommand{\contsubtop}{%
\ContinuedFloat\addtocounter{\@captype}{1}%
\subtop}

\renewcommand{\contsubbottom}{%
\ContinuedFloat\addtocounter{\@captype}{1}%
\subbottom}

\renewcommand{\subconcluded}{}

\let\subfigure\subbottom
\let\subtable\subtop

\let\contsubtable\contsubtop
\let\contsubfigure\contsubbottom

\newcommand{\newfloatentry}[4][\@empty]{TODO: newfloatentry}
\newcommand{\newfloatlist}[5][\@empty]{TODO: newfloatlist}
\newcommand{\newfloatenv}[4][\@empty]{TODO: newfloatenv}
\DeclareRobustCommand{\newfloatpagesoff}[1]{}
\DeclareRobustCommand{\newfloatpageson}[1]{}
\newcommand{\setnewfloatindents}[3]{}
```

§ 494.20 Final patchwork

```
\newlistof{tableofcontents}{toc}{\contentsname}
\newlistof{listoffigures}{lof}{\listfigurename}
\newlistof{listoftables}{lot}{\listtablename}
```

Change History

§ 495 ## Chg Hist

For the most recent changes, see page 1000.

v0.10
General: 2016/03/08 Initial version . . 1
v0.11
General: 2016/03/11 1
Added section: Operating-System portability. 206
Added section: Selecting the operating system. 118
Test Suite: MS-WINDOWS in README.txt 1
Test Suite: limages and index in README.txt 1
v0.12
\LWR@newhtmlfile: Bugfix: TOC with numbered files. 347
General: 2016/03/14 1
Global: Uses \p@(type) in float captions. 1
Test Suite: Sub-figures 1
v0.13
\CaptionSeparator: Fix for newer babel package. 465
\LWR@LwarpStart: \up and \fup . . . 365
General: 2016/03/24 1
Fix dollar-redefined bug for newer package. 884
Removed package: subfig 1
Test Suite: Ordinals, Subcaption . . . 1
v0.14
\LWR@htmlsectionfilename: Fix: Links to home page. 310
General: 2016/03/31 1
floatrow: Added. 692
Docs: Commands for a successful HTML conversion. 121
Docs: Commands into a warpprint environment. 119
Docs: Newclude limitations. . . . 163
Docs: Table: Cross-referencing data structures. 450
Docs: Table: Float data structures. 461
Docs: Trademarks section. 187
Docs: Troubleshooting cross-references. 182
Test Suite: Assigned cleveref name for Test Float. 1
Test Suite: Floatrow 1
v0.15
General: 2016/04/06 1
Added. 697
Ampersand (&): Fixed handling when passed as an argument. . . 399
Docs: Added warning icons for items needing special attention. 190
Docs: Clarify print/HTML output. 118
Docs: Moved the supported features table to the introduction. 75
Files: lwarp_formal.css added. 1
Fix: steps counter 697
Fixed & handling. 695
Test Suite: test_suite_formal.css file added. 1
v0.16
General: 2016/04/11 1
\titlingpage: Improved print-output spacing. 373
xfrac: Adjusted for the use of any font: 938
Added XeLaTeX, LuaLaTeX support. 191
Docs: Font and UTF-8 support. . 105
Docs: Moved location of \usepackage{lwarp}. 107
Docs: Text not converting. 182
Lwarp no longer selects fonts. 105, 215
Removed package: suffix 1
Test Suite: Improved titlingpage. 373
Test Suite: Lwarp no longer selects fonts. 1
Test Suite: Supports XeLaTeX, LuaLaTeX. 1
v0.17
\LWR@htmlsectionfilename: Fix: Links when entire doc is one HTML page. 310
General: 2016/04/14 1
mdframed: Added. 767

Test Suite: Fix: Print-version front-matter page numbers. 1
Test Suite: Mdframed 1
v0.18
\LWR@includegraphicsb: Add: svgz file extension. 718
em, ex, %, px dimensions preserved. 718
Fix: \linewidth, \textwidth, \textheight inside a minipage. 718
Improved HTML output linebreaks. 718
\LWR@myshorttoc: Reorganize \HomeHTMLFilename logic. 469
\LWR@newhtmlfile: sideTOC after title, improving responsive design. . . 346
\LWR@requesttoc: Reorganize \HomeHTMLFilename logic. 367
\LWR@subhyperref: Improved HTML output linebreaks. 458
\LWR@subhyperrefclass: Improved HTML output linebreaks. 458
\LWR@subinlineimage: Suppress extra space. 460
\hspace: \hspace supported. 555
General: 2016/05/19 1
File: lwarp.css: Improved TOC outline display. 1
Files: lwarp.css and lwarp_formal.css: Improved responsive design. 1
Microtype disabled during HTML generation 215
PDF Unicode input characters. . 204
Test Suite: Verse package 1
lateximage: pdfcrop: --hires added. 513
Reorganize \HomeHTMLFilename logic. 513
Suppress extra space. 513
verse: Supports verse, memoir packages. 918
minipage: Fix: \linewidth, \textwidth, \textheight inside a minipage. 533
v0.19
\HTMLFilename: Docs: Escape filename underscores. 310
\HomeHTMLFilename: Docs: Escape filename underscores. 310
\LWR@LwarpStart: Enabled \\ equal to \newline. 364
\LWR@doequation: MathJax support. 499
\LWR@doubledollar: MathJax support. 494
\LWR@filestart: lwarp_mathjax.txt loaded. 362
\LWR@minipagestartpars: Suppresses paragraph tags between minipages. 553
\LWR@subsingledollar: MathJax support. 488
\LateximageFontSizeName: Add: User-adjustable math/lateximage font size. 509
\hspace: Fix: \hspace length computations. 555
\minipagefullwidth: Added: No width tag for the next minipage in HTML. 532
\warpHTMLonly: Added. 214
\warpprintonly: Replaces \rowprintedonly. 214
\xfracHTMLfontsize: Added. . . . 938
General: 2016/06/08 1
CSS for table note item. 883
MathJax support added. 497, 503, 504
multirow: Added optional args. . 785
Adapts to tikz version. 884
Avoids MathJax. 487
cleveref: Loaded \AtEndPreamble. 528
Docs: Math options. 107
Docs: Table: Cross-referencing data structures, updated. 450
File: lwarp.css: tnoteitemheader added. 1
File: lwarp_mathjax.txt added. . . 1
Introduction: MathJax support mentioned. 72
Options: mathsvg and mathjax . 209
Supports colored \rule. 929
titleps: null \pagestyle and \thispagestyle for HTML. . . . 885
v0.20
\BlockClassSingle: Renamed from "LWR@htmldivclassline". 323
\HTMLDescription: Added \NewHTMLdescription. (Renamed in v0.30.) 333
\HTMLFilename: No longer escape underscores. 310
\HomeHTMLFilename: No longer escape underscores. 310
\InlineClass: Renamed from "inlineclass". 323
\LWR@LwarpStart: Fix: math cross references. 366
\LWR@closeparagraph: \unskip extra spaces. 327
No break tags in the start/end of a tabular. 327
\LWR@endofline: Fix: \\ 553

\LWR@filestart: Adds meta description. 362
\LWR@htmldivclass: Added optional style. 321
\LWR@htmlelementclass: Added optional style. 321
\LWR@htmlsectionfilename: HTMLFilename: removed additional trailing '-', and may be empty. 310
Sections called "Index" or "index" have an underscore prepended to their filenames if no prefix. 310
\LWR@includegraphicsb: Fix: \linewidth in a floatrow. 718
Fix: Expands filename. 718
\LWR@longtabledatacaptiontag: Fix: Pars in captions. 435
\LWR@section: Combined higher-level sections together into files. 353
\LWR@setOSWindows: Auto-detects operating system. 208
\LWR@subhtmlelementclass: Factored code. 320
\SetHTMLFileNumber: Add: Control file numbers. 310
\cpagerefFor: User-redefinable word for page references. 529
\dotfill: Inserts an ellipsis. 553
\hfill: Inserts a \qquad. 553
\hrulefill: Inserts a short rule. . . 553
\hspace: Add: Supports HTML thin breakable space. 555
\hyperindexref: Print mode provided in case hyperref not used. 478
\pageref: Added. 457
\tracinglwarp: Added. 226
General: 2017/02/09 1
afterpage: Added. 575
alltt: Added. 580
bookmark: Added. 611
caption and subcaption supported. 1
cleveref and referencing patches: Applied \AfterEndPreamble. . . . 528
draftwatermark: Added. 660
eso-pic: Added. 670
everypage: Added. 671
extramarks: Added. 672
fancyhdr: Added. 678
float: Improved float caption type handling. 690
hyperref: Additional user macros. 730
keyfloat: Added. 746
letterspace: User-interface emulated. 749
listings: Added. 753
ltcaption: Added. 761
lwarp-newproject: Added. 238
microtype: User-interface emulated. 779
needspace: Added. 793
nowidow: Added. 796
placeins: Added. 820
ragged2e: Added. 828
setspace: Improved support. . . 846
textpos: Added. 877
titleps: Added. 885
titlesec: Added. 888
titletoc: Added. 890
titling: Improved compatibility. . 892
tocloft: Added. 899
wallpaper: Added. 923
wrapfig: Added. 924
xetexko-vertical: Added. 937
Added @, <, > columns. 393
Added single-expansion data arrays. 306
Code factored into independent lwarp_html files. 566
Docs: Examples for generating HTML file names. 116
Docs: Improved index. 1
Enhanced titling support. 372
File: lwarp.css: Minor fixes for validation. 1
File: lwarpmk used to compile print, HTML, indexes, and latexmages. 1
Fix: \linewidth in a floatrow. . . 695
Moved sidebar and example code to test suite. 1
Page geometry set to 6in wide with large margins. 216
Parallel versions of aux files for print/HTML. 1
Removed reliance on make, grep, gawk. 1
Tabular: \unskip extra spaces. . . 393
Test Suite: HTML meta descriptions. 1
verbatim: Added. 382
BlockClass: Added optional style. . 322
Renamed from "blockclass". . . . 322
LWR@nestspan: Fix: Minipages inside a span. 318
v0.21
\LWR@LwarpStart: Changed latexmages to a .txt file. 364

\LWR@filestart: Skip title if not given. 362
\LWR@newhtmlfile: Skip title if not given. 346
\marginpar: Fixed source listing. . . 339
\marginparBlock: Fixed source listing. 339
General: 2017/02/23 1
fontenc: Added. 702
lwarpmk: Fix: **lwarpmk again** for WINDOWS. 286
lwarpmk: Fix: **lwarpmk limages** for WINDOWS. 286
lwarpmk: Fix: *lwarpmk* uses lateximages text file instead of shell script. 286
Add: Errors for misplaced packages. 191
Docs: Added internet class. 80
Docs: Added TeX2page, GladTeX. . 80
Docs: Installing on WINDOWS. . . . 86
File: lwarp_tutorial.txt added. . 90
v0.22
\LWR@parseDcolumn: Added tabular D column. 408
\LWR@parsebangcolumn: Added tabular ! column. 403
\LWR@parsetablecols: Unknown table column types become l. Added tabular D, !, X columns. . 410
\LWR@printmccoldata: Added tabular D, !, and X columns. 429
General: 2017/03/02 1
abstract: Added. 568
changepage: Added. 622
dcolumn: Added. 656
ftnright: Added. 708
geometry: Nullified commands. . 710
layout: Added. 749
lscape: Added. 760
mcaption: Added. 767
nameref: Added. 791
nextpage: Added. 793
parskip: Added. 812
showkeys: Added. 848
sidecap: Added. 848
tabularx: Added. 870
varioref: Supported. 130
verse: Added. 917
v0.23
\LWR@parsetablecols: Fix for vert bar column type. 410
\LWR@printmccoldata: Fix for vert bar column type. 429
General: 2017/03/02 1
v0.24
\LWR@htmlfileref: Fix: Index links while \tracinglwarp. 453
\hspace: Add: \hspace \fill converts to 2em 555
\hypertocfloat: List of floats responds to lofdepth, lotdepth. 475
General: 2017/03/15 1
floatrow: Support for subfig. . . . 692
subfig: Added. 863
tikz: For tikz v3.0.0 or later, auto-loads tikz babel library if necessary. 884
Docs: Filename underscore. 111, 124
Fix for inline images. 884
No longer preloads subcaption; conflicted with subfig. 218
picture: Fix for inline images. . . . 531
v0.25
\LWR@loadnever: Added the ability to prevent conflicting packages. . . 193
\addcontentsline: Handles theorems. 467
General: 2016/03/22 1
amsthm: Added. 587
ellipsis: Added. 662
emptypage: Added. 663
framed: Added. 706
lips: Added. 753
mdframed: Help avoid hyphenation. 769
ntheorem: Added. 797
showidx: Added. 848
theorem: Added. 878
Basic LaTeX theorems: improved css. 383
Docs: Adds credits for patched code. 1
Docs: Testing lwarp. 180
Fix: Allows XeLaTeX and LuaLaTeX to preload graphics and graphicx. 195
v0.26
General: 2017/03/31 1
lwarp.css: Improved responsive marginpar and marginblock. . . 245
cutwin: Added. 654
endnotes: Added. 665
floatflt: Added. 691
footmisc: Added. 702
footnotehyper: Added. 704
footnote: Added. 703
marginfix: Added. 766
marginnote: Added. 766
mparhack: Added. 782
pagenote: Supported as-is. 810

sidenotes: Added. 849
Docs: Improved MiKTEX install instructions. 85, 87
Dollar span avoided in a lateximage. 487
Footnotes now are LATEX boxes instead of pagenotes. 334
lateximage: Labels track page numbers of lateximages. 513
Print mode now uses a minipage of \linewidth. 513
picture: Fix for \makebox in picture. 531
v0.27
\LWR@footnotetext: Fix for table footnote par tags. 335
General: 2017/04/04 1
lettrine: Added. 750
microtype: Fix with XeLATEX, LuaLATEX. 779
soul: Added. 856
ulem: Added. 912
Docs: Installing utilities for MACOS. 88
Docs: Limitations of saveboxes. . 124
Page geometry modified to reduce line overflow. 216
v0.28
\@wrindex: Improved indexing. . . . 477
\HTMLAuthor: Added \HTMLauthor. (Renamed in v0.30.) 333
\LWR@LwarpEnd: If FormatEPUB or FormatWP, no bottom nav. 367
\LWR@LwarpStart: FormatWordProcessor forces single-file output. 364
\LWR@filestart: Adds HTML meta author. 362
\LWR@forcenewpage: Forces new PDF page before major environments. 314
\LWR@htmlcomment: Breaks ligatures in HTML comments. 320
\LWR@includegraphicsb: Adapts to graphics syntax. 718
\LWR@newhtmlfile: If FormatEPUB or FormatWP: skips headers, footers, nav. 346
\LWR@parsetablecols: Added L, C, R, J column types. 410
\LWR@startref: Removed space. . . 455
\chapter: If EPUB, prints footnotes before each section. 359
\hyperindexref: Improved indexing. 478
\textup: Fixed span class. 543
General: 2017/04/14 1
glossaries: Added. 712
graphics: Added. 713
tabularx: Fix for optional pos. . . 870
tabulary: Added. 871
lwarpmk: Add: **printglossary** and **htmlglossary** commands. . . . 286
Added boolean FormatEPUB. . . . 232
Added boolean FormatWP. 232
Added boolean HTMLDebugComments. 227
Added boolean HTMLMarkFloats, changed to WPMarkFloats as of v0.42. 232
Docs: Modfying lwarpmk and index processing. 180
File: lwarp_mathjax.txt: Updated CDN repository. 284
Forced oneside to maintain large right margin. 216
v0.29
\LWR@includegraphicsb: Fix: Error when no optional arguments. . . 718
General: 2017/04/15 1
*.lwarpmkconf: Add: language option for config files. 245
lwarpmk.conf: Add: language option for config files. 245
lwarpmk: Add: language option for config files. 286
Add: lwarpmklang option for lwarp. 210
Docs: Using a glossary 101
v0.30
\CSSFilename: Renamed from \NewCSS. 332
\HTMLAuthor: Renamed from \HTMLauthor. 333
\HTMLDescription: Renamed from \NewHTMLdescription. 333
\HTMLFirstPageTop: Renamed from \SetFirstPageTop. 331
\HTMLLanguage: Renamed from \MetaLanguage. 360
\HTMLPageBottom: Renamed from \SetPageBottom. 331
\HTMLPageTop: Renamed from \SetPageTop. 331
General: 2017/04/29 1
lwarp-newproject removed, and combined with lwarp. 238
lwarpmk: Add: xdyfile configuration option. 286
lwarpmk: Fix: *xindy* and *texindy* adjusted for *pdflatex*, *xelatex* and *lualatex*. 286

lwarpmk: Fix: *xindy* now used for print index generation with *latexmk*. 286
lwarpmk: `language` now used for both index and glossary generation. 286
File: `lwarp_html.xdy` renamed to `lwarp.xdy`. 283
Fix: `*.css` files only written in print mode. 245
Fix: `lwarp.xdy` only written in print mode. 283
Fix: `lwarp_mathjax.txt`: Only written in print mode. 284
Option `lwarpmklang` changed to `IndexLanguage`. 210
Option `OSWindows` replaces macro `\warpOSwindows`. 211
Option `xdyFilename` added. 210
Option *latexmk* replaces macro `\UseLatexmk`. 212
Options `HomeHTMLFilename` and `HTMLFilename` replace macros `\HomeHTMLFilename` and `\HTMLFilename`. 211

v0.31

General: 2017/05/15 1
keyfloat: Improved compatibility. 746

v0.32

`\RequirePackage`: Fix: Ignores blanks in package list. 222
General: 2016/06/09 1
glossaries: Prevent error with `\glo@name` not defined. 479
lwarpmk: Fix: **`io.lines()`** changed to **`file:lines()`** due to *luatex* changes. 286

v0.33

`\HTMLAuthor`: Fix: Provides empty default author if none given. . . . 333
`\LWR@loadbefore`: Fix: No `\PackageError` if already loaded. 193
`\LWR@parseatcolumn`: Fix: Column alignment with leftmost `@`. 403
`\LWR@tabledatasinglecolumntag`: Fix: Macros in tabular could cause extra data cell. 416
`\LWR@vspace`: Add: `\vspace` nullified. 556
`\StartDefiningTabulars`: Add: Avoids error: `Misplaced alignment tab character &`. . . 307
General: 2017/07/10 1
amsmath: Removed `fleqn` option. 580
fancyhdr: Fix: Optional args for `\lhead`, etc. 678
Add: Tabular at and bang columns now have their own HTML columns. 393
cleveref: Fix: Loaded `\AtEndPreamble`. 528
Fix: Incorrectly-inline math environments. 504
New handling of `&` to localize catcode changes. 393

v0.34

`\@fnsymbol`: Text symbols instead of math. 374
`\InlineClass`: Moved optional argument in front of mandatory. 323
`\LWR@htmldivclass`: Moved optional argument in front of mandatory. 321
`\LWR@htmlelementclass`: Moved optional argument in front of mandatory. 321
`\LWR@htmlelementclassline`: Moved optional argument in front of mandatory. 321
`\LWR@htmlspanclass`: Moved optional argument in front of mandatory. 319
`\LWR@nullfonts`: Improved font control. 545
`\LWR@restoreorigformatting`:
booktabs: Works inside `lateximage`. 480
Improved font control. 480
`\LWR@subhtmlelementclass`: Moved optional argument in front of mandatory. 320
`\LWR@tabledatacolumntag`:
booktabs: Works inside `lateximage`. 441
`\fboxBlock`: Added. 539
`\makebox`: Fix: Handles paren arg. . 537
General: 2017/08/08 1
babel-french: Adds fixed-width HTML spaces to punctuation. . . 316
balance: Added. 602
booktabs: Works inside `lateximage`. 445, 611
boxedminipage2e: Added. . . . 613
crop: Added. 654
enumerate: Added. 665
enumitem: Added, no longer required. 665
everyshi: Added. 672
fancybox: Added. 674
fancyvrb: Added, no longer required. 680
figcaps: Added. 686

filecontents: Required. Patched for morewrites. 217
floatpag: Added. 692
flushend: Added. 698
fullpage: Added. 708
hyperxmp: Added. 735
idxlayout: Added. 736
marginfit: Added. 765
mdframed: Improved mdtheorem patch. 774
moreverb: Added. 781
paralist: Added. 811
pdflscape: Added. 814
pdfsync: Added. 817
prelim2e: Added. 822
rotfloat: Added. 834
savetrees: Added. 835
shadow: Added. 847
syntonly: Added. 870
titleps: No longer required. 885
titleref: Prevented. 888
xmpincl: Added. 941
Added. 933
Docs: Horizontal space limitations. 1
Docs: Misplaced alignment character. 182
File: lwarp_mathjax.txt: Version change. 284
File: README.txt: updated. 1
Fix: Added the eqnarray environments. 504
Improved font control. 541
Lists refactored to remove enumitem requirement. 384
Verbatim refactored to remove fancyvrb requirement. 380
tabular: booktabs: Works inside lateximage. 446
lateximage: Fix: lateximage with minipage, \parbox, \makebox, \fbox, \framebox, \raisebox, \scalebox, \reflectbox. 513
BlockClass: Moved optional argument in front of mandatory. 322
fminipage: Added. 539
LWR@nestspan: Fix: Minipages, BlocksClass, and lists inside a span. 318
v0.35
General: 2017/08/08 1
Fix: \textbf and related. 541
v0.36
\LWR@HTMLsanitize: Fix for babel-french. 509
\LWR@HTMLsanitizeexpand: Fix for babel-french. 510
\LWR@closeparagraph: Extra HTML source space after paragraphs. . 327
\LWR@currenttextcolor: Fix for \rule when xcolor not loaded. . 551
\LWR@footnotetext: Extra HTML source space after paragraphs. . 335
Force HTML superscripts. 335
\LWR@nullfonts: Fix: Filenames while using MathJax. 545
\LWR@restoreorigformatting: siunitx: Improved super/subscripts in a lateximage. 480
\LWR@section: Improved spacing. . 353
\LWR@stoppars: Extra HTML source space after paragraphs. 329
\fbox: Fix: Uses \fboxrule and \fboxsep. 538
\framebox: Fix: Handles width and horiz position. 537
\makebox: Fix: Handles width and horiz position. 537
General: 2017/08/17 1
babel-french: Adjustements for French variants, load order, footnotes, ellipses. 316
footnote: Extra HTML source space after paragraphs. 703
siunitx: Fix for babel-french. . . . 521
siunitx: Improved symbol support. 851
transparent: Added. 908
upref: Added. 916
xcolor: Added \fcolorboxBlock, \colorboxBlock. 926
xcolor: Fix: Background none in print mode. 926
xcolor: Refactored \LWR@colorstyle. 929
xcolor: Uses \fboxrule and \fboxsep. 926
xcolor: \fcolorbox etc. now work inside lateximage. 926
Docs: Reorganized: Special cases and limitations. 121
Source: Improved formatting. . . . 1
tabular: Fix for babel-french. . . . 446
lateximage: Footnotes appear in regular text instead of the lateximage minipage. 513
v0.37
\include: Maintains independent aux files for HTML. 225
General: 2017/08/19 1

LaTeX accents: Added. 237
babel-french: Adjustment for load order. 316
color: Prevented. 651
comment: Maintains independent cutfiles for print, HTML. 214
siunitx: Improved symbol support. 851
textcomp: Improved support. . . 873
lwarpmk: Removes additional HTML aux files. 286
File handles reorganized. 224
v0.38
\@seccntformat: Added for appendix. 352
\ForceHTMLPage: Added. 350
\ForceHTMLTOC: Added. 350
\LWR@section: \part* starts a new HTML page, for appendix. 353
Modified spacing, uses \numberline. 353
\numberline: Added trailing \quad. 473
\part: Fix with article class. 358
General: 2017/08/27 1
appendix: Added. 591
arabicfront: Added. 593
chappg: Added. 623
color: Forces xcolor as well. 651
fix2col: Added. 687
fncychap: Added. 698
grffile: Added. 726
metalogo: Added. 777
nonumonpart: Added. 795
nopageno: Added. 795
pagenote: Option page disabled. 810
realscripts: Added. 828
relsize: Added. 831
romanbarpagenumber: Added. . 833
romanbar: Added. 833
scalefnt: Added. 836
siunitx: Removed from lwarp core. 851
textcomp: Removed from lwarp core. 873
tocbibind: Added. 897
xltxtra: Added. 941
lwarpmk: Added **print1** and **html1** actions. 286
Added \markboth, \sloppy, etc. . 314
Docs: Enhanced *Supported Features* table. 75
Docs: Index, tocbibind. 135
Docs: Starred sections. 132
v0.39
\@maketitle: titling version. 894
Native LaTeX version. 375
Removed minipages. 375, 894
Supports authblk with s of class oneauthor instead of tabular. 375, 894
\AddSubtitlePublished: Added. . . 377
\LWR@domulticolumn: Add: Optional vpos and # rows. 432
\LWR@restoreorigformatting: Appended with \appto instead of calling various macros. 480
\LWR@tabledatacolumntag: Don't start a data cell if see \TabularMacro. 441
\ResumeTabular: Added. 440
\TabularMacro: Added. 440
\multicolumnrow: Added. . . . 439, 786
\printauthor: Removed minipages. 372
Supports authblk with s of class oneauthor instead of tabular. 372
\thanksmarkseries: Removed minipage footnotes. 896
General: 2017/09/05 1
a4wide: Added. 568
a4: Added. 567
a5comb: Added. 568
addlines: Added. 575
anysize: Added. 591
authblk: Added. 600
bigdelim: Added. 608
bigstrut: Added. 609
ebook: Added. 661
fullwidth: Added. 709
midpage: Added. 780
multirow: Add: New optional vpos argument. 785
multirow: Add: Supports left/right border for bigdelim. 785
multirow: Fix: Long text argument. 785
supertabular: Added. 868
textarea: Added. 873
titling: Improved compatibility. . 892
titling: Removed extraneous center environments. 893
typearea: Added. 911
xtabular: Added. 943
zwpagelayout: Added. 947
Docs: Reorganized tabular discussion. 156
Titlepage \published and \subtitle removed. \AddSubtitlePublished restores. 376
titlepage: Clear pending footnotes. 371
Removed minipages. 371
titlingpage: Clear pending footnotes. 892

v0.40
\@chapcntformat: Added for tocbibind, anonchap. 352
\LWR@HTMLhline: Added. 445
\LWR@includegraphicsb: Add: Full \graphicspath support. 718
\LWR@nullfonts: Fix: Long arguments for expandable command. . . . 545
\LWR@restoreorigformatting: Improved LaTeX logos inside a lateximage. 480
Improved symbols inside a lateximage. 480
Nullified \InlineClass, etc. inside a lateximage. 480
\LWR@tabledatacolumntag: Fix for bigdelim: \ldelim, \rdelim. . . . 441
\chapter: Added support for quotchap. 359
\multicolumnrow: Fix: Adapts to older multirow and xparse. 439
\simplechapterdelim: Added for tocbibind, anonchap. 352
\underline: Added. 550
General: 2017/09/25 1
adjmulticol: Added. 574
anonchap: Added. 591
bigdelim: Improved documentation. 608
cuted: Added. 654
dblfnote: Added. 655
fnpos: Added. 699
graphics: Moved out of the lwarp core. 713
graphics: Restores \includegraphics and \DeclareGraphicsExtensions in a lateximage. 713
graphicx: Moved out of the lwarp core. 725
grffile: Directly supported. 726
midfloat: Added. 780
multirow: Improved bigdelim borders. 785
pfnote: Added. 818
quotchap: Added. 826
sectsty: Added. 845
stabular: Added. 861
tabls: Added. 870
textcomp: Additional symbols, improved XeLaTeX and LuaLaTeX support. 873
tocbibind: Improved for \simplechapter. 897
xfrac: No longer preloaded. 218
xltxtra: Fix for \showhyphens with XeLaTeX. 941
v0.41
\LWR@addcmidruletrim: Add: \cmidrule trims. 422
\LWR@clearmidrules: Add: \cmidrule trims. 419
\LWR@closetabledatacell: Add: Mute > for \bottomrule. 397
Fix: At/bang column with \multirow. 398
Fix: Cancel < for \multicolumn. . 398
\LWR@domulticolumn: Add: \cmidrule trims. 432
Added vertical rules. 433
\LWR@nullifyNoAutoSpacing: Fix: \NoAutoSpacing in a tabular with babel-french. 446
\LWR@parsebarcolumn: Added vertical rules. 405
\LWR@printatbang: Add: \cmidrule trims. 415
Add: Mute at and bang columns for \bottomrule. 415
\LWR@printbartag: Added vertical rules. 415
\LWR@subaddcmidruletrim: Added. 421
\LWR@subcmidrule: Add: \cmidrule trims. 420
\LWR@tabledatasinglecolumntag: Add: \cmidrule trims. 416
Add: Mute < for \bottomrule. . . 416
\LWR@tabularfinishrow: Unfinished tabular rows automatically filled. 400
\mcolrowcell: Added for \multicolumnrow cells. 444
General: 2017/10/07 1
multirow: Add: \cmidrule trims. 785
Added vertical rules. 785
Fix: < spec. 786
Improved rules. 611
tabular: Fix: \NoAutoSpacing in a tabular with babel-french. . . . 446
v0.42
\@ensuredmath: Improved \ensuremath. 495
\@textsubscript: Added. 550
\@textsuperscript: Added. 550
\LWR@HTMLhline: If FormatWP force explicit border. 445
\LWR@addformatwpalignment: If FormatWP add explicit style for cell alignment. 423
\LWR@addrulewidth: If FormatWP force explicit border. 422

\LWR@amsmathbody: Fix: Numbering and naming AMS math environments. 512
\LWR@amsmathbodynumbered: Fix: Numbering and naming AMS math environments. 512
\LWR@doequation: If FormatWP print LaTeX expression. 500
\LWR@domulticolumn: If FormatWP add cell alignment. 433
\LWR@doubledollar: If FormatWP print LaTeX expression. 494
Improved \ensuremath. 494
Improved line spacing with mathjax. 494
\LWR@floatbegin: If FormatWP add a text frame. 462
\LWR@floatend: If FormatWP add a text frame. 463
\LWR@htmlmathlabel: If FormatWP print LaTeX expression. 503
\LWR@includegraphicsb: Fix: Filename expansion. 718
If FormatWP, use explicit size. . . . 719
\LWR@rememberbtag: Fix: Numbering and naming AMS math environments. 511
\LWR@restoreorigformatting: Improved \ensuremath. 482
\LWR@subaddcmidruletrim: 0pt if no rule given. 421
\LWR@subsingledollar: If FormatWP print LaTeX expression. 488
\LWR@tabledatasinglecolumntag: If FormatWP add cell alignment. . . 417
\LaTeX: If FormatWP use explicit style. 560
\TeX: If FormatWP use explicit style. 560
\hspace: If FormatWP add \quads. . . 555
\listoffigures: Added boolean WPMarkLOFT. 471
\listoftables: Added boolean WPMarkLOFT. 471
\marginpar: If FormatWP emulate a wrapfig. 339
\marginparBlock: If FormatWP emulate a wrapfig. 339
\rule: If FormatWP add \quads. . . . 559
\tableofcontents: Added boolean WPMarkTOC. 470
\underline: If FormatWP, use explicit styles for \underline, etc. 550
General: 2017/10/30 1
\textbf and related: If FormatWP, use explicit styles for \textsc, etc. 541
algorithmicx: If FormatWP add \quads. 579
epigraph: If FormatWP add HTML styles. 666
fancybox: If FormatWP add HTML styles. 674
floatflt: Added width. 691
includegraphics: Fix: Class key. . . 717
keyfloat: If FormatWP add explicit HTML style. 748
moreverb: Simplified formatting of listings. 781
multirow: If FormatWP add cell alignment. 785
overpic: Added. 809
realscripts: Fix for subscripts in a lateximage. 828
sidenotes: If FormatWP add explicit HTML style. 850
siunitx: Improved \ensuremath. . . 852
soul: If FormatWP, add explicit styles. 856
textcomp: Improved \interrobangdown. 873
wrapfig: If FormatWP add explicit HTML style. 924
Added boolean WPMarkLOFT. . . . 233
Added boolean WPMarkMath. . . . 233
Added boolean WPMarkMinipages. 232
Added boolean WPMarkTOC. 232
Added boolean WPTitleHeading. 233
Docs: Added support page. 2
Docs: Improper \prevdepth. . . . 182
Docs: Reorganized math limitations 147
File: lwarp_mathjax.txt: Updated siunitx script. 284
Fix: Numbering and naming AMS math environments. 511
If FormatWP force explicit border. 611
If FormatWP, shift section headings. 233
tabbing: Added. 382
lateximage: Fix: Numbering and naming AMS math environments. 513
center: If FormatWP use explicit text-align. 518
minipage: Added boolean WPMarkMinipages. 535
If FormatWP add a text frame. . . . 534
eqnarray: Fix: Numbering and naming AMS math environments. 506
If FormatWP print LaTeX expression. 505
LWR@BlockClassWP: Added to factor code. 323

LWR@figcaption: If FormatWP forces italic captions. 466
v0.43
\LWR@domulticolumn: Fix for vertical rules. 433
Fix: Multicolumn trim. 432, 433
\LWR@maybeprintpendingfootnotes: Added FootnoteDepth. 338
\LWR@nullfonts: Fix: Nullify dollar inside filesnames. 545
\LWR@parsetablecols: Ignore spaces in col spec. 410
\LWR@printmccoltype: Added vertical rules. 428
\LWR@section: Fix: Expansion in comparison. 353
Fix: Math in section name. . 355, 357
Fix: Nullify fonts inside HTML comment. 354
\TabularMacro: \newcommand instead of \relax to fix supertabular and xtab. 440
\href: Made robust. 458
\nameref: Made robust. 457
\nolinkurl: Made robust. 459
\url: Made robust. 459
General: 2017/11/08 1
LWR@currentautosec: Added. . . . 353
breakurl: Added. 614
hyperref: Made robust. 730, 732, 734
hyperref: \Gauge added. 734
luatodonotes: Added. 763
todonotes: Added. 906
Added FootnoteDepth. 334
Docs: HTML settings table. 111
Docs: Reorganized HTML customization. 111
v0.44
\@currentlabelname: Adjustment for koma-script. 452
\HTMLTitle: Added. 333
\LWR@addformatwpalignment: Fix for multicolumn alignment if FormatWP. 423
\LWR@backgroundcolor: Added. . . . 929
\LWR@filestart: Add \HTMLTitle. . 363
Fix \HTMLAuthor. 362
\LWR@listitem: Added list and trivlist. 387
\LWR@patchlists: Added list and trivlist. 390
\LWR@strresult: Fix: \providecommand. 396
\LWR@textcurrentcolor: xcolor: Added \LWR@textcurrentcolor. 929
\addcontentsline: Automatic \LWR@newfloatanchor. 467
\chapter: Add preamble for koma-script. 359
\marginparBlock: Added. 339
\nopagecolor: xcolor: Fix for \nopagecolor. 931
\part: Add preamble for koma-script. 358
\title: Added \thetitle. 332
General: 2017/11/22 1
algorithmicx: Improved comment symbol. 579
atbegshi: Added. 597
cancel: Added. 617
changepage: Additional options. 622
easy-todo: Added. 660
fancyref: Added. 679
fixmetodonotes: Added. 689
fixme: Added. 688
fontenc: Allowed after lwarp. . . . 702
hang: Added. 727
ifoddpage: Added. 737
ltxtable: Added. 761
luatodonotes: Improved. 763
lwarp-patch-komascript: Added. 948
overpic: Fix: Groups for lateximages. 809
pdfsync: Fixes. 817
preview: Added. 822
scrextend: Added. 837
scrhack: Added. 839
scrlayer-notecolumn: Added. . . 841
scrlayer-scrpage: Added. 841
scrlayer: Added. 839
section: Added. 843
soulpos: Added. 858
soulutf8: Added. 858
supertabular: Fix for caption. . . 869
tikz: Fix: Groups for lateximages. 884
tocbasic: Added. 896
tocloft: Added \newlistentry. . 903
tocloft: Improved \newlistof. . . 904
tocstyle: Added. 905
todonotes: Improved. 906
todo: Added. 905
typearea: Added expert commands. 911
watermark: Added. 924
xcolor: Added \LWR@currenttextcolorstyle. . 929
xcolor: Added \LWR@findcurrenttextcolor. . . 929
xtab: Fix for caption. 944
Adjustment for koma-script. . . . 202

AMS environments: Fix: Groups for lateximages. 581
If pdfLaTeX, require T1 and UTF-8 encoding. 203
picture: overpic: Fix: Groups for lateximages. 531
list: Added list and trivlist. . . 387
LWR@nestspan: Added list and trivlist. 318
v0.45
\@currentHref: Added. 457
\@donoparitem: Modified for HTML. 385
\@item: Modified for HTML. 385
\@mklab: Modified for HTML. 385
\CSSFilename: Improved filenames with underscores. 332
\LWR@LwarpStart: Fix: Lateximages on incorrect pages with Mathjax. 366
\LWR@includegraphicsb: Improved URLs with underscores. 718
\LWR@newautoidanchor: Fix: No anchor if frozen autoid. 464
\LWR@notmemoirloadafter: Added. 192
\LWR@printpendingmpfootnotes: Added. 338
\LWR@startref: Fix: Labels with underscores. 456
\LWR@subhyperref: Improved URLs with underscores. 458
\LWR@subhyperrefclass: Improved URLs with underscores. 458
\LWR@sublabel: Fix: Labels with underscores. 454
\LWR@tabledatacolumntag: Fix: Empty line between rows. 444
\chapter: Add optional heading title for memoir. 359
\newpage: Added. 553
\nolinkurl: Fix: Underscore in URL. 459
\normalmarginpar: Added. 339
\reversemarginpar: Added. 339
\section: Add optional heading title for memoir. 359
\tableofcontents: Fix: Empty sidetoc. 470
Fix: Patch \AtBeginDocument. . . 470
\url: Improved URLs with underscores. 459
General: 2018/01/14 1
array: Added. 593
babel-french: Robust commands. 316
backref: Added. 602
breakurl: Fix: Underscore in URL. 614
changebar: Added. 621
cite: Added. 649
continue: Added. 653
endfloat: Added. 663
epigraph: Support for memoir. . 667
fancyvrb: Improvements. . . 680–682
flafter: Added. 689
fltrace: Added. 697
footnpag: Added. 705
fwlw: Added. 709
hanging: Added. 728
hyperref: Fix: Underscore in URL. 730, 731
lwarp-patch-memoir: Added. . . . 951
memhfixc: Added. 776
memoir: Added. 563
natbib: Added. 791
pagesel: Added. 811
prettyref: Added. 822
subfigure: Added. 867
subfig: Fix for subcaption end tag. 866
subfig: Fix: Math in subcaptions. 864
textfit: Added. 877
titleref: Added. 888
turnthepage: Added. 911
Allows memoir's preloaded packages. 195
Docs: xparse warnings. 164
Docs: Fix for double hyphens. . . . 88
Docs: Improved install instructions. 89
Docs: Improved MiKTEX install instructions. 85
Docs: Moved table so doesn't interfere with install docs. 84
File: lwarp_mathjax.txt: Allow MATHJAX inside tabbing. 284
File: lwarp_mathjax.txt: Allow MATHJAX inside verse. 284
Fix: Empty sidetoc. 470
Improved: Robust \, \ , and \textellipsis commands. 552
Separate LWR@thisautoidWP for word processor s. 464
thebibliography: Patched to emphasize titles. 480
minipage: Fix: Improper \prevdepth. 533, 535, 536
v0.46
\LWR@closeparagraph: Fix: Tabular empty lines. 328
\LWR@closeprevious: Fix: Stack unnesting. 314
\LWR@forcenewpage: Fix: Improper \prevdepth. 314
\LWR@lookforpackagename: Fix: Spaces in \usepackage. 221
\LWR@providelength: Added. . . . 205

\LWRPrintStack: Name changed from \PrintStack. 313
\makebox: Fix: Lateximage in a \makebox. 537
\popclose: Fix: Stack unnesting. . . 305
\pushclose: Fix: Stack unnesting. . 305
General: 2018/01/23 1
LWR@tabularpardepth added. . . 396
amsthm: Adapted to trivlist changes. 589
mdframed: Fixes for SVG math or lateximage in title. 771
mdframed: Fixes for footnotes. . 772
ntheorem: Adapted to trivlist changes. 797
theorem: Adapt to trivlist changes. 880, 881
tabular: Fix: Tabular empty lines. . 449
list: Fix: Stack unnesting. 388
v0.47
\LWR@HTML@caption@begin: Fix: Argument passed to \LWR@origcaption@begin. . . . 466
\LWR@LwarpStart: Fix for SVG math in \nameref. 367
\LWR@WPcell: Fix: Line wrap at HTML hyphen. 423
\LWR@createautosec: Fix: Line wrap at HTML hyphen. 351
\LWR@domulticolumn: Fix: Line wrap at HTML hyphen. 433
\LWR@floatbegin: Fix: Line wrap at HTML hyphen. 462
\LWR@htmlclosecomment: Add \mbox to prevent line breaks. 319
\LWR@newautoidanchor: Fix: Line wrap at HTML hyphen. 464
\LWR@printopenlist: Fix: Line wrap at HTML hyphen. 385
\LWR@startref: Fix: Line wrap at HTML hyphen. 455
\LWR@sublabel: Fix: Line wrap at HTML hyphen. 454
\LWR@subsingledollar: Added SVG math image baseline adjust and em sizing. 488
Fix: Line wrap at HTML hyphen. . 492
\captionlistentry: Fix: Line wrap at HTML hyphen. 467
\hypertoc: Fix: Line wrap at HTML hyphen. 474
\hypertocfloat: Fix: Line wrap at HTML hyphen. 475
\makebox: Fix: Line wrap at HTML hyphen. 537
General: 2018/01/30 1
adjmulticol: Fix: Line wrap at HTML hyphen. 574
blowup: Added. 610
caption: Added. 618
caption: Also loads lwarp-caption. 224
changepage: Fix for pagecheck macros. 622
endheads: Added. 664
epigraph: Fix: Line wrap at HTML hyphen. 666
hanging: Fix: Line wrap at HTML hyphen. 729
hang: Fix: Line wrap at HTML hyphen. 727
keyfloat: Fix for SVG math in captions. 747
midpage: Fix: Line wrap at HTML hyphen. 780
multirow: Fix: Line wrap at HTML hyphen. 785
multitoc: Added. 787
ntheorem: Fix: Line wrap at HTML hyphen. 801
realscripts: Fix: Line wrap at HTML hyphen. 828
scrextend: Fix: Line wrap at HTML hyphen. 837
sectionbreak: Added. 844
sidenotes: Fix for SVG math in captions. 849
subfig: Fix for SVG math in captions. 864
subfig: Fix: Support \nameref. . . 863
xurl: Added. 945
lwarpmk: *pdfcrop*: Removed hires option for improved crop accuracy. 286
lateximage: Added CSS style option. 513
Fix: Line wrap at HTML hyphen. . 517
center: Fix: Line wrap at HTML hyphen. 518
minipage: Fix: Line wrap at HTML hyphen. 534
flushleft: Fix: Line wrap at HTML hyphen. 518
flushright: Fix: Line wrap at HTML hyphen. 518
enumerate: Fix: Line wrap at HTML hyphen. 388
itemize: Fix: Line wrap at HTML hyphen. 388
LWR@BlockClassWP: Fix: Line wrap at HTML hyphen. 323

v0.48
\@@@setcpageref: Fix for new v0.21 of cleveref. . . . 529
\@@@setcref: Fix for new v0.21 of cleveref. . . . 528
\@@@setcrefrange: Fix for new v0.21 of cleveref. . . . 529
\@biblabel: Improved bibliography label. . . . 480
\@item: Honors \makelabel. . . . 385
\@maketitle: Fix: Errors with IEEEtran class. . . . 375
\LWR@LwarpStart: Adjusted space around captions. . . . 365
\LWR@ProvidesPackageDrop: Fix: Options with braces. . . . 223
\LWR@addtabularhrulecolor: colortbl: Added. . . . 424
\LWR@addtabularrulecolors: colortbl: Added. . . . 425
\LWR@closetabledatacell: colortbl: Added. . . . 398
\LWR@includegraphicsb: Fix: Virtual page size limited to a group. 718, 722
\LWR@lookforpackagename: Fix: Parsing similar package names. . 221
\LWR@newautopagelabel: Fix: TOC, LOF, LOT links. . . . 344
\LWR@newhtmlfile: Fix: TOC, LOF, LOT links. . . . 348
\LWR@nullfonts: Fix: \newline in title. . . . 545
\LWR@parsedrequirepackagenames: Fix: Parsing similar package names. . . . 219
\LWR@parsetablecols: Fix: Ignore optional tabular column arguments. . . . 411
\LWR@restoreorigformatting: Fix: Spacing in SVG math, lateximage, Ti*k*z. . . . 481
\LWR@section: Fix: TOC, LOF, LOT links. . . . 357
\LWR@tabledatasinglecolumntag: colortbl: Added. . . . 417
\LWR@textcurrentfont: Added. Improves font control. . . . 548
\bibliography: Fix: \BaseJobname for bibliography. . . . 479
\centerline: Added. . . . 519
\l@part: Adapts to classes without \part. . . . 475
\leftline: Added. . . . 519
\mbox: Nullified for HTML. . . . 537
\rightline: Added. . . . 519
\thempfootnote: Removed \itshape. 337
General: 2018/02/14 . . . 1
acronym: Added. . . . 573
acro: Added. . . . 571
chapterbib: Added. . . . 624
colortbl: Added. . . . 413, 424, 651
fancyref: Now directly supported. 679
hypcap: Added. . . . 729
hypernat: Added. . . . 730
hyperref: \texorpdfstring now uses the TEX string. . . . 734
luatodonotes: Improved \todototoc. . . . 763
siunitx: Changes fraction to symbol. . . . 853
siunitx: Improved SVG math. 852, 853
siunitx: Improved color output. . 852
stfloats: Added. . . . 861
todonotes: Improved \todototoc. 906
vmargin: Added. . . . 920
xfrac: Fix: Added groups around super/subscripts to localize LWR@nestspan changes. . . . 938
Docs: Converting an existing document. . . . 104
Improved font control. . . . 548
tabular: colortbl: Added. . . . 447
lateximage: Print mode boxed to natural width. . . . 517
abstract: Allow optional name. . . . 378
v0.49
\LWR@addtabularcellcolor: xcolor: Added tabular row colors. . . . 427
\LWR@domulticolumn: xcolor: Added tabular row colors. . . . 434
\LWR@printlength: Fix: Group printlen changes. . . . 218
\affiliation: Fix: Adapts to classes which already provide. . . . 370
\href: Fix: Adapt to classes. . . . 458
\noalign: Fix: \noalign inside tabular. . . . 445
\url: Fix: Adapt to classes. . . . 459
General: 2018/02/19 . . . 1
amsmath: Fix: Patches for \eqref. 580
eso-pic: Fix for \AddToShipoutPicture. . . . 671
figsize: Added. . . . 686
fnlineno: Added. . . . 699
hypdestopt: Added. . . . 730
hyphenat: Added. . . . 736
lineno: Added. . . . 751
luacolor: Added. . . . 762
pagegrid: Added. . . . 810
pdfrender: Added. . . . 817

resizegather: Added. 833
vertbars: Added. 919
vwcol: Added. 921
xcolor: Added tabular row colors. 413, 935
Fix: Adapt to classes. 552
v0.50
\@ensuredmath: Fix: Use lateximage even if MathJax. 495
Improved SVG math alt tags. . . . 495
\LWR@atbeginverbatim: Improved column alignment. 381
\LWR@doequation: Improved SVG math display. 500
\LWR@doubledollar: Improved SVG math alt tags. 494
Improved SVG math display. 494
\LWR@footnotetext: Robustify macros. 336
\LWR@htmlrefsectionfilename: Fix: SVG math in a section name. . . 311
\LWR@newhtmlfile: Fix: SVG math in a section name. 347
\LWR@nullfonts: Fix: \underline in sectioning file name. 547
\LWR@overline: Added. 551
\LWR@subsingledollar: Fix: Use lateximage even if MathJax. . . . 488
Improved SVG math alt tags. . . . 488
MD5 hash avoids duplicate SVG math. 492
\LWR@vspace: Robustify macros. . . . 556
\newline: Robustify macros. 553
\textsubscript: Robustify macros. 550
\textsuperscript: Robustify macros. 550
General: 2018/03/03 1
lwarp.css: Improved SVG display math centering. 245
lwarp_one_limage.txt: Added. . 284
amsmath: Fix: Upright tags for SVGmath. 580
axodraw2: Added. 601
bytefield: Added. 616
dblfloatfix: Added. 655
diagbox: Added. 656
epstopdf: Added. 668
listings: Force flexible columns. . 754
morefloats: Added. 781
nonfloat: Added. 795
ntheorem: Fix: Not standard nor amsthm selected. 803
pbox: Added. 813
phfqit: Added. 819
schemata: Added. 836
siunitx: Fix: Loads xcolor. 851
siunitx: Improved SVG math alt tags. 853
siunitx: Improved units. 521, 851, 854
xy: Added. 946
lwarpmk: Error if lateximages.txt does not exist. 286
lwarpmk: Error if lwarpmk.conf points to lwarp. 286
lwarpmk: Improved error messages. 286
lwarpmk: MD5 hash avoids duplicate SVG math. 286
lwarpmk: Multiprocess support making lateximages. 286
AMS environments: Improved SVG math display. 581
Fix: Load fontspec if necessary. . 215
Robustify macros. 548
lateximage: Fix: SVG math in a section name. 515
MD5 hash avoids duplicate SVG math. 514, 517
eqnarray: Improved SVG math display. 506
v0.51
\@ensuredmath: Hashes \ensuremath. 495
\@item: Restored list label space. . . 386
\LWR@HTMLsanitize: Fix: Escapes double quotes. 509
\LWR@HTMLsanitizeexpand: Fix: Escapes double quotes. 510
\LWR@LwarpStart: MathJax: Nullifies \ensuremath. 367
\LWR@addbaselinemarker: Improved SVG math baseline. 487
\LWR@atbeginverbatim: Adds vertical offset. 381
\LWR@customizeMathJax: MathJax: Nullifies \ensuremath. 345
\LWR@doequation: Fix: \addcontentsline inside SVG math. Provides an autoid anchor. 500
\LWR@doubledollar: Fix: \addcontentsline inside SVG math. Provides an autoid anchor. 494
\LWR@findcurrenttextcolor: Added \LWR@findcurrenttextcolor when no xcolor. 551
\LWR@newautoidanchor: Fix: No autoid is inside a lateximage. . . 464
\LWR@newhtmlfile: MathJax: Nullifies \ensuremath. 349

\LWR@subsingledollar: Fix: \ensuredmath inside SVG image. 489
Fix: lateximage inside $\mathcal{AMS}$ \text. 490
Fix: Honors text font around SVG math. 489
Fix: SVG math with enclosed lateximage. 489
Improved SVG math baseline. . . . 490
SVG math baseline improved with invisible rule at corner. 493
Typeset SVG math only once during measurement. 490
\LWR@textcurrentcolor: xcolor: \LWR@textcurrentcolor if xcolor not loaded. 551
\addcontentsline: Add missing support for float mechanism if necessary. 468
No anchor ID if inside SVG image. 468
\displaymathnormal: Processing for complicated display math. . . . 502
\displaymathother: Processing for complicated display math. . . . 502
\textcolor: Fix: SVG math color. . . 930
General: 2018/03/24 1
HTML entity used for text dollar. . 487
lwarp_one_limage.txt: *pdftocairo* -noshrink added. 284
afterpackage: No longer required. 217
chemfig: Added. 624
chemformula: Added. 625
chemgreek: Added. 630
chemmacros: Added. 630
chemnum: Added. 648
epstopdf-base: Added. 668
fancybox: Fix: Optional tag for \item in a span. 677
grid: Added. 726
listings: Forces cleared options. . 754
ltxgrid: Added. 761
mhchem: Added. 778
tikz: Fix for \tikz macro. 884
tikz: Fix for tikz with optional argument. 884
titling: Fix for \thanks mark. . . . 894
lwarpmk: *pdfcrop*: Restored hires option. 286
lwarpmk: *pdftocairo* -noshrink added. 286
AMS environments: Fix: \addcontentsline inside SVG math. Provides an autoid anchor. 581
Docs: tikz limitations. 153
Docs: Multiple authors and affiliations. 131
Docs: Things to avoid. 122
Docs: Updated Converting an existing document. 104
Fix: Remember original \# in case is redefined. 235
lateximage: Added additional hashing option. 513
Fix: lateximage inside $\mathcal{AMS}$ \text. 513
Processing for complicated display math. 516
alignat: Fix: Added. 585
eqnarray: Fix: \addcontentsline inside SVG math. Provides an autoid anchor. 506
LWR@displaymathother: Processing for complicated display math. . . 496
LWR@equationother: Processing for complicated display math. . . . 497
v0.52
\@ensuredmath: Improved hashing expansion. 495
\@mpfootnotetext: Fix: Paragraph handling. 337
\CustomizeMathJax: Added. 345
\LWR@addbaselinemarker: Warnings if lwarp_baseline_marker.png is not present or if graphicx/s not loaded. 487
\LWR@customizedMathJax: Added. . 344
\LWR@doequation: Fix: equation* now based on equation instead of displaymath. 500
Fix: equation* with split. . . . 499
\LWR@filenamenoblanks: Fix: \FileDepth with non-utf8 encoding. 343
\LWR@footnotetext: Fix: Paragraph handling. 336
\LWR@nullfonts: Fix: \texorpdfstring in section names. 547
\LWR@section: Fix: Footnote numbering: Limited HTML comment if starred. 354
Fix: Footnote numbering: Use short TOC entry for HTMLDebug comments. 354
\LWR@subsingledollar: Added user-adjustable SVG math font scaling. 491

\LateximageFontScale: Added user-adjustable SVG math font scaling. 509
\href: Fix: #, %, &, ~, _ in URL. 458
\nolinkurl: Fix: #, %, &, ~, _ in URL. 459
\theHTMLTitleSeparator: Fix: \FileDepth with non-utf8 encoding. 361
\url: Fix: #, %, &, ~, _ in URL. 459
General: 2018/04/01 1
breakurl: Fix: #, %, &, ~, _ in URL. . 614
endfloat: Updated for v2.6. 663
fancyvrb: Initial support for \VerbatimFootnotes. 674, 680
hyperref: Fix: #, %, &, ~, _ in URL. 730–733
nicefrac: Added. 793
url: Added. 917
lwarpmk: Fix: Memory overflow when spawning tasks. 286
lwarpmk: Fix: Skip image generation if from page 0. 286
Changed FootnoteDepth default to \subsbusection. 334
Docs: Improved install instructions. 86
Fix: MathJax script line wraps. Reduced right margin. 216
If pdfLaTeX, allow other input encoding. 203
pkggraphics: Added defaults. 715, 716
pkggraphicx: Updated for v1.1a. . 716
pkggraphicx: Updated for v1.1b. . 716
Restore \kill in a lateximage. . 760
tabbing: Fix to allow inside lateximage. 382
lateximage: Fix for hash expansion. 514
v0.53
General: 2018/04/01 1
lwarpmk: Added **lwarpmk cleanlimages**. 286
lwarpmk: Added warning for corrupted images. 286
Docs: **lwarpmk cleanlimages**. . . 102
Docs: **lwarpmk pdftohtml**. . . . 102
v0.54
\LWR@afterendverbatim: Added vspace argument. 382
\LWR@atbeginverbatim: Improved column alignment. 381
\LWR@earlyloadnever: Added. 194
\LWR@endfloatalignment: Honor \centering, etc. in floats. 464
\LWR@floatalignment: Honor \centering, etc. in floats. 464
\LWR@floatbegin: Honor \centering, etc. in floats. 462
\LWR@floatend: Honor \centering, etc. in floats. 463
\LateximageFontSizeName: Defaults to normalsize. 509
\centering: Added debug comment. 518
\raggedleft: Added debug comment. 519
\raggedright: Added debug comment. 519
General: 2018/04/22 1
*.lwarpmkconf: Option IndexLanguage changed to xindyLanguage. 245
*.lwarpmkconf: Option pdftotextEnc added. 245
*.lwarpmkconf: Option xdyFilename changed to xindyStyle. 245
*.lwarpmkconf: Option xindyCodepage added. 245
lwarp.css: Fix: Text-decoration-skip: auto. . . . 245
lwarpmk.conf: Option IndexLanguage changed to xindyLanguage. 245
lwarpmk.conf: Option pdftotextEnc added. 245
lwarpmk.conf: Option xdyFilename changed to xindyStyle. 245
lwarpmk.conf: Option xindyCodepage added. 245
bibunits: Added. 608
chngpage: Added. 649
forest: Added. 705
glossaries: Fix when not using babel or polyglossia. 712
gridset: Added. 727
hyperref: Fix: \hyperref and \hyperlink with special chars in text. 732, 733
hyperref: Fix: \ref in \hyperref and \hyperlink caused nested link. 732, 733
lwarp-patch-memoir: Update for v3.7g. 955
magaz: Added. 764
ragged2e: Fix: \centering, etc. . 828
textcomp: Fix for \textperthousand. 873
tikz: Fixes for \pgfpicture, minipages, fit, align, font. 884
lwarpmk: Added pdftotextenc. . 286

lwarpmk: Added xindycodepage. 286
lwarpmk: Changed language to xindylanguage. 286
lwarpmk: Changed xdyfile to xindystyle. 286
lwarpmk: Improved error if configuration file does not exist. 286
lwarpmk: Increased prominence for error for an unknown command. 286
lwarpmk: Verifies HTML version exists before lwarpmk limages. . 286
lwarpmk: Verifies image references before lwarpmk limages. 286
Add: pdftotextEnc. 210
Add: xindyCodepage. 210
Added early check for disallowed packages. 194
Docs: BibTeX. 133
Docs: Macros in sectioning names. 122
Never load aecompl. 194
Option IndexLanguage changed to xindyLanguage. 210
Option xdyFilename changed to xindyStyle. 210
verse: Fix: Line spacing. 379
v0.55
\LWR@LwarpStart: Fix: Overfull boxes in lateximages. 365
\LWR@floatbegin: Fix: Float optional args. 462
\LWR@nullfonts: Removed extraneous space which appeared in file links. 547
\phantomsection: Fix: \ForceHTMLTOC with \phantomsection. 559
General: 2018/04/26 1
clrdblpg: Added. 650
Fix: \centering, etc. for koma-script. 462
Fix: QED symbols in lateximage. 590, 808
v0.56
\LWR@addcdashline: arydshln: Added. 423
\LWR@addmulticolvertrulecolor: Adds support for dashed vertical rules. 431
Adds support for double vertical rules. 431
\LWR@addtabularhrulecolor: Adds support for arydshln dashed rules. 424
Adds support for double \hlines and \midrules. 424
\LWR@addtabularrulecolors: Adds support for dashed vertical rules. 425
Adds support for double vertical rules. 425
\LWR@closeparagraph: Added support for parnotes. 328
\LWR@domulticolumn: Adds support for dashed vertical rules. 433
Adds support for double vertical rules. 433
\LWR@floatbegin: Adds a <class> per float package style. 462
\LWR@multicolpartext: Fix: \multicolumn parameters. . . . 429
\LWR@openparagraph: Added support for parnotes. 326
\LWR@parsebarcolumn: Adds support for double vertical rules. 405
\LWR@parsecoloncolumn: arydshln: Added. 406
\LWR@parsesemicoloncolumn: arydshln: Added. 406
\LWR@parsetablecols: Added array W column. 412
\LWR@printmccoldata: Added array W column. 430
Fix: \multicolumn parameters. . . 430
\LWR@printmccoltype: Added array W column. 428
Adds support for dashed vertical rules. 428
Adds support for double vertical rules. 428
\LWR@tabledatacolumntag: Fix: \morecmidrules 443
\LWR@textcurrentfont: Added span.textbf, etc. 548
General: 2018/05/12 1
*.lwarpmkconf: Records --shell-escape. 245
lwarp.css: Added div.textbf, etc. 245
lwarp.css: Added span.textbf, etc. 245
lwarpmk.conf: Records --shell-escape. 245
arydshln: Added. 393, 594
lua-check-hyphen: Added. 762
paralist: Fixes for compactenum, compactitem, compactdesc. . . . 811
parnotes: Added. 811
quoting: Added. 827
tocenter: Added. 898
underscore: Added. 915

lwarpmk: Added `lwarpmk pdftosvg`. 286
lwarpmk: Supports `--shell-escape`. 286
Added `\thinspace`. 552
Docs: **`lwarpmk pdftosvg`**. 102
`LWR@blocktextcurrentfont`: Added `div.textbf`, etc. 548
v0.57
`\BlockClassSingle`: Improved print/HTML output selection. . . 323
`\InlineClass`: Improved print/HTML output selection. 323
`\LWR@customizeMathJax`: MathJax: Supports `\footnote`, `\footnotemark`. 345
`\LWR@ref@ignorestar`: subcaption: Fix: `\subref`. 456
`\LWR@subhyperref`: Fix: Text catcodes. 458
`\LWR@subhyperreftext`: Fix: Text catcodes. 458
`\LWR@subsingledollar`: Fix: Dynamic inline math expressions. 489, 490, 492
`\LWR@vspace`: Improved print/HTML output selection. 556
`\StartDefiningMath`: Added. 307
`\boxframe`: xcolor: Fix: Colored `\boxframe`. 934
`\colorbox`: xcolor: New system for switching print and HTML outputs. 931
`\colorboxBlock`: xcolor: New system for switching print and HTML outputs. 931
`\fboxBlock`: Improved print/HTML output selection. 539
`\fcolorbox`: xcolor: New system for switching print and HTML outputs. 932
`\framebox`: Improved print/HTML output selection. 537
`\href`: Fix: Text catcodes. 458
`\inlinemathother`: Added. 308
`\listof`: Fix: Provide `\l@name` if not defined. 472
`\makebox`: Improved print/HTML output selection. 537
`\mathimagename`: Added. 486
`\mbox`: Improved print/HTML output selection. 537
`\multicolumnrow`: multirow: Improved print/HTML output selection. 786
Improved print/HTML output selection. 439
`\newfloat`: rotfloat: Added float styles. 835
rotfloat: Fix for listof sideways floats. 835
`\packagediagramname`: Added. . . . 486
`\parbox`: Improved print/HTML output selection. 536
`\raisebox`: Improved print/HTML output selection. 541
`\reflectbox`: Improved print/HTML output selection. 725
`\resizebox`: Improved print/HTML output selection. 725
`\rotatebox`: Improved print/HTML output selection. 723
`\rule`: Fix: Colored rules. 558
`\scalebox`: Improved print/HTML output selection. 724
`\textcolor`: xcolor: New system for switching print and HTML outputs. 930
General: 2018/06/06 1
`lwarp.css`: Added ruled, boxed, boxruled floats. 245
`lwarp.css`: Increased float vertical margins. 245
algorithm2e: Added. 575
bigdelim: Improved print/HTML output selection. 608
breakurl: Fix: Text catcodes. . . . 614
colortbl: New system for switching print and HTML outputs. . . 651, 652
ellipsis: Added `\midwordellipsis`. 662
errata: Added. 669
float: Added float styles. 690
float: Fix: Do not pre-define `\l@name`. 690
ltablex: Added. 760
marginnote: Fix: Long optional argument. 766
multirow: Improved print/HTML output selection. 785
register: Added. 830
subcaption: Fix: `\subref`. 731
trimclip: Added. 909
vowel: Added. 920
xellipsis: Added. 936
xfrac: Improved print/HTML `\scalebox` control. 938
xltabular: Added. 940
xpiano: Added. 942
lwarpmk: Improved code factoring. 286

lwarpmk: Improved error handling. 286
Docs: Recompiling *lwarpmk* or CSS files. 180
Docs: Recreating the index for lwarp source. 177
New system for switching print and HTML outputs. 230
minipage: Improved print/HTML output selection. 533
BlockClass: Improved print/HTML output selection. 322
fminipage: Improved print/HTML output selection. 539
LWR@BlockClassWP: Improved print/HTML output selection. . . 323
v0.58
\LWR@HTML@caption@begin: Improved print/HTML output selection. . . 466
\LWR@HTML@caption@end: Improved print/HTML output selection. . . 467
\LWR@HTML@ref: Improved print/HTML output selection. 456
\LWR@doindexentry: Adds support for \see, \seealso, \emph, \textbf, etc. 478
\LWR@hyperindexrefnullified: Adds support for \see, \seealso, \emph, \textbf, etc. 478
\LWR@indexitem: Accepts optional arg for repeatindex. 477
\dotfill: Improved print/HTML output selection. 553
\hfill: Improved print/HTML output selection. 553
\hrulefill: Improved print/HTML output selection. 553
\hyperindexref: Adds support for \see, \seealso, \emph, \textbf, etc. 478
\printindex: Fix: Extra \newpage to flush pending \index writes. . . . 765
General: 2018/07/07 1
*.lwarpmkconf: Added option makeindexstyle. 245
*.lwarpmkconf: Added options makeindex and xindy. 245
*.lwarpmkconf: Generated \AtBeginDocument. 245
lwarp.xdy: Requires makeindex.xdy. 283
lwarp.xdy: Supports bold, italic. 283
lwarp_html.ist: Added. 282
lwarpmk.conf: Added option makeindexstyle. 245
lwarpmk.conf: Added options makeindex and xindy. 245
lwarpmk.conf: Generated \AtBeginDocument. 245
array: Improved print/HTML output selection. 593
attachfile2: Added. 598
attachfile: Added. 597
cases: Added. 621
imakeidx: Added. 738
index: Added. 742
intopdf: Added. 743
lwarp-patch-komascript: Modified indexing. 949
lwarp-patch-memoir: Fix for \specialindex. 971
lwarp-patch-memoir: Fix for multiple indexes. 971
makeidx: Added. Moved from lwarp core. 765
memoir: Fix for \firsthlline, \lasthline. 440
memoir: Fix for booktabs. 445
pdfpages: Added. 814
pdfx: Added. 818
repeatindex: Added. 832
splitidx: Added. 859
textcomp: Improved print/HTML output selection. 873
lwarpmk: Added makeindex and xindy options. 286
lwarpmk: Added -p option for project name. 286
lwarpmk: Added optional list of names for **lwarpmk printindex** and /cmdslwarpmk htmlindex. . 286
lwarpmk: Glossary generation now uses *makeglossaries*. 286
lwarpmk: **lwarpmk clean** removes all *.ind and *.idx files. 286
Added makeindex option. 211
Added xindy option. 212
Added option makeindexStyle. . 210
Docs: Index, *makeindex*, imakeidx. 135
Docs: Misplaced \omit. 182
Fix: memoir and ccaption. . . . 195
Improved print/HTML output selection. 552
Replaced each \csuse with \@nameuse to force error if undefined. 1
tabbing: Improved print/HTML output selection. 382

v0.59
\LWR@addbaselinemarker: Uses .eps if DVI *latex*. 487
\LWR@includegraphicsb: Fix: Expand filename. 719
Now works with .pdf and .eps filename extensions. 718
\LWR@latexmkcmd: Fix: --shell-escape with *latexmk*. . 240
\LWR@writeconf: Compilation commands now preassigned by lwarp instead of being computed by *lwarpmk*. 244
\[: Fix with \displaymathnormal. . 495
General: 2018/09/07 1
Slunits: Added. 850
accsupp: Added. 571
amsmath: Moved from the lwarp core. 580
asymptote: Added. 596
axessibility: Added. 601
breqn: Added. 614
bxpapersize: Added. 616
canoniclayout: Added. 618
chemformula: Fix for \NMR. 645
draftcopy: Added. 659
epstopdf-base: Improved. 668
epstopdf: Improved. 668
fnbreak: Added. 698
nccfancyhdr: Added. 792
pdftricks: Added. 818
pst-eps: Added. 824
pstricks: Added. 825
units: Added support for MathJax. 915
xunicode: Added. 944
lwarpmk: Added **lwarpmk epstopdf**. 286
lwarpmk: Consolidated compiling options into printlatexcmd and HTMLlatexcmd. 286
lwarpmk: Double insead of single-dashed --shell-escape option. 286
lwarpmk: Error if lwarpmk.conf format changed. 286
lwarpmk: Warning if operating system changed. 286
Added option dvipdfmx. 212
Added option dvipdfm. 212
Added option dvips. 212
Docs: **lwarpmk epstopdf**. 102
File: lwarp_mathjax.txt: Fix: Removed chapter number from tagged non-numeric MathJax equations. 284
File: lwarp_mathjax.txt: Updated to MathJax v2.7.4. 284
picture: Added an alt tag. 531
v0.60
\LWR@clearmidrules: tabular: Fix for midrules. 419
\LWR@parsenormalcolumn: tabular: Improved memory management: Not using xstring. 407
\LWR@tabledatasinglecolumntag: tabular: Improved memory management: Not using xstring. 417
\LWR@tabularendofline: Fix: Slowdown for long tables. 402
General: 2018/09/19 1
tabular: Improved memory management: Global boolean. 394, 395
tabular: Improved memory management: Not using xstring. 396, 785
2up: Added. 567
booklet: Added. 610
bophook: Added. 613
diagbox: Fix for par tags. 657
draftfigure: Added. 659
fancytabs: Added. 680
fullminipage: Added. 708
grid-system: Added. 726
layaureo: Added. 748
leading: Added. 749
listings: Fix for HTML entities. . . 754
listings: Fix if inside a list. . . 754, 757
thumbs: Added. 883
thumb: Added. 883
widows-and-orphans: Added. . . 924
v0.61
\DeclareGraphicsExtensions: Fix: EPS for DVI LaTeX. 713
\LWR@HTMLLatexCmd: Added HTMLLatexCmd option. 244
Added PrintLatexCmd option. . . 244
\LWR@addcompilecmd: Removed spaces. 240
\LWR@closetabledatacell: Fix: Par tags in tabular. 397
\LWR@hyperindexrefnullified: Made robust, ... 478
\LWR@includegraphicsb: Fix: EPS for DVI LaTeX. 719
Set keys before using filename, for epsfig. 719
\LWRopseq: Added spaces. 207
\RequirePackage: Support up to 20 packages. 222

\inlinemathnormal: Changed name from \StopDynamicMath to \inlinemathnormal. 308
\inlinemathother: Changed name from \StartDynamicMath to \inlinemathother. 308
\listof: Fix: newfloat lists. 472
\lwarpsetup: Added. 208
General: 2018/10/13 1
lwarp.css: Footnotes text align left. 245
lwarp.css: Minipage table and footnotes: tighter margin. 245
chkfloat: Added. 649
cmdtrack: Added. 651
copyrightbox: Added. 653
dprogress: Added. 659
epsfig: Added. 667
lua-visual-debug: Added. 762
pdfprivacy: Added. 817
psfragx: Added. 823
psfrag: Added. 823
pstool: Added. 824
refcheck: Added. 829
srcltx: Added. 860
srctex: Added. 860
supertabular: Fix for caption w/o opt arg. 869
thinsp: Added. 882
threadcol: Added. 882
uspace: Added. 917
vpe: Added. 921
xbmks: Added. 925
xtab: Fix for caption w/o opt arg. 943
Added HTMLLatexCmd option. . . . 211
Added PrintLatexCmd option. . . 211
Docs: \tracinglwarp 226
Docs: HTML entities. 122
Docs: Compiling using custom shell commands. 166
Docs: Fonts. 105
Docs: HTMLDebugComments 111, 226
Docs: Multiple indexes. 187
Don't write configuration files if processing pstool image. 238
Spaces redefined \AtBeginDocument. 552
v0.62
\@partcntformat: Added for ctex. . 352
\@partnameformat: Added for ctex. 352
\InlineClass: Added optional word-processing style. Replaces \LWR@HTMLtextstyle. 323
\LWR@PreloadedPackage: Added. . . 519
\LWR@ProvidesPackagePass: Fix: Unknown option error. 223
\LWR@endofline: Extra space if optional arg. 553
\LWR@filestart: Refactored. 363
\LWR@includegraphicsb: Fix: FormatWP. 719
Fix: Filename expansion. 721
\LWR@isolate: Added. 205
\LWR@textcurrentfont: Added print version. 551
Tracks depth to avoid nesting repeated font changes. 548
\colorboxBlock: Fix: Horiz white space. 931, 932
\fcolorbox: Fix: No longer requires xifthen. 525
\fcolorboxBlock: Fix: Horiz white space. 932, 933
\l@chapter: Don't define if no \chapter. Fix for algorithm2e. . 475
\slshape: Added. 549
\textup: Fixed WP span class. . . . 543
\theHTMLSection: Added. 361
\theHTMLTitleSection: Added. . . . 361
\theHTMLTitleSeparator: Refactored. 361
General: 2018/11/19 1
\textbf and related: Improved font detection. 541
lwarp.css: Added css for xfrac, nicefrac. 245
lwarp.css: Fixed css for \textup. 245
lwarp.css: Reduced margins in titlepage. 245
lwarp_formal.css: Fix: Font for verse. 278
2in1: Added. 567
CJKutf8: Prevented unless xeCJK. 650
CJK: Prevented unless xeCJK. . . 650
asymptote: Improved alt tags. . . 596
bitpattern: Added. 610
calc: Fix: Required for print version. 217
chngpage: Fix: Loads lwarp-chngpage. 649
ctexpatch: Added patch. 565
flippdf: Added. 689
musicography: Added. 788
nicefrac: Improved font control and css, honors nice, ugly. . . . 793
notespages: Added. 796
octave: Added. 808
pdfcomment: Added. 813
pdfmarginpar: Added. 814

register: Updated to v1.8. 830
rviewport: Added. 835
semantic-markup: Added. 845
textcomp: Fix conflict with xunicode. 875
tram: Added. 908
twoup: Added. 911
ulem: Improved compatibility with CJKulem. 912
ulem: Now works in a lateximage. 912
unitsdef: Added. 916
units: Improved font control and css, honors loose, tight. 915
xechangebar: Added. 936
xfrac: Improved css. 938
xunicode: Fix conflict with textcomp. 944
Added early checks for CJK, CJKutf8. 194
Docs: asymptote. 155
Docs: miktex-poppler-bin-* 89
Docs: *MiKTeX Console* 85
Docs: Improved MiKTEX install instructions. 85
Docs: UTF-8 locale. 169
File: lwarp_mathjax.txt: Removed inoperable siunitx extension. . . 284
Fix for \em. 544
Fix: Horiz white space. 933
Logos: CSS instead of <sup>, <sub>. 560
Logos: Fix for XƎTEX logo if graphics is not loaded. 562
Logos: Improved CSS. 560
Logos: Made robust. 560
fcolorminipage: Fix: Horiz white space. 934
Fix: No longer requires xifthen. . 526
fminipage: Fix: Horiz white space. 540, 541
LWR@blocktextcurrentfont: Added print version. 551
v0.63
\LWR@HTMLLatexCmd: ujarticle and related: Compile options. 244
\LWR@LwarpStart: Fixes for xeCJK. 364
\LWR@atbeginverbatim: Fix for xeCJK. 381
\LWR@checkloadbefore: Added. . . . 193
\LWR@checkloadfilename: Added to reduce number of lwarp-* files. 220
\LWR@checkloadnever: Added. 193
\LWR@compileuplatex: Added. 241
\LWR@createautosec: Fix for xeCJK. 351
\LWR@earlyclassloadnever: Added. 194
\LWR@filestart: Fix: Break ligature for luatexko. 363
\LWR@firstoffour: Added. 205
\LWR@htmlclosecomment: Fix: Break ligature for luatexko. 319
\LWR@isolate: Fix for xeCJK. 205
\LWR@notltjloadafter: Added more classes. 192
Added. 192
\LWR@subhtmlelementclass: Fix for xeCJK. 320
\LinkHome: Fix: Print version. . . . 312
\linkhomename: Added. 312
General: 2018/12/03 1
lwarp.css: Added css for vertical writing. 245
lwarp.css: Improved css for mdframed. 245
amsthm, mdframed: Fix for enforced load order. 587
emumitem: v3.6: Nullifiy \DrawEnumitemLabel. 665
everyhook: Fix for bxjs* classes. . 217
geometry: Fix for bxjs* classes. . 216
mdframed: Avoid thin rules. . . . 769
mdframed: Improved font control. 772–774
stfloats: Adapted to ltj* classes. . 861
xpinyin: Added. 942
zhlineskip: Added. 946
Added pTEXsupport. 191
Docs: \linkhomename. 111
Docs: \sidetocname. 113
Fix: Default \LWR@mdfive. 203
Improved titles. 772, 773
pTEX: Encoding. 203
pTEX: Load upquote. 204
pTEX: No newunicodechar. 204
LWR@BlockClassWP: Fix for xeCJK. . 323
v0.64
\LWR@HTMLLatexCmd: utarticle and related: Added. 244
\LWR@checkloadfilename: Prevented bitfield, doublespace, newthm, rplain, si. 220
\LWR@section: Support for ujarticle and related. 355
\enskip: Made robust. 554
\qquad: Made robust. 554
\quad: Made robust. 554
\theHTMLTitleSeparator: Added utarticle and related. 361
General: 2018/12/08 1
addlines: Updated to v0.3. 575
biblatex: Added patch for CTEX. . 607

bsheaders: Added. 616
gmeometric: Added. 713
marginal: Added. 765
rmpage: Added. 833
scrlayer-scrpage: Fixes. 842
scrlayer: Fixes. 841
scrpage2: Added. 842
ujarticle and related: Improved \today. 564
Added utarticle and related. . . . 564
v0.65
\LWR@LwarpEnd: Improved CSS for page layout. 368
\LWR@LwarpStart: Improved CSS for page layout. 366
\LWR@PreloadedPackage: \AtBeginDocument to avoid option clashes. 519
\LWR@hyperindexrefnullified: Added \textsi. 478
\LWR@includegraphicsb: graphics: Added \includegraphics alt key. 719, 721
\LWR@newhtmlfile: Error if duplicate file name. 347
Improved CSS for page layout. 346, 349
\LWR@nullfonts: Added \sishape. . 545
Added \textsi. 545
\LWR@restoreorigformatting: Added \sishape. 481
Fix: tabular*. 481
\enskip: Changed to Unicode EN SPACE. 554
\quad: Changed to Unicode EM SPACE. 554
\sishape: Added \sishape. 549
\textsi: Added. 543
General: 2018/12/22 1
lwarp.css: Added \sishape, \textsi. 245
lwarp.css: Improved CSS for page layout. 245
lwarp.css: Improved CSS for quotations. 245
lwarp.css: SideTOCto left for improved \marginpars. 245
lwarp_formal.css: SideTOCto left for improved \marginpars. . . . 278
lwarp_sagebrush.css: SideTOCto left for improved \marginpars. . 273
bounddvi: Added. 613
embrac: Added. 662
footnoterange: Added. 704
gentombow: Added. 709
geometry: Fix for bxjs* classes. . 216
graphics: Added \includegraphics alt key. 523, 714, 716, 717
lltjext: Added. 757
multicolrule: Added. 784
multicol: Added \docolaction. . 784
plarydshln: Added. 820
plextarydshln: Added. 821
plextcolortbl: Added. 821
plext: Added. 820
pxatbegshi: Added. 825
pxeveryshi: Added. 825
pxftnright: Added. 826
pxjahyper: Added. 826
tascmac: Added. 871
versonotes: Added. 919
Added \sishape. 235
Added early checks for jarticle, tarticle, and related. 194
Fix for \rensuji. 564
Fix space between class and id. . 324
tabular: Added support for plext. . 446
Fix: tabular*. 446
Fix: Rule color. 447
minipage: Refactored to later allow Japanese <t/y> argument. 533
LWR@figcaption: Uses <figurecaption> instead of <figcaption>. 466
v0.66
\@mpfootnotetext: Improved HTML formatting. 337
\IgnoreMinipageWidths: Added, . . 533
\LWR@LwarpStart: Fix: TOC, LOF, LOT links. 367
\LWR@checkloadfilename: Prevented colortab, epsf, hyper, picinpar, picins, sistyle, ucs. 220
\LWR@closeparagraph: Fix: Combined span, tabular, and lateximage. 328
Improved HTML formatting. . . . 327
\LWR@closeparagraph@br: Factored. 327
\LWR@fboxstyle: Use current text color. 538
\LWR@filenamenoblanks: Fix: Section names detokenized. 342
Fix: Section names with macros. 342
Fix: Section names with percent. 343
Improved file name generation. . 341
Limits filename length. 344
\LWR@findcurrenttextcolor: Fix: Color if xcolor not loaded. . . . 551
\LWR@footnotetext: Improved HTML formatting. 336

\LWR@htmlfileref: No longer use zref. 453
\LWR@htmlsectionfilename: Sanitize underscores. 310
\LWR@includegraphicsb: Improved HTML formatting. 718
\LWR@indentHTML: Added. 317
\LWR@lateximagedepthref: No longer use zref. 453
\LWR@lateximagenumberref: No longer use zref. 453
\LWR@lwarplabel: No longer use zref. 453
\LWR@nameref: No longer use zref. . 453
\LWR@nullfonts: Logos. 547
\LWR@openparagraph: Improved HTML formatting. 326
\LWR@section: Fix: TOC, LOF, LOT links. 357
Improved HTML formatting. 357
\LWR@setexparray: Fix with \par. . 306
\LWR@setref: No longer use zref. . . 452
\LWR@simplifyname: Added. 341
\LWR@startref: No longer use zref. 455
\LWR@stoppars: Improved HTML formatting. 330
\LWR@subhtmlelementclass: Improved HTML formatting. . . 320
\LWR@subhyperrefclass: Improved HTML formatting. 458
\LWR@subinlineimage: Improved HTML formatting. 460
\LWR@writeconf: Added ImagesDirectory and ImagesName. 244
\LinkHome: Fix: Document cross-references. 312
\UseMinipageWidths: Added, ... 533
\fbox: Fix: Removed extra space. . . 538
\hyperindexref: Fix: Long index entries. 478
\minipagefullwidth: Made \global. 532
\rotatebox: Improved HTML formatting. 723
\rule: Improved HTML formatting. . 558
\scalebox: Improved HTML formatting. 724
\textgreater: Made robust. 309
\textless: Made robust. 309
General: 2019/02/08 1
LWR@currentautosec: Fix for LOF, LOTfloat in home page. 353
lwarp.css: Improved CSS for definition lists. 245
lwarp_formal.css: Improved CSS for table notes. 278
lwarp_one_limage.txt: Image directory and prefix. 284
acronym: Fix for acronym in caption. 573
acronym: No longer uses zref. . . 573
ar: Added. 592
caption: Fix for options clash. . . 224
ed: Added. 661
extramarks: Updated to v3.10. . . 672
fancyhdr: Updated to v3.10. . . . 678
kotexutf: Patch for references. . . 566
memoir: Docs re: version numbers. 161
multicolrule: Updated for v1.2. . 784
nameauth: Added. 789
register: Verified for v1.9. 830
subcaption: Added. 862
tocbasic: Updated to v3.26a. . . . 896
truncate: Added. 910
zref: No longer used. 218
lwarpmk: Added ImagesDirectory and ImagesName. 286
lwarpmk: Fix for **cleanlimages**. . 286
Added early checks for colortab, epsf, hyper, picinpar, picins, sistyle, ucs. 194
Added option ImagesDirectory. 210
Added option ImagesName. 210
Added support for indentfirst. . . 331
Docs: Updated Converting an existing document. 104
Fix: Minipages inside multicols. 783
Improved HTML formatting. . 675, 681
Package dates added where possible. 567
Sanitize filenames. 213
tabular: Fix: Minipages inside tabular. 449
lateximage: Adds \BaseJobname for multiple projects. 513
Improved HTML formatting. 514
minipage: Honor \LWR@forceminipagefullwidth. 534
v0.67
\FilenameNullify: Added. . . . 548, 551
\FilenameSimplify: Added. . . 341, 345
\LWR@doequation: xfakebold: Added support. 500
\LWR@doubledollar: xfakebold: Added support. 494
\LWR@filenamenoblanks: Improved file name generation. 341

\LWR@lookforpackagename:
easyReview: Supported. 221
\LWR@nullfonts: Add'l symbols. . . 545
\LWR@simplifycustom: Added. 341
\LWR@subsingledollar: xfakebold:
Added support. 488, 492, 493
General: 2019/02/23 1
academicons: Added. 570
bbding: Added. 602
changes: Added. 622
color: Fix for version number. . . 651
dingbat: Added. 658
eurosym: Added. 671
fitbox: Added. 687
fontawesome5: Added. 701
fontawesome: Added. 700
foreign: Added. 705
gloss: Added. 710
karnaugh-map: Added. 743
marvosym: Added. 767
multicap: Added. 782
nomencl: Added. 794
notes: Added. 795
pifont: Added. 819
struktex: Added. 862
textcomp: Nullify in filenames. . 876
typicons: Added. 912
umoline: Added. 914
xfakebold: Added support. . . . 486
xfakebold: Added. 937
xunicode: Nullify in filenames. . 945
AMS environments: Added
xfakebold support. 581
eqnarray: xfakebold: Added support. 506

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.

Symbols

\$ *487*
\& *309*, 7012
\(. 9439
*-images.txt (file) *513*
*_html.aux (file) *344*, *450*, *452*, *513*
*_html.lof (file) *465*
*_html.lot (file) *465*
*_html.tex (file) *239*
\, ... *123*
--shell-escape (option) *105*
\@@@setcpageref 10220
\@@@setcref 10183
\@@@setcrefrange 10197
\@author *369*
\@begintheorem 6678
\@biblabel 9097
\@captype 8715
\@chapcntformat 5884
\@currentHref 8550
\@currentlabelname 8409
\@date *369*
\@dlbfloat 8681
\@donoparitem 6704
\@endtheorem 6690
\@ensuredmath 9449
\@float 8681
\@fnsymbol 6467
\@footnotetext 5473
\@include 1039
\@item 6717
\@makecaption 8736
\@makefnmark 5441
\@makefntext 5440
\@maketitle 56, 6490
\@mklab 6698
\@mpfootnotetext 5475
\@nbitem 6784
\@opargbegintheorem 6684
\@partcntformat 5885
\@partnameformat 5886
\@rowc@lors 7333
\@rowcolors 7332
\@seccntformat 5882
\@starttoc 8849
\@textsubscript 10867
\@textsuperscript 10863
\@title *369*
\@wrglossary 9041
\@wrindex 9030
\[. 9439
\\ *553*
$ *493*
$$ *493*
2in1 (package) *567*
2up (package) *567*

A

a4 (package) *567*
a4wide (package) *568*
a5comb (package) *568*
abstract (environment) 6553
abstract (package) *132*, *568*
\abstractname *114*, 6552
academicons (package) *570*
accsupp (package) *571*
acro (package) *571*
acronym (package) *573*
\addcontentsline 8800
addlines (package) *575*
\AddSubtitlePublished 6518
adjmulticol (package) *574*
Adobe (program) *81*
\affiliation 6370
afterpage (package) *575*
algorithm2e (package) *575*
algorithmicx (package) *579*
algorithmx (package) *159*
align (environment) 131
align* (environment) 158
alignat (environment) 239
alignat* (environment) 266
alltt (package) *580*

\AmS 11155
amsmath (package) 580
amsthm (package) 587
\and 369
anonchap (package) 590
anysize (package) 591
appendix (package) 133, 591
ar (package) 592
arabicfront (package) 593
array (package) 593
\arrayrulecolor 7340
\arrayrulecolornexttoken 7340
arydshln (package) 594
AsciiDoc (program) 81
AsciiDoctor (program) 81
Asciidoctor-LaTeX (program) 81
asymptote (package) 155, 596
atbegshi (package) 597
attachfile (package) 597
attachfile2 (package) 598
\attrib 162, 379, 917
\attribution 6563
authblk (package) 132, 600
\author 120, 369
autosec 351
axessibility (package) 601
axodraw2 (package) 601

B

babel (package) 163
\backmatter 5860
backref (package) 602
balance (package) 602
\BaseJobname 4782
BaseJobname (option) 110, 209
bbding (package) 602
\bfseries 10846
biblatex (package) 607
\bibliography 9088
\BibTeX 11150
bibunits (package) 608
bigdelim (package) 159, 608
bigstrut (package) 609
bitpattern (package) 610
BlockClass (environment) 5151
\BlockClassSingle 5165
blowup (package) 610
booklet (package) 610
bookmark (package) 611
booktabs (package) 611
boolean:
    CombineHigherDepths 113, 340
    FileSectionNames 114, 310
    FormatEPUB 170, 232
    FormatWP 172, 232
    HTMLDebugComments 114, 227
    LWR@amsmultline 503
    LWR@copiedsidetoc 470
    LWR@doingapar 325
    LWR@doingcmidrule 395
    LWR@doingstartpars 325
    LWR@doingtbrule 395
    LWR@dynamicmath 308
    LWR@emptyatbang 395
    LWR@exitingtabular 441
    LWR@forceminipagefullwidth 532
    LWR@freezethisautoid 464
    LWR@indisplaymathimage 486
    LWR@infloatrow 717
    LWR@intabularmetadata 395
    LWR@isstartingequation 511
    LWR@mathmacro 307
    LWR@minipagefullwidth 532
    LWR@minipagethispar 533
    LWR@opttablecol 408
    LWR@origmathjax 208
    LWR@skipatbang 395
    LWR@skippingmcolrowcell 395
    LWR@skippingmrowcell 395
    LWR@starredlongtable 434
    LWR@startedrow 394
    LWR@tableparcell 395
    LWR@tabularcelladded 394
    LWR@tabularmutemods 441
    LWR@tracinglwarp 226
    LWR@validtablecol 408
    LWR@verbtags 380
    LWR@xfakebold 486
    mathjax 208
    usingOSWindows 207
    warpingHTML 208
    warpingprint 208
    WPMarkFloats 173, 232
    WPMarkLOFT 174, 233
    WPMarkMath 174, 233
    WPMarkMinipages 174, 232
    WPMarkTOC 174, 232
    WPTitleHeading 175, 233
bophook (package) 613
bounddvi (package) 613
boxedminipage2e (package) 613
\boxframe 190
breakurl (package) 614
breqn (package) 614
bsheaders (package) 616
BVerbatim (environment) 222
bxpapersize (package) 616
bytefield (package) 616

C

calc (package) 217
cancel (package) 617

canoniclayout (package) 618
capt-of (package) 468
caption (package) 160, 224, 468, 618
\caption@begin 8781
\caption@end 8781
\captionlistentry 8785
\captionof 8816
\CaptionSeparator 8735
cases (package) 621
center (environment) 9975
\centering 9999
\centerline 10018
changebar (package) 621
changepage (package) 622
changes (package) 622
chappg (package) 623
\chapter 6086
chapterbib (package) 624
chemfig (package) 624
chemformula (package) 625
chemgreek (package) 630
chemmacros (package) 630
chemnum (package) 648
chkfloat (package) 649
chngpage (package) 649
cite (package) 649
CJK (package) 650
CJKutf8 (package) 650
class (key) [Gin] 717
class:
  internet 80
  komascript 161
  memoir 161, 162, 918
\cleardoublepage 11027
\clearpage 11027
cleveref (package) 130, 527
clrdblpg (package) 650
cm-super (package) 105
cmap (package) 106
cmdtrack (package) 651
color (package) 153, 651
\colorbox 73
\colorboxBlock 86, 10080
colortbl (package) 651
CombineHigherDepths (boolean) . . 113, 340
comment (package) 214
\ConTeXt 11147
continue (package) 653
copyrightbox (package) 653
counter:
  FileDepth 113, 340
  FootnoteDepth 114, 334
  lofdepth 474
  lotdepth 474
  LWR@cellcolordepth 427
  LWR@currentautosec 352
  LWR@externalfilecnt 486
  LWR@hdashedlines 395
  LWR@hlines 394
  LWR@htmlfilenumber 310
  LWR@latestautopage 466
  LWR@lateximagedepth 508
  LWR@lateximagenumber 508
  LWR@LIpage 508
  LWR@midrulecounter 418
  LWR@minipagedepth 532
  LWR@nextautoid 465
  LWR@nextautopage 465
  LWR@nextequation 497
  LWR@prevFileDepth 352
  LWR@previousautopagelabel 344
  LWR@spandepth 325
  LWR@startingequation 511
  LWR@tablecolspecindex 396
  LWR@tablecolspecwidth 396
  LWR@tableLaTeXcolindex 396
  LWR@tabletotalLaTeXcols 396
  LWR@tabletotalLaTeXcolsnext . . . 397
  LWR@tabularDepth 395
  LWR@tabularpardepth 395
  LWR@thisautoid 463
  LWR@thisautoidWP 464
  SideTOCDepth 113, 472
  tocdepth 113
\cpagerefFor 10219
crop (package) 654
\CSSFilename 114, 117, 5396
\CustomizeMathJax 5708, 5733
cuted (package) 654
cutwin (package) 654

D

\date 120
dblfloatfix (package) 655
dblfnote (package) 655
dcolumn (package) 656
\DeclareGraphicsExtensions 2
dejavu (package) 105
description (environment) 6859
diagbox (package) 656
dingbat (package) 658
\displaymathnormal . . 150, 485, 502, 9632
\displaymathother . . . 150, 485, 502, 9643
ditaa (package) 169
\dotfill 10946
\doublerulesepcolor 7342
\doublerulesepcolornexttoken 7342
dprogress (package) 659
draftcopy (package) 659
draftfigure (package) 659
draftwatermark (package) 660

dvipdfm (option) *107*, *212*
dvipdfmx (option) *107*, *212*
dvips (option) *107*, *212*

E

easy-todo (package) *660*
ebook (package) *661*
ed (package) *661*
ellipsis (package) *662*
embrac (package) *662*
\emph 10545
emptypage (package) *663*
\end@dlbfloat 8698
\end@float 8698
endfloat (package) *663*
endheads (package) *664*
endnotes (package) *133*, *665*
\enlargethispage 11026
\enskip *554*, 10972
enumerate (environment) 6840
enumerate (package) *665*
enumitem (package) *665*
environ (package) *218*
environment:
  fminipage *126*
  lateximage *507*
  picture *530*
  titlepage *120*
  titlingpage *120*
  warpall *119*
  warpHTML *116*, *118*
  warpprint *116*, *119*
environments:
  abstract 6553
  align 131
  align* 158
  alignat 239
  alignat* 266
  BlockClass 5151
  BVerbatim 222
  center 9975
  description 6859
  enumerate 6840
  eqnarray 9692
  equation 9610
  equation* 9618
  fcolorminipage 166, 10121
  flalign 185
  flalign* 212
  flushleft 9991
  flushright 9983
  fminipage 10484
  gather 77
  gather* 104
  itemize 6834
  lateximage 9842, 9970
  list 6804
  longtable 2
  LWR@BlockClassWP 5192
  LWR@blocktextcurrentfont 10838, 10895
  LWR@displaymathnormal 9479
  LWR@displaymathother 9484
  LWR@equationother 9498
  LWR@figcaption 8750
  LWR@nestspan 5004
  math 9478
  minipage 10304
  multline 20
  multline* 48
  picture 10279
  quote 6579
  tabbing 6663
  tabular 8283
  thebibliography 9098
  theindex 9006
  titlepage 6387
  titlingpage 14
  verbatim 6653
  verse 2, 6594
  warpall 787
  warpHTML 788
  warpprint 788
epigraph (package) *666*
epsfig (package) *667*
epstopdf (package) *154*, *668*
epstopdf (program) *152*, *522*
epstopdf-base (package) *668*
eqnarray (environment) 9692
equation (environment) 9610
equation* (environment) 9618
errata (package) *669*
eso-pic (package) *670*
\etalchar *133*
etoolbox (package) *191*
eurosym (package) *671*
everyhook (package) *217*
everypage (package) *671*
everyshi (package) *672*
expl3 (package) *217*
extramarks (package) *672*

F

fancybox (package) *127*, *673*
fancyhdr (package) *678*
fancyref (package) *679*
fancytabs (package) *680*
fancyvrb (package) *680*
\fbox *126*, 10459
\fboxBlock *126*, 10467
\fcolorbox 103, 10082
\fcolorboxBlock 126, 10119
fcolorminipage (environment) . 166, 10121

figcaps (package) 686
figsize (package) 686
file:
*-images.txt 513
*_html.aux 344, 450, 452, 513
*_html.lof 465
*_html.lot 465
*_html.tex 239
glyphtounicode.tex 106
lwarp.css 118, 245
lwarp.ist 144, 282
lwarp.xdy 145, 283
lwarp_baseline_marker.eps 487
lwarp_baseline_marker.png 487
lwarp_formal.css 278
lwarp_mathjax.txt 284
lwarp_one_limage.cmd 283
lwarp_sagebrush.css 273
lwarp_tutorial.txt 90
lwarpmk.conf 245
lwarpmk.lua 180
project.css 118
project.lwarpmkconf 245
sample_project.css 118, 282
tutorial.tex 90
filecontents (package) 217
FileDepth (counter) 113, 340
\FilenameLimit 114, 5574
\FilenameNullify 129, 10811, 10896
\FilenameSimplify 129, 5595, 5734
FileSectionNames (boolean) 114, 310
fitbox (package) 687
fix2col (package) 687
fixme (package) 164, 687
fixmetodonotes (package) 689
flafter (package) 689
\flagverse 918
flalign (environment) 185
flalign* (environment) 212
Flare (program) 81
flippdf (package) 689
float (package) 159, 690
floatflt (package) 691
floatpag (package) 692
floatrow (package) 160, 692
fltrace (package) 697
\flushbottom 4920
flushend (package) 698
flushleft (environment) 9991
flushright (environment) 9983
fminipage (environment) 126, 10484
fnbreak (package) 698
fncychap (package) 698
fnlineno (package) 699
fnpos (package) 699
fontawesome (package) 700
fontawesome5 (package) 701
fontenc (package) 106, 702
fontspec (package) 106
footmisc (package) 702
footnote (package) 703
FootnoteDepth (counter) 114, 334
footnotehyper (package) 704
footnoterange (package) 704
footnpag (package) 705
\ForceHTMLPage 132, 5835
\ForceHTMLTOC 132, 5841
foreign (package) 705
forest (package) 705
FormatEPUB (boolean) 170, 232
FormatWP (boolean) 172, 232
\framebox 10435
framed (package) 706
FrameMaker (program) 81
\frontmatter 5857
ftnright (package) 708
fullminipage (package) 708
fullpage (package) 708
fullwidth (package) 709
\fup 10871
\fussy 4922
fwlw (package) 709

G

gather (environment) 77
gather* (environment) 104
GELLMU (program) 80
gentombow (package) 709
geometry (package) 216, 710
gettitlestring (package) 217
[Gin]:
class (key) 717
GladTeX (program) 80
gloss (package) 134, 710
glossaries (package) 134, 711
GlossaryCmd (option) . . . 110, 134, 212, 711
glyphtounicode.tex (file) 106
gmeometric (package) 713
graphics (package) 151, 713
graphicx (package) 151, 725
grffile (package) 153, 726
grid (package) 726
grid-system (package) 726
gridset (package) 727

H

hang (package) 727
hanging (package) 728
Hevea (program) 80
\hfill 10942
\HomeHTMLFilename 4784

HomeHTMLFilename (option) . 107, 111, 211
\href 8578
\hrulefill 10944
\hskip 123
\hspace 123, 554, 10980, 11010, 11016
htlatex (program) 80
\HTMLAuthor 115, 121, 5422
HTMLDebugComments (boolean) 114, 227
\HTMLDescription 115, 121, 5427
\HTMLentity 4766
\HTMLFilename 4783
HTMLFilename (option) 107, 111, 211
\HTMLFirstPageTop 115, 5384
htmlglossary (option) [lwarpmk] . 134, 711
HTMLIndexCmd (option) 109, 211
\HTMLLanguage 114, 6156
HTMLLatexCmd (option) 107, 166, 211
\HTMLleftmargini (length) 162, 379, 380, 918
\HTMLPageBottom 115, 370, 5390
\HTMLPageTop 115, 5387
\HTMLTitle 114, 121, 5417
\HTMLTitleAfterSection 114, 6179
\HTMLTitleBeforeSection 114, 6174
\HTMLunicode 4774
\HTMLvleftskip (length) 162, 379, 380, 918
hypcap (package) 729
hypdestopt (package) 730
\hyperindexref 9066
hypernat (package) 730
hyperref (package) 130, 457, 730
\hypertoc 8945
\hypertocfloat 8967
hyperxmp (package) 735
hyphenat (package) 735

I

idxlayout (package) 736
\if@titlepage 6365
ifoddpage (package) 737
ifplatform (package) 191
\IgnoreMinipageWidths . . . 125, 531, 10294
ImagesDirectory (option) 210
ImagesName (option) 210
imakeidx (package) 737
\include 1039
\includegraphics 284
indentfirst (package) 331
InDesign (program) 81
index (package) 742
\InlineClass 5176
\inlinemathnormal 150, 485, 4759
\inlinemathother 150, 485, 4755
inputenc (package) 106
inputenx (package) 106
internet (class) 80
intopdf (package) 743
itemize (environment) 6834
\itshape 10851

K

karnaugh-map (package) 743
\kern 123
key:
  [Gin]:
    class 717
keyfloat (package) 161, 746
komascript (class) 161
kotex (package) 165
kvoptions (package) 208

L

\l@chapter 8986
\l@figure 8997
\l@paragraph 8995
\l@part 8985
\l@section 8992
\l@subparagraph 8996
\l@subsection 8993
\l@subsubsection 8994
\l@table 8998
\LaTeX 11107
LaTeX2HTML (program) 80
\LaTeXe 11107
lateximage (environment) . 507, 9842, 9970
\LateximageFontScale 9764
\LateximageFontSizeName 9763
latexmk (option) 107, 111, 212
latexmk (program) 167
LatexmkIndexCmd (option) 109, 211
LaTeXML (program) 80
layaureo (package) 748
layout (package) 749
leading (package) 749
\leftline 10017
length:
  \HTMLleftmargini . . 162, 379, 380, 918
  \HTMLvleftskip 162, 379, 380, 918
  \LWR@cmidrulewidth 419
  \LWR@heavyrulewidth 419
  \LWR@lightrulewidth 419
  \LWR@minipageheight 532
  \LWR@minipagewidth 532
  \LWR@tempheight 554
  \LWR@tempraise 554
  \LWR@tempwidth 554
  \LWR@thiscmidrulewidth 419
  \VerbatimHTMLWidth 380
  \vleftmargini 162, 379, 918
  \vleftskip 162, 379, 918
letltxmacro (package) 191
letterspace (package) 749
lettrine (package) 750

LibreOffice (program) 81
\linebreak 11020
lineno (package) 751
\LinkHome 4842, 4847
\linkhomename 111, 4839
Linux (program) 118, 206
lips (package) 753
list (environment) 6804
listings (package) 753
\listof 8903
\listoffigures 8877
\listoftables 8890
lltjext (package) 757
lmodern (package) 105, 106
lofdepth (counter) 474
longtable (environment) 2
longtable (package) 158, 758
lotdepth (counter) 474
lscape (package) 760
ltablex (package) 760
ltcaption (package) 761
ltxgrid (package) 761
ltxtable (package) 761
lua-check-hyphen (package) 762
lua-visual-debug (package) 762
luacolor (package) 762
\LuaLaTeX 11136
LuaLaTeX (program) [requirement] 84
\LuaTeX 11136
luatodonotes (package) 164, 762
lwarp (package) 107
lwarp-patch-komascript (package) . . 948
lwarp-patch-memoir (package) 950
lwarp.css (file) 118, 245
lwarp.ist (file) 144, 282
lwarp.xdy (file) 145, 283
lwarp_baseline_marker.eps (file) 487
lwarp_baseline_marker.png (file) 487
lwarp_formal.css (file) 278
lwarp_mathjax.txt (file) 284
lwarp_one_limage.cmd (file) 283
lwarp_sagebrush.css (file) 273
lwarp_tutorial.txt (file) 90
[lwarpmk]:
  htmlglossary (option) 134, 711
  printglossary (option) 134, 711
lwarpmk (option) 110, 210, 286
lwarpmk (program) 180, 286
lwarpmk epstopdf (program) 152, 522
lwarpmk pdftosvg (program) 152, 522
lwarpmk.conf (file) 245
lwarpmk.lua (file) 180
\lwarpsetup 651
\LWR@addbaselinemarker 9195
\LWR@addcdashline 7611
\LWR@addcmidruletrim 7577
\LWR@addcmidrulewidth 7608
\LWR@addcompilecmd 1418
\LWR@addformatwpalignment 7627
\LWR@addleftmostbartag 7412
\LWR@addmathjax 9547
\LWR@addmulticolvertrulecolor . . . 7886
\LWR@addrulewidth 7582
\LWR@addtabularcellcolor 7755
\LWR@addtabularhrulecolor 7655
\LWR@addtabularrowcolor 7640
\LWR@addtabularrulecolors 7701
\LWR@afterendverbatim 6636
\LWR@amsmathbody 9829
\LWR@amsmathbodynumbered 9835
LWR@amsmultline (boolean) 503
\LWR@applyxfakebold 9187
\LWR@atbeginverbatim 6617
\LWR@backgroundcolor 36
\LWR@beginhideamsmath 9677
LWR@BlockClassWP (environment) . . . 5192
LWR@blocktextcurrentfont (environment) 10838, 10895
\LWR@botnavigation 4865
LWR@cdashlines 418
LWR@cellcolordepth (counter) 427
\LWR@cellHTMLcolor 7337
\LWR@checkloadbefore 59
\LWR@checkloadfilename 895
\LWR@checkloadnever 73
\LWR@clearmidrules 7492
\LWR@closeparagraph 5290
\LWR@closeparagraph@br 5280
\LWR@closeprevious 4892
\LWR@closetabledatacell 6971
\LWR@cmidrulewidth (length) 419
LWR@colafterspec 397
LWR@colatspec 397
LWR@colbangspec 397
LWR@colbarspec 397
LWR@colbeforespec 397
\LWR@columnHTMLcolor 7335
\LWR@columnspeclookahead 7090
\LWR@compilecmd 1415
\LWR@compileuplatex 1447
\LWR@convertto 590
LWR@copiedsidetoc (boolean) 470
\LWR@copyfile 1074
\LWR@createautosec 5866
LWR@currentautosec (counter) 352
\LWR@currentcss 5395
\LWR@currenttextcolor 10882
\LWR@customizedMathJax 5707
\LWR@customizeMathJax 5713
\LWR@descitem 6846

LWR@displaymathnormal (environment) 9479
LWR@displaymathother (environment) 9484
\LWR@docdashline 7541
\LWR@docmidrule 7525
\LWR@doequation 9560
\LWR@doindexentry 9049
LWR@doingapar (boolean) 325
LWR@doingcmidrule (boolean) 395
LWR@doingstartpars (boolean) 325
LWR@doingtbrule (boolean) 395
\LWR@domulticolumn 7924
\LWR@doubledollar 9393
LWR@dynamicmath (boolean) 308
\LWR@earlyclassloadnever 89
\LWR@earlyloadnever 79
LWR@emptyatbang (boolean) 395
\LWR@endfloatalignment 8731
\LWR@endhideamsmath 9685
\LWR@endofline 10953
\LWR@ensuredoingapar 5256
LWR@equationother (environment) . . 9498
\LWR@equationtag 9815
LWR@exitingtabular (boolean) 441
\LWR@expandableformatted 1136
\LWR@expandableformattedenv 1161
LWR@externalfilecnt (counter) 486
\LWR@fboxstyle 10453
LWR@figcaption (environment) 8750
\LWR@filenamenoblanks 5606
\LWR@filestart 6188
\LWR@findcurrenttextcolor 10885
\LWR@findword 892
\LWR@firstoffour 605
\LWR@floatalignment 8717
\LWR@floatalignmentname 8716
\LWR@floatbegin 8649
\LWR@floatend 8683
\LWR@floatstyle 2
\LWR@footnotetext 5444
LWR@forceminipagefullwidth (boolean) 532
\LWR@forceminwidth 10442
\LWR@forcenewpage 4909
\LWR@formatted 1121
\LWR@formattedenv 1151
\LWR@formatting 1120
\LWR@fourthoffour 605
LWR@freezethisautoid (boolean) 464
\LWR@futurenonspacelet 6918
\LWR@FVstyle 56
\LWR@getexparray 4728
\LWR@getmynexttoken 6925
LWR@hdashedlines (counter) 395
\LWR@heavyrulewidth (length) 419
\LWR@hidelatexequation 9533
LWR@hlines (counter) 394
\LWR@HTML@caption@begin 8760
\LWR@HTML@caption@end 8772
\LWR@HTML@Gin@setfile 115
\LWR@HTML@ref 8521
\LWR@htmlblockcomment 5090
\LWR@htmlblocktag 5092
\LWR@HTMLcline 8268
\LWR@htmlclosecomment 5062
\LWR@htmlcomment 5083
\LWR@htmldivclass 5128
\LWR@htmldivclassend 5131
\LWR@htmlelement 5141
\LWR@htmlelementclass 5115
\LWR@htmlelementclassend 5120
\LWR@htmlelementclassline 5134
\LWR@htmlelementend 5144
LWR@htmlfilenumber (counter) 310
\LWR@htmlfileref 8428
\LWR@HTMLhline 8251
\LWR@HTMLLatexCmd 1457
\LWR@htmlmathlabel 9662
\LWR@htmlmulticolumn 7973
\LWR@htmlopencomment 5062
\LWR@htmlrefsectionfilename 4829
\LWR@HTMLsanitize 9767
\LWR@HTMLsanitizeexpand 9784
\LWR@htmlsectionfilename 4794
\LWR@htmlspan 5039
\LWR@htmlspanclass 5047
\LWR@htmltag 5057
\LWR@htmltagc 4994
\LWR@hyperindexrefnullified 9054
\LWR@imagesizebox 114
\LWR@includegraphicsb 133
\LWR@indentHTML 4991
\LWR@indexitem 9018
\LWR@indexnameref 9048
\LWR@indexsubitem 9022
\LWR@indexsubsubitem 9026
LWR@indisplaymathimage (boolean) . . 486
LWR@infloatrow (boolean) 717
\LWR@instertatbangcols 6963
LWR@intabularmetadata (boolean) . . . 395
\LWR@isolate 596
LWR@isstartingequation (boolean) . . 511
\LWR@itemizeitem 6827
LWR@latestautopage (counter) 466
LWR@lateximagedepth (counter) 508
\LWR@lateximagedepthref 8431
LWR@lateximagenumber (counter) 508
\LWR@lateximagenumberref 8434
\LWR@lateximagesfile 1036
\LWR@latexmkcmd 1433
\LWR@latexmkdvipdfm 1438
\LWR@latexmkvar 1427

\LWR@lightrulewidth (length) *419*
LWR@LIpage (counter) *508*
\LWR@listitem 6789
\LWR@loadafter 23
\LWR@loadbefore 50
\LWR@loadnever 65
\LWR@longtabledatacaptiontag 7984
\LWR@lookforpackagename 933
\LWR@lwarpconfversion 1409
\LWR@LwarpEnd 6339, 11200
\LWR@lwarplabel 8437
\LWR@LwarpStart 6268, 11200
\LWR@maketitlesetup 33, 6459
LWR@mathmacro (boolean) *307*
\LWR@maybenewtablerow 7344
\LWR@maybeprintpendingfootnotes . 5518
LWR@midrulecounter (counter) *418*
LWR@midrules *418*
LWR@minipagedepth (counter) *532*
LWR@minipagefullwidth (boolean) ... *532*
\LWR@minipageheight (length) *532*
\LWR@minipagestartpars 10960
\LWR@minipagestoppars 10963
LWR@minipagethispar (boolean) *533*
\LWR@minipagewidth (length) *532*
\LWR@multicolother 7827
\LWR@multicolpartext 7821
\LWR@multicolskip 7833
\LWR@multirowborder 3
\LWR@mynexttoken 6917
\LWR@myshorttoc 8826
\LWR@nameref 8425
LWR@nestspan (environment) 5004
\LWR@new@label 8473
\LWR@newautoidanchor 8704
\LWR@newautopagelabel 5699
\LWR@newhtmlfile 5737
LWR@nextautoid (counter) *465*
LWR@nextautopage (counter) *465*
LWR@nextequation (counter) *497*
\LWR@notltjloadafter 37
\LWR@notmemoirloadafter 34
\LWR@nullfonts 10670
\LWR@nullifycomment 886
\LWR@nullifyNoAutoSpacing 8270
\LWR@nulllistfills 6796
\LWR@opacity 113
\LWR@openparagraph 5262
LWR@opttablecol (boolean) *408*
\LWR@orig@setBold 9185
\LWR@orig@unsetBold 9186
\LWR@origcolspec 6957
\LWR@originname 298
\LWR@originnames 305
LWR@origmathjax (boolean) *208*
\LWR@overline 10877
\LWR@parseaftercolumn 7160
\LWR@parseatcolumn 7100
\LWR@parsebangcolumn 7126
\LWR@parsebarcolumn 7171
\LWR@parsebeforecolumn 7149
\LWR@parsecoloncolumn 7196
\LWR@parseDcolumn 7242
\LWR@parsedrequirepackagenames ... 885
\LWR@parsenormalcolumn 7221
\LWR@parsepcolumn 7233
\LWR@parsesemicoloncolumn 7217
\LWR@parsetablecols 7248
\LWR@parsewcolumn 7237
\LWR@patcherror 591
\LWR@patchlists 6865
\LWR@pdfencoding 547
\LWR@PreloadedPackage 10022
LWR@prevFileDepth (counter) *352*
LWR@previousautopagelabel (counter) *344*
\LWR@printatbang 7387
\LWR@printbartag 7377
\LWR@printcloselist 6696
\LWR@PrintLatexCmd 1457
\LWR@printlength 867
\LWR@printmccoldata 7836
\LWR@printmccoltype 7779
\LWR@printopenlist 6697
\LWR@printpendingfootnotes 5508
\LWR@printpendingmpfootnotes 5527
\LWR@printthetitle 6411
\LWR@providelength 587
\LWR@ProvidesPackageDrop 1016
\LWR@ProvidesPackagePass 1003
\LWR@pushoneclose 5872
\LWR@quickfile 1033
\LWR@ref@ignorestar 8529
\LWR@remembertag 9819
\LWR@replaceEPSSVG 122
\LWR@requesttoc 6332
\LWR@requirepackagenames 884
\LWR@restoreorigaccents 1366
\LWR@restoreorigformatting 9117
\LWR@restoreoriglists 6893
\LWR@rotboxorigin 297
\LWR@rotstyle 105
\LWR@rowHTMLcolor 7336
\LWR@ruleHTMLcolor 7338
\LWR@sanitize 722
\LWR@sanitized 721
\LWR@scalestyle 108
\LWR@secondoffour 605
\LWR@section 5889
\LWR@sectionnumber 5863
\LWR@select@html@hspace 10980

\LWR@select@html@nohspace 11010
\LWR@select@print@hspace 11013
\LWR@setexparray 4719
\LWR@setlatestname 8411
\LWR@setOSWindows 639
\LWR@setref 8419
\LWR@shellescapecmd 1410
\LWR@sidetoc 8919
\LWR@simplifycustom 5594
\LWR@simplifyname 5583
\LWR@singledollar 9418
LWR@skipatbang (boolean) *395*
LWR@skippingmcolrowcell (boolean) . . *395*
LWR@skippingmrowcell (boolean) *395*
LWR@spandepth (counter) *325*
LWR@starredlongtable (boolean) *434*
LWR@startedrow (boolean) *394*
LWR@startingequation (counter) *511*
\LWR@startingequationtag 9814
\LWR@startnewdepth 5876
\LWR@startpars 5332
\LWR@startref 8486
\LWR@stoppars 5350
\LWR@stripperiod 8410
\LWR@strresult 6955
\LWR@subaddcmidruletrim 7564
\LWR@subaddtabularcellcolor 7749
\LWR@subcdashline 7528
\LWR@subcmidrule 7510
\LWR@subhtmlelementclass 5097
\LWR@subhyperref 8555
\LWR@subhyperrefclass 8568
\LWR@subhyperreftext 8563
\LWR@subinlineimage 8628
\LWR@sublabel 8450
\LWR@subnewref 8515
\LWR@subsingledollar 9231
\LWR@subtableofcontents 8839
\LWR@subtabularhtmlcolumns 8056
\LWR@syncmathjax 9513
LWR@tablecolspec *396*
LWR@tablecolspecindex (counter) . . . *396*
LWR@tablecolspecwidth (counter) . . . *396*
\LWR@tabledatacolumntag 8147
\LWR@tabledatasinglecolumntag . . . 7426
LWR@tableLaTeXcolindex (counter) . . . *396*
LWR@tableparcell (boolean) *395*
LWR@tabletotalLaTeXcols (counter) . . *396*
LWR@tabletotalLaTeXcolsnext (counter) *397*
LWR@tabularcelladded (boolean) *394*
LWR@tabularDepth (counter) *395*
\LWR@tabularendofline 7070
\LWR@tabularfinishrow 7024
\LWR@tabularhtmlcolumns 8066
\LWR@tabularleftedge 7417
LWR@tabularmutemods (boolean) *441*
LWR@tabularpardepth (counter) *395*
\LWR@tdaddstyle 7552
\LWR@tdendstyles 7558
\LWR@tdstartstyles 7551
\LWR@tempcolor 10883
\LWR@tempcolortwo 10883
\LWR@tempheight (length) *554*
\LWR@tempraise (length) *554*
\LWR@tempwidth (length) *554*
\LWR@textcurrentcolor 19, 10888
\LWR@textcurrentfont 10817, 10894
\LWR@thirdoffour 605
LWR@thisautoid (counter) *463*
LWR@thisautoidWP (counter) *464*
\LWR@thiscmidrulewidth (length) *419*
\LWR@thiscolspec 7425
\LWR@thisfilename 5577
\LWR@thisnewfilename 5582
\LWR@titlingmaketitle 83, 6510
\LWR@topnavigation 4862
\LWR@traceinfo 1102
LWR@tracinglwarp (boolean) *226*
LWR@trimlrules *418*
LWR@trimrrules *418*
\LWR@unknownengine 1422
LWR@validtablecol (boolean) *408*
LWR@verbtags (boolean) *380*
\LWR@vspace 11017
\LWR@WPcell 7623
\LWR@writeconf 1559
\LWR@xcolorrowHTMLcolor 7334
LWR@xfakebold (boolean) *486*
\LWRopquote 634
\LWRopseq 635
\LWRPrintStack 4870
\LWRsetnextfloat 8744
\LyX 11168

M

Mac OS (program) *118*, *206*
Madcap (program) *81*
magaz (package) *764*
\mainmatter 5853
make (program) *169*
\makebox 10413
makeidx (package) *146*, *765*, *897*
\MakeIndex 11150
makeindex (option) *107*, *111*, *211*
makeindex (program) *144*
makeindexStyle (option) . . . *107*, *144*, *210*
\makelabel *384*
\maketitle 47, *120*, 6471
marginal (package) *765*
marginfit (package) *765*
marginfix (package) *766*

marginnote (package) 766
\marginpar 124, 338, 5539
\marginparBlock 124, 338, 5550, 5567
\markboth 4917
\markright 4918
marvosym (package) 767
math (environment) 9478
\mathimagename 115, 9177
MathJax (program) 148, 149, 483, 484
MathJax (program) [requirement] 84
mathjax (boolean) 208
mathjax (option) 107, 111, 209
mathsvg (option) 107, 111, 209
\mbox 10410
mcaption (package) 767
\mcolrowcell 8222
mdframed (package) 128, 767
\mdseries 10845
memhfixc (package) 776
memoir (class) 161, 162, 918
metalogo (package) 777
mhchem (package) 777
microtype (package) 215, 779
midfloat (package) 780
midpage (package) 780
\MiKTeX 11167
minipage (environment) 10304
\minipagefullwidth 10292
morefloats (package) 781
moreverb (package) 781
mparhack (package) 782
\mrowcell 8219
MS-Windows (program) 118, 206
multicap (package) 782
multicol (package) 783
multicolrule (package) 784
\multicolumnrow 39, 8089
\multirow 785
multirow (package) 784
multitoc (package) 787
multline (environment) 20
multline* (environment) 48
musicography (package) 787

N

nameauth (package) 789
\Nameref 8547
\nameref 8538
nameref (package) 791
natbib (package) 791
nccfancyhdr (package) 792
needspace (package) 793
newclude (package) 163
\newfloat 4
newfloat (package) 218
\newline 10951
\newpage 10948
\newtheorem 383
newtxmath (package) 151
newunicodechar (package) 106
nextpage (package) 793
nicefrac (package) 151, 793
\noalign 8240
\nolinebreak 11021
\nolinkurl 8594
nomencl (package) 135, 794
nonfloat (package) 795
nonumonpart (package) 795
\nopagebreak 11025
\nopagecolor 72
nopageno (package) 795
\normalfont 10859
\normalmarginpar 5564
notes (package) 795
notespages (package) 796
nowidow (package) 796
ntheorem (package) 150, 797
\numberline 8940
numindex (option) [tocbibind] 147, 897

O

octave (package) 808
OpenOffice (program) 81
option:
  --shell-escape 105
  [lwarpmk]:
    htmlglossary 134, 711
    printglossary 134, 711
  [tocbibind]:
    numindex 147, 897
  [tocloft]:
    titles 133
  BaseJobname 110, 209
  dvipdfm 107, 212
  dvipdfmx 107, 212
  dvips 107, 212
  GlossaryCmd 110, 134, 212, 711
  HomeHTMLFilename 107, 111, 211
  HTMLFilename 107, 111, 211
  HTMLIndexCmd 109, 211
  HTMLLatexCmd 107, 166, 211
  ImagesDirectory 210
  ImagesName 210
  latexmk 107, 111, 212
  LatexmkIndexCmd 109, 211
  lwarpmk 110, 210, 286
  makeindex 107, 111, 211
  makeindexStyle 107, 144, 210
  mathjax 107, 111, 209
  mathsvg 107, 111, 209
  OSWindows 110, 118, 206, 211
  pdftotextEnc 110, 210

PrintIndexCmd 107, 211
PrintLatexCmd 107, 166, 211
warpHTML 110, 209
warpprint 110, 209
xindy 107, 111, 212
xindyCodepage 107, 210
xindyLanguage 107, 210
xindyStyle 107, 145, 210
\OSPathSymbol 638
OSWindows (option) 110, 118, 206, 211
overpic (package) 155, 809

P

package:
2in1 567
2up 567
a4 567
a4wide 568
a5comb 568
abstract 132, 568
academicons 570
accsupp 571
acro 571
acronym 573
addlines 575
adjmulticol 574
afterpage 575
algorithm2e 575
algorithmicx 579
algorithmx 159
alltt 580
amsmath 580
amsthm 587
anonchap 590
anysize 591
appendix 133, 591
ar 592
arabicfront 593
array 593
arydshln 594
asymptote 155, 596
atbegshi 597
attachfile 597
attachfile2 598
authblk 132, 600
axessibility 601
axodraw2 601
babel 163
backref 602
balance 602
bbding 602
biblatex 607
bibunits 608
bigdelim 159, 608
bigstrut 609
bitpattern 610
blowup 610
booklet 610
bookmark 611
booktabs 611
bophook 613
bounddvi 613
boxedminipage2e 613
breakurl 614
breqn 614
bsheaders 616
bxpapersize 616
bytefield 616
calc 217
cancel 617
canoniclayout 618
capt-of 468
caption 160, 224, 468, 618
cases 621
changebar 621
changepage 622
changes 622
chappg 623
chapterbib 624
chemfig 624
chemformula 625
chemgreek 630
chemmacros 630
chemnum 648
chkfloat 649
chngpage 649
cite 649
CJK 650
CJKutf8 650
cleveref 130, 527
clrdblpg 650
cm-super 105
cmap 106
cmdtrack 651
color 153, 651
colortbl 651
comment 214
continue 653
copyrightbox 653
crop 654
cuted 654
cutwin 654
dblfloatfix 655
dblfnote 655
dcolumn 656
dejavu 105
diagbox 656
dingbat 658
ditaa 169
dprogress 659
draftcopy 659

draftfigure 659
draftwatermark 660
easy-todo 660
ebook 661
ed 661
ellipsis 662
embrac 662
emptypage 663
endfloat 663
endheads 664
endnotes 133, 665
enumerate 665
enumitem 665
environ 218
epigraph 666
epsfig 667
epstopdf 154, 668
epstopdf-base 668
errata 669
eso-pic 670
etoolbox 191
eurosym 671
everyhook 217
everypage 671
everyshi 672
expl3 217
extramarks 672
fancybox 127, 673
fancyhdr 678
fancyref 679
fancytabs 680
fancyvrb 680
figcaps 686
figsize 686
filecontents 217
fitbox 687
fix2col 687
fixme 164, 687
fixmetodonotes 689
flafter 689
flippdf 689
float 159, 690
floatflt 691
floatpag 692
floatrow 160, 692
fltrace 697
flushend 698
fnbreak 698
fncychap 698
fnlineno 699
fnpos 699
fontawesome 700
fontawesome5 701
fontenc 106, 702
fontspec 106
footmisc 702
footnote 703
footnotehyper 704
footnoterange 704
footnpag 705
foreign 705
forest 705
framed 706
ftnright 708
fullminipage 708
fullpage 708
fullwidth 709
fwlw 709
gentombow 709
geometry 216, 710
gettitlestring 217
gloss 134, 710
glossaries 134, 711
gmeometric 713
graphics 151, 713
graphicx 151, 725
grffile 153, 726
grid 726
grid-system 726
gridset 727
hang 727
hanging 728
hypcap 729
hypdestopt 730
hypernat 730
hyperref 130, 457, 730
hyperxmp 735
hyphenat 735
idxlayout 736
ifoddpage 737
ifplatform 191
imakeidx 737
indentfirst 331
index 742
inputenc 106
inputenx 106
intopdf 743
karnaugh-map 743
keyfloat 161, 746
kotex 165
kvoptions 208
layaureo 748
layout 749
leading 749
letltxmacro 191
letterspace 749
lettrine 750
lineno 751
lips 753
listings 753

lltjext 757
lmodern 105, 106
longtable 158, 758
lscape 760
ltablex 760
ltcaption 761
ltxgrid 761
ltxtable 761
lua-check-hyphen 762
lua-visual-debug 762
luacolor 762
luatodonotes 164, 762
lwarp 107
lwarp-patch-komascript 948
lwarp-patch-memoir 950
magaz 764
makeidx 146, 765, 897
marginal 765
marginfit 765
marginfix 766
marginnote 766
marvosym 767
mcaption 767
mdframed 128, 767
memhfixc 776
metalogo 777
mhchem 777
microtype 215, 779
midfloat 780
midpage 780
morefloats 781
moreverb 781
mparhack 782
multicap 782
multicol 783
multicolrule 784
multirow 784
multitoc 787
musicography 787
nameauth 789
nameref 791
natbib 791
nccfancyhdr 792
needspace 793
newclude 163
newfloat 218
newtxmath 151
newunicodechar 106
nextpage 793
nicefrac 151, 793
nomencl 135, 794
nonfloat 795
nonumonpart 795
nopageno 795
notes 795
notespages 796
nowidow 796
ntheorem 150, 797
octave 808
overpic 155, 809
pagegrid 810
pagenote 133, 810
pagesel 811
paralist 811
parnotes 811
parskip 812
pbox 813
pdfcomment 813
pdflscape 814
pdfmarginpar 814
pdfpages 814
pdfprivacy 817
pdfrender 817
pdfsync 817
pdftricks 154, 817
pdfx 818
perltex 168
pfnote 818
phfqit 819
pifont 819
placeins 820
plarydshln 820
plext 820
plextarydshln 821
plextcolortbl 821
polyglossia 163
prelim2e 822
prettyref 822
preview 822
printlen 218
psfrag 154, 823
psfragx 823
pst-eps 824
pstool 155, 824
pstricks 154, 825
pxatbegshi 825
pxeveryshi 825
pxftnright 826
pxjahyper 826
pythontex 168
quotchap 826
quoting 827
ragged2e 828
realscripts 828
refcheck 829
refcount 218
register 830
relsize 124, 830
repeatindex 832
resizegather 833

rmpage 833
romanbar 833
romanbarpagenumber 833
rotating 834
rotfloat 834
rterface 168
rviewport 835
sagetex 168
savetrees 835
scalefnt 836
schemata 836
scrextend 836
scrhack 839
scrlayer 839
scrlayer-notecolumn 841
scrlayer-scrpage 841
scrpage2 842
section 843
sectionbreak 844
sectsty 845
semantic-markup 845
setspace 846
shadow 847
showidx 848
showkeys 848
sidecap 848
sidenotes 849
SIunits 850
siunitx 151, 521, 851
soul 856
soulpos 858
soulutf8 858
splitidx 858
srcltx 860
srctex 860
stabular 861
stfloats 861
struktex 862
subcaption 160, 862
subfig 160, 863
subfigure 867
supertabular 159, 868
sympytex 168
syntonly 870
tabls 870
tabularx 870
tabulary 871
tascmac 871
textarea 873
textcomp 106, 123, 873
textfit 877
textpos 877
theorem 878
thinsp 882
threadcol 882
threeparttable 882
thumb 883
thumbs 883
tikz 153, 884
titleps 885
titleref 888
titlesec 888
titletoc 890
titling 132, 892
tocbasic 896
tocbibind 146, 147, 897
tocenter 898
tocloft 133, 147, 591, 899
tocstyle 905
todo 905
todonotes 164, 906
tram 908
transparent 908
trimclip 909
trivfloat 159, 909
truncate 910
turnthepage 911
twoup 911
typearea 911
typicons 912
ulem 912
umoline 914
underscore 915
units 151, 915
unitsdef 916
upref 916
url 130, 917
uspace 917
varioref 130
varwidth 219
verbatim 218
verse 162, 917, 918
versonotes 919
vertbars 919
vmargin 920
vowel 920
vpe 921
vwcol 921
wallpaper 923
watermark 924
widows-and-orphans 924
wrapfig 924
xbmks 925
xcolor 153, 524, 926
xechangebar 936
xellipsis 936
xetexko-vertical 937
xfakebold 937
xfrac 938
xifthen 218

xltabular 940
xltxtra 941
xmpincl 941
xparse 164, 217
xpatch 191
xpiano 942
xpinyin 942
xstring 218
xtab 159, 943
xunicode 944
xurl 945
xy 946
zhlineskip 946
zwpagelayout 947
\packagediagramname 115, 9178
\pagebreak 11022
\pagecolor 71
pagegrid (package) 810
pagenote (package) 133, 810
\pagenumbering 4923
\pageref 8533
\pagerefPageFor 8532
pagesel (package) 811
\pagestyle 4915
Pandoc (program) 81
\paragraph 6138
paralist (package) 811
\parbox 10401
parnotes (package) 811
\parsemulticolumnalignment 7871
parskip (package) 812
\part 6073
pbox (package) 813
pdfcomment (package) 813
pdfcrop (program) [requirement] 84
pdfLaTeX (program) [requirement] 84
pdflscape (package) 814
pdfmarginpar (package) 814
pdfpages (package) 814
pdfprivacy (package) 817
pdfrender (package) 817
pdfseparate (program) [requirement] 84, 89
pdfsync (package) 817
pdftocairo (program) 152, 522
pdftocairo (program) [requirement] 84, 89
pdftotext (program) [requirement] . 84, 89
pdftotextEnc (option) 110, 210
pdftricks (package) 154, 817
pdfx (package) 818
perl (program) [requirement] 89
perltex (package) 168
pfnote (package) 818
\phantomsection 11085
phfqit (package) 819
picture (environment) 530, 10279
pifont (package) 819
placeins (package) 820
plarydshln (package) 820
Plastex (program) 80
plext (package) 820
plextarydshln (package) 821
plextcolortbl (package) 821
\PN@parnotes@auto 5261
polyglossia (package) 163
\popclose 4692
\postchaptername 6057
\postpartname 6055
\postsectionname 6059
\prechaptername 6057
prelim2e (package) 822
\prepartname 6055
\presectionname 6059
prettyref (package) 822
preview (package) 822
\printauthor 369, 6419, 6438
\printdate 369, 6430, 6440
printglossary (option) [lwarpmk] 134, 711
\printindex 2
PrintIndexCmd (option) 107, 211
PrintLatexCmd (option) 107, 166, 211
printlen (package) 218
\printthanks 369, 371
\printtitle 369, 6403, 6437
program:
[requirement]:
LuaLaTeX 84
MathJax 84
pdfcrop 84
pdfLaTeX 84
pdfseparate 84, 89
pdftocairo 84, 89
pdftotext 84, 89
perl 89
XeLaTeX 84
Adobe 81
AsciiDoc 81
AsciiDoctor 81
Asciidoctor-LaTeX 81
epstopdf 152, 522
Flare 81
FrameMaker 81
GELLMU 80
GladTeX 80
Hevea 80
htlatex 80
InDesign 81
LaTeX2HTML 80
latexmk 167
LaTeXML 80
LibreOffice 81

Linux *118*, *206*
lwarpmk *180*, *286*
lwarpmk epstopdf *152*, *522*
lwarpmk pdftosvg *152*, *522*
Mac OS *118*, *206*
Madcap *81*
make *169*
makeindex *144*
MathJax *148*, *149*, *483*, *484*
MS-Windows *118*, *206*
OpenOffice *81*
Pandoc *81*
pdftocairo *152*, *522*
Plastex *80*
TeX2page *80*
TeX4ht *80*
TeXMaths *174*
TtH *80*
Unix *118*, *206*
Windows *118*, *206*
Word *81*
xindy *145*
project.css (file) *118*
project.lwarpmkconf (file) *245*
psfrag (package) *154*, *823*
psfragx (package) *823*
pst-eps (package) *824*
pstool (package) *155*, *824*
pstricks (package) *154*, *825*
\pushclose 4665
pxatbegshi (package) *825*
pxeveryshi (package) *825*
pxftnright (package) *826*
pxjahyper (package) *826*
pythontex (package) *168*

Q

\qquad *554*, 10971
\quad *554*, 10966
quotchap (package) *826*
quote (environment) 6579
quoting (package) *827*

R

ragged2e (package) *828*
\raggedbottom 4919
\raggedleft 10005
\raggedright 10011
\raisebox 10536
realscripts (package) *828*
\ref 8521
refcheck (package) *829*
refcount (package) *218*
\reflectbox 359
register (package) *830*
relsize (package) *124*, *830*
repeatindex (package) *832*
[requirement]:
LuaLaTeX (program) *84*
MathJax (program) *84*
pdfcrop (program) *84*
pdfLaTeX (program) *84*
pdfseparate (program) *84*, *89*
pdftocairo (program) *84*, *89*
pdftotext (program) *84*, *89*
perl (program) *89*
XeLaTeX (program) *84*
\RequirePackage 965
\resizebox 368
resizegather (package) *833*
\ResumeTabular 8134
\reversemarginpar 5563
\rightline 10019
\rmfamily 10847
rmpage (package) *833*
romanbar (package) *833*
romanbarpagenumber (package) *833*
\rotatebox 317
rotating (package) *834*
rotfloat (package) *834*
\rowcolor 7339
\rownum 7331
rterface (package) *168*
\rule 11029
rviewport (package) *835*

S

sagetex (package) *168*
sample_project.css (file) *118*, *282*
savetrees (package) *835*
\sb 10861
\scalebox 340
scalefnt (package) *836*
schemata (package) *836*
scrextend (package) *836*
scrhack (package) *839*
scrlayer (package) *839*
scrlayer-notecolumn (package) *841*
scrlayer-scrpage (package) *841*
scrpage2 (package) *842*
\scshape 10852
\section 6112
section (package) *843*
sectionbreak (package) *844*
sectsty (package) *845*
semantic-markup (package) *845*
\SetHTMLFileNumber 4785
setspace (package) *846*
\sffamily 10848
\sfrac *938*
shadow (package) *847*
showidx (package) *848*

showkeys (package) 848
sidecap (package) 848
sidenotes (package) 849
SideTOCDepth (counter) 113, 472
\sidetocname 113, 8916
\simplechapterdelim 5883
\sishape 10853
SIunits (package) 850
siunitx (package) 151, 521, 851
\sloppy 4921
\slshape 10858
soul (package) 856
soulpos (package) 858
soulutf8 (package) 858
\sp 10860
splitidx (package) 858
srcltx (package) 860
srctex (package) 860
stabular (package) 861
\StartDefiningMath 4743
\StartDefiningTabulars 4733
stfloats (package) 861
\StopDefiningMath 4747
\StopDefiningTabulars 4737
struktex (package) 862
subcaption (package) 160, 862
subfig (package) 160, 863
subfigure (package) 867
\subparagraph 6146
\subsection 6121
\subsubsection 6129
supertabular (package) 159, 868
sympytex (package) 168
syntonly (package) 870

T

tabbing (environment) 6663
\tableofcontents 115, 8858
tabls (package) 870
tabular (environment) 8283
\TabularMacro 8132
tabularx (package) 870
tabulary (package) 871
tascmac (package) 871
\TeX 11093
TeX2page (program) 80
TeX4ht (program) 80
TeXMaths (program) 174
textarea (package) 873
\textbf 10563
\textcolor 48
textcomp (package) 106, 123, 873
textfit (package) 877
\textgreater 4778
\textit 10608
\textless 4776
\textmd 10554
\textnormal 10655
textpos (package) 877
\textrm 10572
\textsc 10617
\textsf 10581
\textsi 10629
\textsl 10646
\textsubscript 10864
\textsuperscript 10862
\texttt 10590
\textup 10599
\tfl@chapter@fix 910
\thanks 120
\thanksmarkseries 91
\theauthor 370
thebibliography (environment) 9098
\thedate 370
\theHTMLSection 6185
\theHTMLTitleSection 6184
\theHTMLTitleSeparator 6161
theindex (environment) 9006
\thempfootnote 5505
theorem (package) 878
\thetitle 370
thinsp (package) 882
\thispagestyle 4916
threadcol (package) 882
threeparttable (package) 882
thumb (package) 883
thumbs (package) 883
tikz (package) 153, 884
\title 120, 5406
titlepage (environment) 120, 6387
titleps (package) 885
titleref (package) 888
titles (option) [tocloft] 133
titlesec (package) 888
titletoc (package) 890
titling (package) 132, 892
titlingpage (environment) 14, 120
tocbasic (package) 896
[tocbibind]:
  numindex (option) 147, 897
tocbibind (package) 146, 147, 897
tocdepth (counter) 113
tocenter (package) 898
[tocloft]:
  titles (option) 133
tocloft (package) 133, 147, 591, 899
tocstyle (package) 905
todo (package) 905
todonotes (package) 164, 906
\tracinglwarp 185, 1101
tram (package) 908

transparent (package) 908
trimclip (package) 909
trivfloat (package) 159, 909
truncate (package) 910
\ttfamily 10849
TtH (program) 80
turnthepage (package) 911
tutorial.tex (file) 90
twoup (package) 911
typearea (package) 911
typicons (package) 912

U

ulem (package) 912
umoline (package) 914
\underline 10872
underscore (package) 915
units (package) 151, 915
unitsdef (package) 916
Unix (program) 118, 206
\up 10870
upref (package) 916
\upshape 10850
\url 8611
url (package) 130, 917
\UseMinipageWidths 125, 531, 10293
usingOSWindows (boolean) 207
uspace (package) 917

V

varioref (package) 130
varwidth (package) 219
verbatim (environment) 6653
verbatim (package) 218
\VerbatimHTMLWidth (length) 380
\verbatiminput 6645
verse (environment) 2, 6594
verse (package) 162, 917, 918
versonotes (package) 919
vertbars (package) 919
\vleftmargini (length) 162, 379, 918
\vleftskip (length) 162, 379, 918
vmargin (package) 920
vowel (package) 920
vpe (package) 921
vwcol (package) 921

W

wallpaper (package) 923
warpall (environment) 119, 787
warpHTML (environment) 116, 118, 788
warpHTML (option) 110, 209
\warpHTMLonly 116, 119, 778
warpingHTML (boolean) 208
warpingprint (boolean) 208
warpprint (environment) . . . 116, 119, 788
warpprint (option) 110, 209
\warpprintonly 116, 119, 777
watermark (package) 924
widows-and-orphans (package) 924
Windows (program) 118, 206
Word (program) 81
WPMarkFloats (boolean) 173, 232
WPMarkLOFT (boolean) 174, 233
WPMarkMath (boolean) 174, 233
WPMarkMinipages (boolean) 174, 232
WPMarkTOC (boolean) 174, 232
WPTitleHeading (boolean) 175, 233
wrapfig (package) 924

X

xbmks (package) 925
xcolor (package) 153, 524, 926
xechangebar (package) 936
\XeLaTeX 11138
XeLaTeX (program) [requirement] 84
xellipsis (package) 936
\XeTeX 11138
xetexko-vertical (package) 937
xfakebold (package) 937
xfrac (package) 938
\xfracHTMLfontsize 3
xifthen (package) 218
xindy (option) 107, 111, 212
xindy (program) 145
xindyCodepage (option) 107, 210
xindyLanguage (option) 107, 210
xindyStyle (option) 107, 145, 210
xltabular (package) 940
xltxtra (package) 941
xmpincl (package) 941
xparse (package) 164, 217
xpatch (package) 191
xpiano (package) 942
xpinyin (package) 942
xstring (package) 218
xtab (package) 159, 943
xunicode (package) 944
xurl (package) 945
xy (package) 946

Z

zhlineskip (package) 946
zwpagelayout (package) 947

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.

Symbols

\, ... 123
~ 123

A

accents
  in section & file names 349
adapting
  class 179
  document 104
  package 178
affiliation
  multiple authors 131
algorithmic
  with newfloat, trivfloat 910
array
  mhchem 777
author
  HTML meta tag 121, 333
  multiple 131

B

baseline
  tabular 408
bibliography
  HTML page and TOC 132
bitmapped fonts 105
bugs 182

C

Calibre 170
chemistry
  Greek symbols 630
class
  modifying for lwarp 179
code listings 122
compiling
  custom 166
Computer Modern 105
converting
  class 179
  document 104
  package 178
CSS
  file selection 117
  lwarp.css 118
  per HTML page 117
  project-specific changes 118

D

danger icon 190
debugging 182
  HTML debug comments 226
  tracing log 226
Deja Vu 105
description
  HTML meta tag 121, 333
display math
  complicated objects 150
document
  convert existing 104
DVI LaTeX 100, 105
dynamic math 150
dynamic math expressions 307

E

endnotes
  HTML page and TOC 132
EPS image
  converting 102
  using 151, 522
EPUB
  conversion software 170
  HTML conversion settings 170, 232
error messages 182
export
  to word processor 172

F

FAQ 182
filename
  accent in 349
  hashed 488, 512
  international languages 162
  length 114
  simplify 129
  underscore in 111, 124
  unique 114
font
  Computer Modern 105
  Deja Vu 105
  ligatures 106

packages 106
selection 105
size
lateximage 148, 483, 508
math, SVG 148, 483, 508
xfrac 938
type 1 vector 105
type 3 bitmapped 105
footnotes 333
foreign
section names 162
framed objects 126
Frequently Asked Questions 182

G

generator
HTML meta tag 362
GIF images 152, 522
gloss 134
glossaries
HTML page and TOC 132
language 134
options 134
processing 101
graphics
formats 151, 522
Greek
chemistry symbols 630

H

hash
SVG image filename 488, 512
heading, word processor 175
horizontal space 123
between minipages 553
HTML
conversion settings 111
debug comments 226
EPUB 170, 232
word processor 172, 232
conversion suggestions 121
entities, conversion 122
filename generation 116
headings 189
meta tag
author 121, 333
description 121, 333
generator 362
title 114, 121, 332, 333
viewport 363
selecting print/HTML definitions . . 227
tabular column conversion 409
verbatim, in 122
hyperref
and xindy 135

I

icon
warning 190
image
GIF 152, 522
graphicx package 522
hashed filename 488, 512
PDF or EPS
converting 102, 152, 522
using 151, 522
PNG and JPG 152, 522
processing 286
\includegraphics
using 151, 522
index
custom makeindex style file 144
custom xindy style file 145
custom display styles 146
formatting 478
HTML page and TOC 132
letter headings 738
placement and TOC options . . . 146, 897
processing 94, 95, 135
see, seealso, ranges 135
source code 135
table of contents 146, 897
UTF-8 106
xindy and hyperref 135
inline math
complicated objects 150
international
section names 162
item
empty 384

J

JAVASCRIPT
MATHJAX 147, 482
JPG images 152, 522

L

label
in HTML 366
math environment 503
language
glossaries 134
language HTML metadata 360
lateximage
font size 148, 483, 508
processing 286
Latin Modern font 105
LIBREOFFICE
conversion recommendations 175
import into 172
section headings 175
ligatures 106, 215

line numbers 190
list
  empty item 384
  filename 124
listings, program code 122
locale 169
LuaLaTeX
  detection 191
  file & section names 349
lwarp
  loading 107
  options 107
lwarp.ist
  customizing 144
lwarp.xdy
  customizing 145
lwarpmk
  customizing 180

M

make utility 169
makeindex
  customizing 144
margin
  numbers 190
  tags 190
markup languages 81
math
  display with complicated objects . . 150
  dynamic 150, 307
  font size — SVG 148, 483, 508
  inline with complicated objects . . . 150
  MathJax custom functions . . . 149, 484
  MathJax summary 148, 483
  mathjax option 209
  mathsvg option 209
  mhchem 777
  SVG summary 148, 483
  word processor conversion 174
MathJax
  custom functions 149, 484
  mathjax option 209
  summary 148, 483
MD5 hash
  SVG image filename 488, 512
memoir
  verse 918
meta tag, HTML
  author 121, 333
  description 121, 333
  generator 362
  title 121, 332, 333
  viewport 363
minipage
  framed 126
  horizontal space between 553
modifying
  class 179
  document 104
  package 178
\multicolumn
  with \multirow 786
\multirow
  with \multicolumn 786

N

newfloat
  with trivfloat, algorithmic 910
nomencl 135
numbers
  left margin 190

P

package
  modifying for lwarp 178
  required 215
\par
  hooks 329
PDF image
  converting 102
  using 151, 522
Perl 89
PNG images 152, 522
Poppler 84, 89
print
  selecting print/HTML definitions . . 227
problems 182
program listings 122
programs
  utility 83
\published 376

S

section
  depths 189
  file names 129
  heading, word processor 175
  international languages 162
settings
  CSS project-specific 118
  CSS selection 117
  HTML conversion 111
  lwarp package options 107
  selecting output 118
  title page 120
shell escape 105
siunitx
  with *TeXMaths* 174
space
  horizontal 123
    between minipages 553
stack depths 189
\subtitle 376

SVG
converting from PDF or EPS 102
dynamic math 307
image processing 286
math summary 148, 483
mathsvg option 209

T

tabular
baseline 408
HTML columnn conversion 409
macros inside 157, 392
\multicolumn with \multirow 786
tikz
catcodes 884
dollar redefined 884
title
HTML meta tag 121, 332, 333
titlepage
\subtitle and \published 376
tracing log 226
trivfloat
with newfloat, algorithmic 910
troubleshooting 182
HTML debug comments 226
tracing log 226
type 1 vector fonts 105
type 3 bitmapped fonts 105

U

underscore
filename 111, 124
Unicode
enhanced coverage 105
file & section names 349
input characters 204
selection 105
UTF-8
enhanced coverage 105
file & section names 349
index 106
locale 169
selection 105
utility
programs 83

V

vector fonts 105
verbatim
code and HTML tags 122
viewport
HTML meta tag 363

W

warning icon 190
word processor
conversion recommendations 175
HTML conversion settings 172, 232
section headings 175

X

XeLaTeX
detection 191
file & section names 349
xindy
and hyperref 135
customizing 145
xparse
warnings 165

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 with higher page numbers 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 for a given package.

A

abstract
- missing TOC . . . 132, 568

acronym
- multiply-defined labels . . . 573

\AddSubtitlePublished . . . 376
affiliation . . . 370
alt tags . . . 150, 183
AMSmath
- ntheorem
  - numbering . . . 151, 797

appendix
- incorrect TOC link . . . 133, 591

array
- chemformula . . . 164

authblk
- \theauthor . . . 369, 370
- titling . . . 132, 369, 600, 892

author
- affiliation . . . 370
- formatting . . . 120

B

babel
- French . . . 163

backref . . . 602
backref . . . 130
bibliography
- HTML page and TOC . . . 132

bibtex
- \etalchar . . . 133
- Improper \prevdepth . . . 133

bigdelim . . . 159, 608
boxes . . . 124
breqn
- darray . . . 614
- MathJax . . . 614

C

Calibre
- EPUB conversion . . . 170

caption
- numbering . . . 160
- options . . . 160

cases
- MathJax . . . 621

chemfig . . . 164
chemformula
- MathJax . . . 164, 625

chemgreek
- fontspec mapping . . . 630
- text-mode symbols . . . 630

chemmacros
- \makepolymerdelims . . . 631
- redox reactions . . . 631

colortbl . . . 158, 393, 651
cross reference
- incorrect link . . . 184
- missing . . . 183

CSS . . . 185

D

ditaa . . . 169
duplcate filename . . . 129

E

endnotes
- HTML page and TOC . . . 132

\ensuremath . . . 182
epstopdf . . . 154, 668
EPUB
- encoding . . . 170
- page order . . . 170
- search order . . . 170
- section breaks . . . 171

\etalchar . . . 133

F

fancybox
- \VerbatimFootnotes . . . 131, 674, 680

fancyvrb
- \VerbatimFootnotes . . . 131, 674, 680

figure
macro in name . . . 183
file
inaccessible . . . 113
filename
accents . . . 349
corrupted . . . 129, 162
duplicate . . . 129
international, UTF-8 . . . 162
Korean . . . 165
macro in name . . . 129
math in . . . 104, 129, 147, 482
Missing $ inserted . . . 182
fixme . . . 164, 687
float
alignment . . . 123, 159
out of sequence . . . 184
floatrow
\FBwidth and \FBheight . . . 160, 693
with subfig . . . 160, 692
floatrow
\ttabbox . . . 156, 391
font
ligatures . . . 106
missing symbols . . . 123, 873
monospace . . . 106
package conflicts . . . 105
selection . . . 105
UTF-8 . . . 105
fontspec . . . 105
footnote
displaymath . . . 131, 674, 680
in math . . . 131
in sectioning command . . . 130
memoir . . . 131, 334
misnumbered . . . 130
paragraph tags . . . 131, 674, 680
sectioning, footmisc . . . 131, 333
sectioning, verbatim . . . 131, 674, 680
title . . . 369
verbatim . . . 131
\VerbatimFootnotes . . . 131, 674, 680
forest . . . 705
formatting
\bfseries etc. . . . 122, 541
frames . . . 126

G

gloss . . . 134
glossaries
HTML page and TOC . . . 132
makeglossaries not found . . . 134, 711
page numbers . . . 135, 712
style . . . 134, 711
graphics
\rotatebox, \scalebox, \reflectbox . . . 153, 523
image format priorities . . . 152, 523
images out of order . . . 96, 184
optional arguments . . . 152, 523
scale option . . . 104, 151, 522
scaled . . . 714
viewport . . . 152, 523
Greek packages . . . 630
grffile . . . 153, 726

H

horizontal spacing . . . 160, 863
HTML
&, <, > . . . 122, 541
alt tags . . . 150
author . . . 333
corrupted . . . 120, 124, 182, 185
entities . . . 122, 541
image appear as HTML code . . . 96, 184
inaccessible pages . . . 113, 115
starred section . . . 132
invalid . . . 122
missing pages
filename not unique . . . 114
recompile . . . 185
SideTOCDepth and FileDepth . . . 113
SideTOCDepth and tocdepth . . . 113
page did not update . . . 113, 185
settings
changed . . . 111
undefined . . . 182
validation . . . 122
\HTMLAuthor . . . 333
HTMLIndexCmd
filenames . . . 109
hyperref
backref . . . 130
comments between arguments . . . 130
incorrect link . . . 184

I

image
appears as HTML code . . . 96, 184
format priorities . . . 152, 523
incorrect . . . 96, 184
viewport . . . 152, 523
Improper \prevdepth
bibtex . . . 133
boxes . . . 182
\includegraphics
optional arguments . . . 152, 523
scale . . . 104, 151, 522
index
empty . . . 146

empty link 129, 183
formatting 478
HTML page and TOC 132
missing entries 135
numbers, not links 218
reference ranges 145
see and seealso 145
styling references 145
xindy and hyperref 145
xstring bug 218

K

keyfloat 161, 746

L

label
\nameref empty 129, 183
characters 129, 183
Label(s) may have changed 182
\LateximageFontSizeName 509
Leaders not followed by proper glue . . 182
LibreOffice
import 172
ligatures 106
LINUX 118, 206
list
empty item 124, 384
listings 184
lists
label formatting 384
locale 169
ltxtable
numbering 761
lualaTeX 105
\LWRbackslash 166
\LWRdollar 166
\LWRhash 166
\LWRopquote 166
\LWRopseq 166
\LWRpercent 166

M

MAC OS 118, 206
makeglossaries
not found 134, 711
\makelabel 384
\marginpar 124, 338
math
alt tags 183
appears as HTML code 96, 184
chemformula 164
custom macros 183
dynamic 307
equation numbering
ntheorem 151, 797
file name 104, 129, 147, 482
footnote 131
in TeX boxes 148, 483
incorrect 96, 184, 307
non-math contents 183
section name 104, 129, 147, 482
slow or failed compile 183
subequations 149, 484
tabbing 155, 382
Ti*k*z 150
MATHJAX
breqn 614
cases 621
chemformula 164, 625
errors 150, 183, 307
mhchem 777
siunitx 149, 151, 484, 521, 851
subequations 149, 484
unsupported packages . . 149, 183, 485
memoir
options clash 161, 950
version clash 161, 950
mhchem
MATHJAX 777
nested dollar signs 777
minipage
alignment 125, 532
horizontal space between 553
in a span 125, 531
inline 125, 531
multicols, width in 125, 531
size 125, 531
tabular, width in 125, 531
Misplaced \noalign 158, 758
tabular
rules 158, 393
Misplaced \omit
tabular 230
Misplaced alignment tab character &
floatrow 160, 692
frames 126
supertabular 159, 868, 943
tabular
macros 156, 307, 391
Missing $ inserted
filename or URL 182
morewrites 182
morewrites 146
MS-WINDOWS 118, 206
multiply-defined labels
acronym 573
\multirow and \multicolumn 157, 392, 786
multirow 156, 391

N

newclude 163
newtxmath 151

nicefrac 151
No room for a new \write 182
nomencl 135
ntheorem 797
font 150, 797
numbering 151, 797

O

operating system 118, 206
options clash
memoir 161, 950
overpic 155, 809

P

package
version numbers with memoir . 161, 950
page
inacessible 113
page counter
references 130
SVG images 96, 184
page numbers 130, 184, 527
pdftricks 154, 818
perltex 168
pfnote
numbering 131, 818
polyglossia
Undefined control sequence 163
poppler
Syntax Warning (ligature) 700
PrintIndexCmd 109
psfrag 154
pstool 155, 824
pstricks 154, 825
pythontex 168

R

reference
% character between arguments . . 184
empty link 129, 183
incorrect link 184
label characters 129, 183
missing or incorrect 183
page number 130, 184, 527
repeatindex 832
rterface 169

S

sample_projects.css
overwritten 117
scale (\includegraphics option) . . . .
. . . . . 104, 151, 522
sectioning
accents 349
duplicate name 129
international language 162
macro in name 129, 183
math in name 104, 129, 147, 482
missing 113
starred section 132
word processor import 175
siunitx
MathJax 151, 521, 851
with TeXMaths 174
siunitx 151, 521, 851
MathJax 149, 484
S column 158, 393
tabular S and s columns . 151, 521, 851
splitidx
\thepage and \AtWriteToIndex 137, 859
subcaption
numbering 160
subfig
options 160, 863
with floatrow 160, 692
SVG image
appears as HTML code 96, 184
incorrect 96, 184
math incorrect 307
out of order 96, 184
sympytex 169
Syntax Warning (ligature) 700

T

tabbing
math 155, 382
table
macro in name 183
numbering 761
Table of Contents
missing 115
tabular
* column specification 157, 392
baselines 408
corrupt rows 156, 391, 440
\endhead, \endfoot, \endlastfoot
. . . . . 158, 758
inside an environment . . 156, 391, 440
\kill 158, 759
macro inside 157, 392
Misplaced \noalign 158, 393
Misplaced alignment tab character &
macros 156, 391
rules 307
\multicolumn with \multirow 786
\multirow 444
multirow 156, 391
\newcolumntype 157, 392
numbering 940
row corruption 157, 392
rules 158, 393
S column 158, 393

TeXMaths 174
text
&, <, > 122, 541
\bfseries etc. 122, 541
corrupted 105
textcomp
missing symbols 123, 873
\theauthor and authblk 369, 370
tikz
in math 150
matrices, & 153, 884
title
affiliation 370
newlines 120
\thanks 369
titling
authblk 132, 369, 600, 892
hooks 376
tocloft
chapter titles 133, 147, 591, 899
todonotes 164, 762, 906
tracing lwarp 185
tram 908
transparent 908

U

Undefined control sequence
polyglossia 163
Unicode
fonts 105
UTF-8 locale 169
units 151
UNIX 118, 206
URL
Missing $ inserted 182
UTF-8
locale 169

V

verbatim
footnote 131
framed 127, 673
VerbatimFootnotes 131, 674, 680
verse
spacing 162, 379, 918
version numbers
with memoir 161, 950
viewport 152, 523

W

warpall 182
warpHTML 119, 182
warpprint 119, 182
WINDOWS 118, 206
word processor
import 172
sectioning headings 175

X

xeLaTeX 105
xfakebold 937
xfrac 938
xindy
hyperref 145
options
HTMLIndexCmd 109
LatexmkIndexCmd 109
PrintIndexCmd 109
xltabular
numbering 940
xparse 164
xstring 218
xy 946

Index of Indexes

C

Change History 975

G

General Index 1020

I

Index of Objects 1001

T

Troubleshooting Index 1024