Lwarp

LaTeX HTML5

The lwarp package

LaTeX to HTML

v0.79 — 2020/02/01

© 2016–2020 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 error testing is provided for configuration files, package load order, and image generation.

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

Complicated tables are supported, which copy/paste well into LibreOffice Writer.

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

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

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

To update existing projects, see section 1: Updates.

Lwarp is still in development. Changes are likely.

License:

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.

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

Support TEX development **2**

List of Figures **38**

List of Tables **38**

1 Updates **39**

2 Introduction **62**

2.1 Typesetting conventions 64

2.2 Supported packages and features 65

3 Alternatives **71**

3.1 internet class 71

3.2 TEX4HT 71

3.3 Translators 71

3.4 AsciiDoc and Asciidoctor 72

3.4.1 Asciidoctor-LATEX 72

3.5 Pandoc 72

3.6 Word processors 72

3.7 Commercial systems 72

3.8 Comparisons 72

4 Installation **74**

4.1 Installing the lwarp package 76

4.2 Installing the *lwarpmk* utility 77

4.2.1 Using a local copy of *lwarpmk* 79

4.3 Installing additional utilities 79

5 Tutorial **81**

5.1 Starting a new project 81

5.2 Compiling the print version with *lwarpmk* 85

5.3 Compiling the HTML version with *lwarpmk* 86

5.4 Generating the SVG images 87

5.5 Using MathJax for math 88

5.6 Changing the CSS style . . . 89
5.7 Customizing the HTML output . . . 89
5.8 Using *latexmk* . . . 90
5.9 Using XeLaTeX or LuaLaTeX . . . 91
5.10 Using DVI LaTeX . . . 91
5.11 Using a glossary . . . 92
5.11.1 gloss package . . . 92
5.11.2 glossaries package . . . 92
5.12 Cleaning auxiliary files . . . 93
5.13 Cleaning auxiliary and output files . . . 93
5.14 Cleaning the images from the `<project>-images` directory . . . 93
5.15 Converting PDF or EPS images to SVG . . . 93
5.16 Creating HTML from an incomplete compile . . . 93
5.17 Processing multiple projects in the same directory . . . 93
5.18 Using the *make* utility . . . 94
5.19 What next? . . . 95
6 Converting an existing document . . . 96
7 Additional details . . . 97
7.1 Localization . . . 97
7.2 Accessibility . . . 97
7.3 Shell escape . . . 98
7.4 Font and UTF-8 support . . . 98
7.4.1 Indexes, glossaries, and encoding . . . 100
7.5 lwarp package loading and options . . . 101
7.6 Customizing the HTML output . . . 106
7.6.1 Example HTML file naming . . . 112
7.7 Customizing the CSS . . . 113
7.8 Selecting the operating system . . . 114
7.9 Selecting actions for print, HTML, or MathJax output . . . 114
7.10 Commands to be placed into the `warpprint` environment . . . 115

7.11 Title page 116
7.12 HTML page meta descriptions 117
7.13 HTML homepage meta title 117
7.14 HTML page meta author 117
8 Special cases and limitations 118
8.1 Things to avoid 118
8.1.1 Invalid HTML 119
8.2 Formatting 119
8.2.1 Text formatting 119
8.2.2 Horizontal space 119
8.2.3 Text alignment 119
8.2.4 Accents 120
8.2.5 textcomp package 120
8.2.6 Superscripts and other non-math uses of math mode 120
8.2.7 Empty \item followed by a new line of text or a nested list: . . . 120
8.2.8 Filenames and URLs in lists or footnotes 120
8.2.9 relsize package 120
8.3 Boxes and minipages 120
8.3.1 Marginpars 120
8.3.2 Save Boxes 121
8.3.3 Minipages 121
8.3.4 Side-by-side minipages 122
8.3.5 Framed minipages and other environments 122
8.3.6 fancybox package 123
8.3.7 mdframed package 124
8.4 Section names 125
8.5 Cross-references 126
8.5.1 Page references 126
8.5.2 cleveref and varioref packages 126
8.5.3 Hyperlinks, hyperref, and url 127

8.5.4 Footnotes and page notes . . . 127
8.5.5 xr, xr-hyper, and xcite packages . . . 128
8.6 Front and back matter . . . 129
8.6.1 Custom classes with multiple authors and affiliations . . . 129
8.6.2 Starred chapters and sections . . . 129
8.6.3 abstract package . . . 130
8.6.4 titling and authblk . . . 130
8.6.5 tocloft package . . . 130
8.6.6 appendix package . . . 130
8.6.7 pagenote package . . . 130
8.6.8 endnotes package . . . 130
8.6.9 *BibTeX* . . . 131
8.6.10 xcite package . . . 131
8.6.11 gloss package . . . 131
8.6.12 glossaries package . . . 131
8.6.13 nomencl package . . . 132
8.6.14 Indexing overview . . . 132
8.6.15 Indexing with basic LaTeX and makeidx . . . 133
8.6.16 Indexing with index . . . 134
8.6.17 Indexing with splitidx . . . 134
8.6.18 Indexing with imakeidx . . . 136
8.6.19 Indexes with memoir . . . 139
8.6.20 Using a custom *makeindex* style file . . . 142
8.6.21 Using a custom *xindy* style file . . . 143
8.6.22 Additional indexing limitations . . . 143
8.6.23 Index positions, TOC, tocbibind . . . 144
8.7 Math . . . 145
8.7.1 Math in section names . . . 145
8.7.2 Rendering tradeoffs . . . 145
8.7.3 SVG option . . . 146

8.7.4 MathJax option . . . 146
8.7.5 Customizing MathJax . . . 147
8.7.6 MathJax limitations . . . 147
8.7.7 Catcode changes . . . 148
8.7.8 Complicated inline math objects . . . 148
8.7.9 Complicated display math objects . . . 148
8.7.10 ntheorem package . . . 149
8.7.11 mathtools package . . . 149
8.7.12 siunitx package . . . 150
8.7.13 units and nicefrac packages . . . 150
8.7.14 physics package . . . 151
8.7.15 newtxmath package . . . 151
8.8 Graphics . . . 151
8.8.1 tikz package . . . 153
8.8.2 grffile package . . . 153
8.8.3 color package . . . 153
8.8.4 xcolor package . . . 153
8.8.5 epstopdf package . . . 154
8.8.6 pstricks package . . . 154
8.8.7 pdftricks package . . . 154
8.8.8 psfrag package . . . 154
8.8.9 pstool package . . . 155
8.8.10 asymptote package . . . 155
8.8.11 overpic package . . . 155
8.8.12 Multimedia packages . . . 155
8.9 Tabbing . . . 156
8.10 Tabular . . . 156
8.10.1 `tabular` environment . . . 156
8.10.2 multirow package . . . 158
8.10.3 longtable package . . . 159

8.10.4 threeparttablex package . . . 160
8.10.5 supertabular and xtab packages . . . 160
8.10.6 colortbl package . . . 160
8.10.7 ctable package . . . 160
8.10.8 bigdelim package . . . 161
8.11 Floats . . . 161
8.11.1 Float contents alignment . . . 161
8.11.2 float, trivfloat, and/or algorithmicx together . . . 161
8.11.3 caption and subcaption packages . . . 161
8.11.4 subfig package . . . 162
8.11.5 floatrow package . . . 162
8.11.6 keyfloat package . . . 163
8.12 KOMA-SCRIPT classes . . . 163
8.13 MEMOIR class . . . 163
8.14 International languages . . . 164
8.15 Miscellaneous packages . . . 164
8.15.1 verse and memoir . . . 164
8.15.2 newclude package . . . 165
8.15.3 babel package . . . 165
8.15.4 polyglossia package . . . 165
8.15.5 todonotes and luatodonotes packages . . . 166
8.15.6 fixme . . . 166
8.15.7 chemfig package . . . 166
8.15.8 chemformula package . . . 166
8.15.9 mhchem package . . . 166
8.15.10 xparse package . . . 166
8.15.11 kotex package . . . 167
9 Compiling using custom shell commands . . . 168
9.1 Command options . . . 168
9.2 Literal character macros . . . 168

9.3 *latexmk* . . . 169
9.4 perltex package . . . 170
9.5 pythontex package . . . 170
9.6 Other packages . . . 170
9.7 *make* program . . . 171
9.8 UTF-8 locale . . . 171
10 EPUB conversion . . . 172
11 Word-processor conversion . . . 174
11.1 Activating word-processor conversion . . . 174
11.2 Additional modifications . . . 175
11.3 Recommendations . . . 177
11.4 Limitations . . . 178
12 Modifying lwarp . . . 179
12.1 Creating a development system . . . 179
12.2 Modifying a package for lwarp . . . 180
12.2.1 Adding a package to the lwarp.dtx file . . . 181
12.3 Modifying a class for lwarp . . . 181
12.4 Testing lwarp . . . 182
12.5 Modifying *lwarpmk* . . . 182
13 Troubleshooting . . . 184
13.1 lwarp package error conditions and warnings . . . 184
13.1.1 Configuration file `lwarpmk.conf` . . . 184
13.1.2 Image generation with **`lwarpmk limages`** . . . 184
13.1.3 Default bitmapped font . . . 185
13.1.4 Packages . . . 185
13.1.5 Compiling . . . 185
13.2 Using the lwarp package . . . 186
13.2.1 Debug tracing output . . . 190
13.3 Compiling the `lwarp.dtx` file . . . 190
14 Trademarks . . . 191

1 lwarp.sty 192

15 Implementation. 192

16 Section depths and HTML headings. 193

17 Source code 194

18 Detecting the TEX engine — *pdflatex*, *lualatex*, *xelatex* 195

19 Early package requirements 195

20 Package load order 195

20.1 Tests of package load order 196

20.2 Error for disallowed packages and classes loaded before lwarp . . . 198

20.3 Enforcing package loading after lwarp. 199

21 MD5 hashing. 208

22 pdfLATEX T1 and UTF-8 encoding. 208

23 Unicode input characters 209

24 Avoid a bitmapped font 209

25 Upright quotes 210

26 Miscellaneous tools 210

26.1 Lengths and units. 210

26.2 Counters 211

26.3 Patching 211

26.4 Chinese text isolation 211

26.5 Inserting vertical space 212

26.6 Argument selection 212

26.7 Global boxes 212

26.8 Converting a macro name to a cs name 213

27 Operating-System portability. 213

27.1 Literal characters 213

27.2 Common portability code 215

27.3 UNIX, LINUX, and MAC OS 215

27.4 MS-WINDOWS 215

28 Package options . . . 215
28.1 Additional options support . . . 220
28.2 Conditional compilation . . . 221
29 Required packages . . . 223
30 Loading packages . . . 229
31 Additional required packages . . . 234
32 File handles . . . 235
33 Include a file . . . 235
34 Copying a file . . . 236
35 Debugging messages . . . 237
36 Defining print and HTML versions of macros and environments . . . 238
37 HTML-conversion output modifications . . . 242
37.1 User-level controls . . . 242
37.2 Heading adjustments . . . 244
38 Remembering original formatting macros . . . 245
39 Accents . . . 247
40 Configuration files . . . 249
40.1 Decide whether to generate configuration files . . . 249
40.2 <project>_html.tex . . . 249
40.3 *lwarpmk* configuration files . . . 249
40.3.1 Helper macros . . . 250
40.3.2 lwarpmk.conf . . . 255
40.3.3 <project>.lwarpmkconf . . . 256
40.4 lwarp.css . . . 256
40.5 lwarp_sagebrush.css . . . 285
40.6 lwarp_formal.css . . . 290
40.7 sample_project.css . . . 294
40.8 lwarp.ist . . . 294
40.9 lwarp.xdy . . . 295
40.10 lwarp_one_limage.cmd . . . 296

40.11 lwarp_mathjax.txt . . . 296
40.12 lwarpmk.lua — lwarpmk option . . . 298
41 Stacks . . . 316
41.1 Assigning depths . . . 317
41.2 Closing actions . . . 317
41.3 Closing depths . . . 318
41.4 Pushing and popping the stack . . . 318
42 Data arrays . . . 319
43 Localizing catcodes . . . 320
44 Localizing dynamic math . . . 321
45 HTML entities . . . 322
46 HTML filename generation . . . 323
47 Homepage link . . . 325
48 \LWRPrintStack diagnostic tool . . . 326
49 Closing stack levels . . . 327
50 PDF pages and styles . . . 328
51 HTML tags, spans, divs, elements . . . 329
51.1 Mapping LaTeX sections to HTML sections . . . 329
51.2 Babel-French tag modifications . . . 330
51.3 HTML output formatting . . . 331
51.4 HTML tags . . . 331
51.5 Block tags and comments . . . 333
51.6 Div class and element class . . . 334
51.7 Single-line elements . . . 335
51.8 HTML5 semantic elements . . . 336
51.9 High-level block and inline classes . . . 336
51.10 Closing HTML tags . . . 338
52 Paragraph handling . . . 339
53 Paragraph start/stop handling . . . 342
54 Indentfirst . . . 345

55 Page headers and footers 345
56 CSS 346
57 MathJax script 346
58 Title, HTML meta author, HTML meta description 347
59 Footnotes 348
59.1 Regular page footnotes 349
59.2 Minipage footnotes 349
59.3 Titlepage thanks 349
59.4 Regular page footnote implementation 349
59.5 Minipage footnote implementation 352
59.6 Printing pending footnotes 353
60 Marginpars 354
61 Splitting HTML files 355
61.1 Sanitizing expressions for HTML 360
61.2 Customizing MathJax 361
62 Sectioning 367
62.1 User-level starred section commands 367
62.2 Book class commands 368
62.3 Sectioning support macros 369
62.4 Pre- and post- sectioning names 376
62.5 `\section` and friends 376
63 Starting a new file 378
64 Starting HTML output 381
65 Ending HTML output 384
66 Title page 386
66.1 Setting the title, etc. 387
66.2 `\if@titlepage` 388
66.3 Changes for `\affiliation` 388
66.4 Printing the thanks 389
66.5 Printing the title, etc. in HTML 389

66.6 Printing the title, etc. in print form . . . 390
66.7 \maketitle for HTML output . . . 391
66.8 \published and \subtitle . . . 394
67 Abstract . . . 395
68 Quote and verse . . . 396
68.1 Attributions . . . 396
68.2 Quotes, quotations . . . 396
68.3 Verse . . . 397
68.3.1 LaTeX core verse environment . . . 397
68.3.2 verse and memoir . . . 398
69 Verbatim and tabbing . . . 398
70 Theorems . . . 401
71 Lists . . . 401
71.1 List environment . . . 402
71.2 Itemize . . . 405
71.3 Enumerate . . . 406
71.4 Description . . . 406
71.5 Patching the lists . . . 407
72 Tabular . . . 408
72.1 Limitations . . . 408
72.2 Temporary package-related macros . . . 410
72.2.1 arydshln . . . 410
72.3 Token lookahead . . . 411
72.4 Tabular variables . . . 411
72.4.1 Multicolumn variables . . . 415
72.4.2 Longtable variables . . . 415
72.4.3 Midrule variables . . . 415
72.5 Handling &, @, !, and bar . . . 415
72.5.1 Handling & . . . 417
72.5.2 Filling an unfinished row . . . 418

72.6 Handling \\ . . . 420
72.7 Looking ahead in the column specifications . . . 421
72.8 Parsing @, >, <, !, bar columns . . . 421
72.9 Parsing ‘l’, ‘c’, or ‘r’ columns . . . 425
72.10 Parsing ‘p’, ‘m’, or ‘b’ columns . . . 425
72.11 Parsing ‘w’ columns . . . 426
72.12 Parsing ’*’ columns . . . 426
72.13 Parsing ‘D’ columns . . . 426
72.14 Expanding the star column specifications . . . 427
72.15 Parsing the column specifications . . . 428
72.16 colortbl and xparse tabular color support . . . 433
72.17 Starting a new row . . . 434
72.18 Printing vertical bar tags . . . 435
72.19 Printing @ or ! tags . . . 435
72.20 Data opening tag . . . 436
72.21 Midrules . . . 438
72.22 Cell colors . . . 444
72.23 Multicolumns . . . 447
72.23.1 Parsing multicolumns . . . 447
72.23.2 Multicolumn factored code . . . 451
72.23.3 Multicolumn . . . 454
72.23.4 Longtable captions . . . 454
72.23.5 Counting HTML tabular columns . . . 456
72.24 Multirow if not loaded . . . 458
72.25 Multicolumnrow . . . 458
72.26 Utility macros inside a table . . . 459
72.27 Special-case tabular markers . . . 460
72.28 Checking for a new table cell . . . 461
72.29 \mrowcell . . . 463
72.30 \mcolrowcell . . . 464

72.31 HTML tabular environment . . . 464
73 Cross-references . . . 471
73.1 Setup . . . 471
73.2 New lwarp labels. . . . 473
73.3 Labels . . . 474
73.4 References . . . 476
73.5 Hyper-references . . . 478
74 Floats . . . 483
74.1 Float environment . . . 483
74.2 Float tracking . . . 485
74.3 Caption inside a float environment . . . 487
74.4 Caption and LOF linking and tracking . . . 487
75 Table of Contents, LOF, LOT . . . 490
75.1 Reading and printing the TOC . . . 491
75.2 High-level TOC commands. . . . 494
75.3 Side TOC . . . 494
75.4 Low-level TOC line formatting . . . 495
76 Index and glossary . . . 499
77 Bibliography presentation . . . 502
78 Restoring original formatting . . . 503
79 Math . . . 505
79.1 Limitations . . . 505
79.2 HTML alt tag names . . . 505
79.3 Inline and display math . . . 506
79.4 MathJax support . . . 519
79.5 Equation environment . . . 521
79.6 \displaymathnormal and \displaymathother . . . 523
79.7 AMS Math environments . . . 525
79.7.1 Support macros . . . 525
79.7.2 Environment patches . . . 526

80 Lateximages 529
80.1 Description 529
80.2 Support counters and macros 530
80.3 Font size 530
80.4 Equation numbers 531
80.5 HTML alt tags 532
80.6 lateximage environment 532
81 center, flushleft, flushright 539
82 Preloaded packages 540
83 siunitx 542
84 Graphics print-mode modifications 543
84.1 General limitations 543
84.2 Print-mode modifications 545
85 xcolor boxes 545
86 chemmacros environments 548
87 cleveref 549
88 picture environment 552
89 Minipages and Boxes 552
89.1 Computed lengths 553
89.2 Virtual page size 553
89.3 Footnote handling 554
89.4 Minipage handling 554
89.5 \parbox, \mbox, \makebox, \framebox, \fbox, \raisebox 558
90 Direct formatting 563
91 Skips, spaces, font sizes 576
92 \phantomsection 584
93 \LaTeX and other logos 584
94 \AtBeginDocument, \AtEndDocument 587
95 Loading KOMA-SCRIPT class patches 588
96 Loading MEMOIR class patches 588

97 ut* class patches . . . 588
98 CTEX patches . . . 589
99 kotexutf patches . . . 590
100 babel and polyglossia warnings . . . 591
101 MATHJAX warnings . . . 592
2 lwarp-2in1.sty . . . 594
3 lwarp-2up.sty . . . 594
4 lwarp-a4.sty . . . 594
5 lwarp-a4wide.sty . . . 595
6 lwarp-a5comb.sty . . . 595
7 lwarp-abstract.sty . . . 595
8 lwarp-academicons.sty . . . 597
9 lwarp-afterpage.sty . . . 598
10 lwarp-accessibility.sty . . . 598
11 lwarp-accsupp.sty . . . 599
12 lwarp-acro.sty . . . 599
13 lwarp-acronym.sty . . . 601
14 lwarp-adjmulticol.sty . . . 602
15 lwarp-addlines.sty . . . 603
16 lwarp-afterpage.sty . . . 603
17 lwarp-algorithm2e.sty . . . 604
18 lwarp-algorithmicx.sty . . . 607
19 lwarp-alltt.sty . . . 608
20 lwarp-amsmath.sty . . . 609
21 lwarp-amsthm.sty . . . 612
22 lwarp-anonchap.sty . . . 616
23 lwarp-anysize.sty . . . 616
24 lwarp-appendix.sty . . . 617

25 lwarp-ar.sty 617
26 lwarp-arabicfront.sty 619
27 lwarp-array.sty 619
28 lwarp-arydshln.sty 620
29 lwarp-asymptote.sty 622
30 lwarp-atbegshi.sty 623
31 lwarp-attachfile.sty 623
32 lwarp-attachfile2.sty 624
33 lwarp-authblk.sty 626
34 lwarp-autonum.sty 627
35 lwarp-awesomebox.sty 628
36 lwarp-axessibility.sty 629
37 lwarp-axodraw2.sty 630
38 lwarp-backnaur.sty 630
39 lwarp-backref.sty 630
40 lwarp-balance.sty 631
41 lwarp-bbding.sty 631
42 lwarp-biblatex.sty 636
43 lwarp-bibunits.sty 636
44 lwarp-bigdelim.sty 637
45 lwarp-bigfoot.sty 638
46 lwarp-bigstrut.sty 639
47 lwarp-bitpattern.sty 639
48 lwarp-blowup.sty 640
49 lwarp-bm.sty 640
50 lwarp-booklet.sty 640
51 lwarp-bookmark.sty 641

52 lwarp-booktabs.sty 641
53 lwarp-bophook.sty 643
54 lwarp-bounddvi.sty 644
55 lwarp-boxedminipage2e.sty 644
56 lwarp-braket.sty 644
57 lwarp-breakurl.sty 645
58 lwarp-breqn.sty 645
59 lwarp-bsheaders.sty 647
60 lwarp-bxpapersize.sty 647
61 lwarp-bytefield.sty 647
62 lwarp-cancel.sty 648
63 lwarp-canoniclayout.sty 649
64 lwarp-caption.sty 649
65 lwarp-cases.sty 652
66 lwarp-changebar.sty 653
67 lwarp-changelayout.sty 653
68 lwarp-changepage.sty 654
69 lwarp-changes.sty 654
70 lwarp-chappg.sty 657
71 lwarp-chapterbib.sty 657
72 lwarp-chemfig.sty 657
73 lwarp-chemformula.sty 659
74 lwarp-chemgreek.sty 664
75 lwarp-chemmacros.sty 665
175 chemmacros 665
175.1 Changes to the user's document 665
175.2 Code 666
175.3 Loading modules 666

175.4 New environments . . . 666
175.5 Acid-base . . . 666
175.6 Charges . . . 668
175.7 Nomenclature . . . 669
175.8 Particles . . . 671
175.9 Phases. . . . 671
175.10 Mechanisms . . . 672
175.11 Newman . . . 674
175.12 Orbital . . . 674
175.13 Reactions . . . 675
175.14 Redox . . . 676
175.15 Scheme . . . 676
175.16 Spectroscopy . . . 677
175.17 Thermodynamics . . . 681

76 lwarp-chemnum.sty . . . 683
77 lwarp-chkfloat.sty . . . 684
78 lwarp-chngpage.sty . . . 684
79 lwarp-cite.sty . . . 684
80 lwarp-CJK.sty . . . 685
81 lwarp-CJKutf8.sty . . . 685
82 lwarp-clrdblpg.sty . . . 685
83 lwarp-cmdtrack.sty . . . 685
84 lwarp-color.sty . . . 686
85 lwarp-colortbl.sty . . . 686
86 lwarp-continue.sty . . . 688
87 lwarp-copyrightbox.sty . . . 688
88 lwarp-crop.sty . . . 689
89 lwarp-ctable.sty . . . 689

90 lwarp-cuted.sty 692
91 lwarp-cutwin.sty 692
92 lwarp-dblfloatfix.sty 693
93 lwarp-dblfnote.sty 693
94 lwarp-dcolumn.sty 693
95 lwarp-diagbox.sty 693
96 lwarp-dingbat.sty 695
97 lwarp-dprogress.sty 696
98 lwarp-draftcopy.sty 696
99 lwarp-draftfigure.sty 697
100 lwarp-draftwatermark.sty 697
101 lwarp-easy-todo.sty 697
102 lwarp-ebook.sty 698
103 lwarp-ed.sty 699
104 lwarp-ellipsis.sty 699
105 lwarp-embrac.sty 700
106 lwarp-emptypage.sty 700
107 lwarp-endfloat.sty 701
108 lwarp-endheads.sty 701
109 lwarp-endnotes.sty 702
110 lwarp-enumerate.sty 703
111 lwarp-enumitem.sty 703
112 lwarp-epigraph.sty 704
113 lwarp-epsfig.sty 705
114 lwarp-epstopdf.sty 706
115 lwarp-epstopdf-base.sty 706
116 lwarp-eqlist.sty 707

117 lwarp-eqparbox.sty 707
118 lwarp-errata.sty 708
119 lwarp-eso-pic.sty 710
120 lwarp-eurosym.sty 710
121 lwarp-everypage.sty 711
122 lwarp-everyshi.sty 711
123 lwarp-extramarks.sty 711
124 lwarp-fancybox.sty 712
125 lwarp-fancyhdr.sty 717
126 lwarp-fancyref.sty 718
127 lwarp-fancytabs.sty 719
128 lwarp-fancyvrb.sty 719
129 lwarp-figcaps.sty 725
130 lwarp-figsize.sty 726
131 lwarp-fitbox.sty 726
132 lwarp-fix2col.sty 727
133 lwarp-fixme.sty 727
134 lwarp-fixmetodonotes.sty 728
135 lwarp-flafter.sty 729
136 lwarp-flippdf.sty 729
137 lwarp-float.sty 729
138 lwarp-floatflt.sty 731
139 lwarp-floatpag.sty 732
140 lwarp-floatrow.sty 732
141 lwarp-fltrace.sty 737
142 lwarp-flushend.sty 737
143 lwarp-fnbreak.sty 738

144 lwarp-fncychap.sty 738
145 lwarp-fnlineno.sty 739
146 lwarp-fnpara.sty 739
147 lwarp-fnpos.sty 739
148 lwarp-fontawesome.sty 739
149 lwarp-fontawesome5.sty 741
150 lwarp-fontaxes.sty 741
151 lwarp-fontenc.sty 742
152 lwarp-footmisc.sty 742
153 lwarp-footnote.sty 743
154 lwarp-footnotebackref.sty 744
155 lwarp-footnotehyper.sty 745
156 lwarp-footnoterange.sty 745
157 lwarp-footnpag.sty 745
158 lwarp-foreign.sty 745
159 lwarp-forest.sty 746
160 lwarp-framed.sty 746
161 lwarp-ftcap.sty 749
162 lwarp-ftnright.sty 749
163 lwarp-fullminipage.sty 749
164 lwarp-fullpage.sty 749
165 lwarp-fullwidth.sty 749
166 lwarp-fwlw.sty 750
167 lwarp-gentombow.sty 750
168 lwarp-geometry.sty 750
169 lwarp-gloss.sty 751
170 lwarp-glossaries.sty 752

171 lwarp-gmeometric.sty 754

172 lwarp-graphics.sty 754

272 graphics 754

272.1 Graphics extensions 754

272.2 Length conversions and graphics options 755

272.3 Printing HTML styles 758

272.4 \includegraphics 758

272.5 Boxes 764

173 lwarp-graphicx.sty 767

174 lwarp-grffile.sty 767

175 lwarp-grid.sty 767

176 lwarp-grid-system.sty 767

177 lwarp-gridset.sty 768

178 lwarp-hang.sty 768

179 lwarp-hanging.sty 770

180 lwarp-hypbmsec.sty 771

181 lwarp-hypcap.sty 771

182 lwarp-hypdestopt.sty 771

183 lwarp-hypernat.sty 771

184 lwarp-hyperref.sty 772

185 lwarp-hyperxmp.sty 777

186 lwarp-hyphenat.sty 777

187 lwarp-idxlayout.sty 778

188 lwarp-ifoddpage.sty 778

189 lwarp-imakeidx.sty 779

190 lwarp-index.sty 783

191 lwarp-inputtrc.sty 784

192 lwarp-intopdf.sty 785

193 lwarp-karnaugh-map.sty 785
194 lwarp-keyfloat.sty 788
195 lwarp-layaureo.sty 794
196 lwarp-layout.sty 794
197 lwarp-layouts.sty 794
198 lwarp-leading.sty 797
199 lwarp-letterspace.sty 797
200 lwarp-lettrine.sty 797
201 lwarp-lineno.sty 798
202 lwarp-lips.sty 801
203 lwarp-listings.sty 801
204 lwarp-listliketab.sty 805
205 lwarp-lltjext.sty 805
206 lwarp-longtable.sty 806
207 lwarp-lscape.sty 809
208 lwarp-ltablex.sty 809
209 lwarp-ltcaption.sty 810
210 lwarp-ltxgrid.sty 810
211 lwarp-ltxtable.sty 811
212 lwarp-lua-check-hyphen.sty 811
213 lwarp-lua-visual-debug.sty 811
214 lwarp-luacolor.sty 811
215 lwarp-luatodonotes.sty 812
216 lwarp-lyluatex.sty 814
217 lwarp-magaz.sty 816
218 lwarp-makeidx.sty 816
219 lwarp-manyfoot.sty 817

220 lwarp-marginal.sty . . . 819
221 lwarp-marginfit.sty . . . 819
222 lwarp-marginfix.sty . . . 819
223 lwarp-marginnote.sty . . . 820
224 lwarp-marvosym.sty . . . 820
225 lwarp-mathtools.sty . . . 821
226 lwarp-mcaption.sty . . . 825
227 lwarp-mdframed.sty . . . 826
327 mdframed . . . 826
327.1 Limitations . . . 826
327.2 Package loading . . . 827
327.3 Patches . . . 827
327.4 Initial setup . . . 827
327.5 Color and length HTML conversion . . . 828
327.6 Environment encapsulation . . . 828
327.7 Mdframed environment . . . 829
327.8 Titles and subtitles . . . 830
327.9 New environments . . . 832
228 lwarp-media9.sty . . . 835
229 lwarp-memhfixc.sty . . . 837
230 lwarp-metalogo.sty . . . 837
231 lwarp-metalogox.sty . . . 838
232 lwarp-mhchem.sty . . . 838
233 lwarp-microtype.sty . . . 841
234 lwarp-midfloat.sty . . . 842
235 lwarp-midpage.sty . . . 842
236 lwarp-minibox.sty . . . 842
237 lwarp-minitoc.sty . . . 843

238 lwarp-morefloats.sty 843
239 lwarp-moreverb.sty 844
240 lwarp-movie15.sty 845
241 lwarp-mparhack.sty 846
242 lwarp-multicap.sty 847
243 lwarp-multicol.sty 847
244 lwarp-multicolrule.sty 848
245 lwarp-multimedia.sty 849
246 lwarp-multirow.sty 850
247 lwarp-multitoc.sty 853
248 lwarp-musicography.sty 853
249 lwarp-nameauth.sty 856
250 lwarp-nameref.sty 858
251 lwarp-natbib.sty 858
252 lwarp-nccfancyhdr.sty 859
253 lwarp-needspace.sty 860
254 lwarp-nextpage.sty 860
255 lwarp-nfssext-cfr.sty 860
256 lwarp-nicefrac.sty 869
257 lwarp-niceframe.sty 870
258 lwarp-nomencl.sty 870
259 lwarp-nonfloat.sty 871
260 lwarp-nonumonpart.sty 871
261 lwarp-nopageno.sty 871
262 lwarp-notes.sty 871
263 lwarp-notespages.sty 872
264 lwarp-nowidow.sty 872

265 lwarp-ntheorem.sty 873

365 ntheorem 873

365.1 Limitations 873

365.2 Options 874

365.3 Remembering the theorem style 874

365.4 HTML cross-referencing 877

365.5 \newtheoremstyle 877

365.6 Standard styles 878

365.7 Additional objects 879

365.8 Renewed standard configuration 880

365.9 amsthm option 881

365.10 Ending a theorem 883

365.11 \NoEndMark 883

365.12 List-of 883

365.13 Symbols 884

365.14 Cross-referencing 884

266 lwarp-octave.sty 884

267 lwarp-overpic.sty 886

268 lwarp-pagegrid.sty 886

269 lwarp-pagenote.sty 887

270 lwarp-pagesel.sty 887

271 lwarp-paralist.sty 887

272 lwarp-parallel.sty 888

273 lwarp-parcolumns.sty 889

274 lwarp-parnotes.sty 891

275 lwarp-parskip.sty 892

276 lwarp-pbox.sty 893

277 lwarp-pdfcol.sty 893

278 lwarp-pdfcolfoot.sty 894
279 lwarp-pdfcolmk.sty 894
280 lwarp-pdfcolparallel.sty 894
281 lwarp-pdfcolparcolumns.sty 895
282 lwarp-pdfcomment.sty 895
283 lwarp-pdfcrypt.sty 896
284 lwarp-pdflscape.sty 896
285 lwarp-pdfmarginpar.sty 896
286 lwarp-pdfpages.sty 896
287 lwarp-pdfprivacy.sty 899
288 lwarp-pdfrender.sty 899
289 lwarp-pdfsync.sty 899
290 lwarp-pdftricks.sty 900
291 lwarp-pdfx.sty 900
292 lwarp-perpage.sty 900
293 lwarp-pfnote.sty 902
294 lwarp-phfqit.sty 902
295 lwarp-physics.sty 902
296 lwarp-pifont.sty 908
297 lwarp-placeins.sty 909
298 lwarp-plarydshln.sty 909
299 lwarp-plext.sty 909
300 lwarp-plextarydshln.sty 910
301 lwarp-plextcolortbl.sty 911
302 lwarp-prelim2e.sty 911
303 lwarp-prettyref.sty 911
304 lwarp-preview.sty 911

305 lwarp-psfrag.sty 912
306 lwarp-psfragx.sty 912
307 lwarp-pst-eps.sty 913
308 lwarp-pstool.sty 913
309 lwarp-pstricks.sty 914
310 lwarp-pxatbegshi.sty 914
311 lwarp-pxeveryshi.sty 914
312 lwarp-pxftnright.sty 915
313 lwarp-pxjahyper.sty 915
314 lwarp-quotchap.sty 915
315 lwarp-quoting.sty 916
316 lwarp-ragged2e.sty 917
317 lwarp-realscripts.sty 917
318 lwarp-refcheck.sty 918
319 lwarp-register.sty 919
320 lwarp-relsize.sty 920
321 lwarp-repeatindex.sty 921
322 lwarp-resizegather.sty 922
323 lwarp-rmpage.sty 922
324 lwarp-romanbar.sty 922
325 lwarp-romanbarpagenumber.sty 922
326 lwarp-rotating.sty 923
327 lwarp-rotfloat.sty 923
328 lwarp-rviewport.sty 924
329 lwarp-savetrees.sty 924
330 lwarp-scalefnt.sty 925
331 lwarp-schemata.sty 925

332 lwarp-scrextend.sty 925
333 lwarp-scrhack.sty 928
334 lwarp-scrlayer.sty 928
335 lwarp-scrlayer-notecolumn.sty 930
336 lwarp-scrlayer-scrpage.sty 930
337 lwarp-scrpage2.sty 931
338 lwarp-section.sty 932
339 lwarp-sectionbreak.sty 933
340 lwarp-sectsty.sty 934
341 lwarp-semantic-markup.sty 934
342 lwarp-setspace.sty 935
343 lwarp-shadow.sty 936
344 lwarp-shapepar.sty 937
345 lwarp-showidx.sty 937
346 lwarp-showkeys.sty 937
347 lwarp-showtags.sty 938
348 lwarp-sidecap.sty 938
349 lwarp-sidenotes.sty 939
350 lwarp-SIunits.sty 940
351 lwarp-siunitx.sty 941
352 lwarp-slantsc.sty 951
353 lwarp-slashed.sty 952
354 lwarp-soul.sty 952
355 lwarp-soulpos.sty 954
356 lwarp-soulutf8.sty 954
357 lwarp-splitidx.sty 955
358 lwarp-srcltx.sty 956

359 lwarp-srctex.sty 957
360 lwarp-stabular.sty 957
361 lwarp-stackengine.sty 957
362 lwarp-steinmetz.sty 959
363 lwarp-stfloats.sty 960
364 lwarp-struktex.sty 960
365 lwarp-subcaption.sty 961
366 lwarp-subfig.sty 961
367 lwarp-subfigure.sty 966
368 lwarp-supertabular.sty 966
369 lwarp-svg.sty 968
370 lwarp-syntonly.sty 969
371 lwarp-tabfigures.sty 969
372 lwarp-tablefootnote.sty 969
373 lwarp-tabls.sty 969
374 lwarp-tabularx.sty 970
375 lwarp-tabulary.sty 970
376 lwarp-tascmac.sty 971
377 lwarp-textarea.sty 972
378 lwarp-textcomp.sty 973
379 lwarp-textfit.sty 976
380 lwarp-textpos.sty 977
381 lwarp-theorem.sty 977
382 lwarp-thinsp.sty 981
383 lwarp-threadcol.sty 981
384 lwarp-threeparttable.sty 982
385 lwarp-threeparttablex.sty 983

386 lwarp-thumb.sty 984
387 lwarp-thumbs.sty 984
388 lwarp-tikz.sty 984
389 lwarp-titleps.sty 986
390 lwarp-titleref.sty 989
391 lwarp-titlesec.sty 989
392 lwarp-titletoc.sty 991
393 lwarp-titling.sty 992
394 lwarp-tocbasic.sty 997
395 lwarp-tocbibind.sty 997
396 lwarp-tocdata.sty 999
397 lwarp-tocenter.sty 1000
398 lwarp-tocloft.sty 1001
399 lwarp-tocstyle.sty 1007
400 lwarp-todo.sty 1007
401 lwarp-todonotes.sty 1008
402 lwarp-topcapt.sty 1010
403 lwarp-tram.sty 1010
404 lwarp-transparent.sty 1010
405 lwarp-trimclip.sty 1011
406 lwarp-trivfloat.sty 1011
407 lwarp-truncate.sty 1013
408 lwarp-turnthepage.sty 1013
409 lwarp-twoup.sty 1013
410 lwarp-typearea.sty 1013
411 lwarp-typicons.sty 1014
412 lwarp-ulem.sty 1015

413 lwarp-umoline.sty 1016
414 lwarp-underscore.sty 1017
415 lwarp-unicode-math.sty 1017
416 lwarp-units.sty 1018
417 lwarp-unitsdef.sty 1019
418 lwarp-upref.sty 1020
419 lwarp-url.sty 1020
420 lwarp-uspace.sty 1020
421 lwarp-verse.sty 1020
422 lwarp-versonotes.sty 1022
423 lwarp-vertbars.sty 1023
424 lwarp-vmargin.sty 1023
425 lwarp-vowel.sty 1024
426 lwarp-vpe.sty 1024
427 lwarp-vwcol.sty 1024
428 lwarp-wallpaper.sty 1026
429 lwarp-watermark.sty 1027
430 lwarp-widetable.sty 1027
431 lwarp-widows-and-orphans.sty 1028
432 lwarp-witharrows.sty 1028
433 lwarp-wrapfig.sty 1029
434 lwarp-xbmks.sty 1031
435 lwarp-xcolor.sty 1031
535 xcolor 1031
535.1 Limitations 1031
535.2 xcolor definitions: location and timing 1032
535.3 Package loading 1034
535.4 Remembering and restoring original definitions 1034

535.5 \normalcolor . . . 1034
535.6 HTML color style . . . 1034
535.7 HTML border . . . 1035
535.8 High-level macros . . . 1036
535.9 Row colors . . . 1040

436 lwarp-xechangebar.sty . . . 1042
437 lwarp-xellipsis.sty . . . 1042
438 lwarp-xetexko-vertical.sty . . . 1043
439 lwarp-xfakebold.sty . . . 1043
440 lwarp-xfrac.sty . . . 1044
441 lwarp-xltabular.sty . . . 1046
442 lwarp-xltxtra.sty . . . 1046
443 lwarp-xmpincl.sty . . . 1047
444 lwarp-xpiano.sty . . . 1047
445 lwarp-xpinyin.sty . . . 1048
446 lwarp-xr.sty . . . 1048
447 lwarp-xr-hyper.sty . . . 1049
448 lwarp-xtab.sty . . . 1049
449 lwarp-xunicode.sty . . . 1051
450 lwarp-xurl.sty . . . 1052
451 lwarp-xy.sty . . . 1052
452 lwarp-zhlineskip.sty . . . 1053
453 lwarp-zwpagelayout.sty . . . 1053
454 lwarp-patch-komascript.sty . . . 1055
455 lwarp-patch-memoir.sty . . . 1057

555 patch-memoir . . . 1057

555.1 Packages . . . 1058
555.2 Preliminary setup . . . 1059

555.3 Page layout 1059
555.4 Text and fonts 1061
555.5 Titles 1062
555.6 Abstracts 1062
555.7 Docment divisions 1062
555.8 Pagination and headers 1066
555.9 Paragraphs and lists 1067
555.10 Contents lists 1067
555.11 Floats and captions 1071
555.12 Page notes 1074
555.13 Decorative text 1076
555.14 Poetry 1076
555.15 Boxes, verbatims and files 1077
555.16 Cross referencing 1077
555.17 Back matter 1078
555.18 Miscellaneous 1079
555.19 ccaption emulation 1080
555.20 Final patchwork 1082

456 lwarp-common-multimedia.sty 1082

Change History 1088

557 Chg Hist 1088

Index of Objects 1119

General Index 1140

Troubleshooting Index 1144

Index of Indexes 1150

List of Figures

1 tutorial.tex listing 82

List of Tables

1 Typesetting conventions 64
2 LaTeX lwarp package — Supported features 65
3 Required software programs 75
4 Configuration files created by print version 84
5 lwarp package options 102
6 HTML settings 107
7 Literal character macros 169
8 Section HTML headings for word-processor conversion 177
9 Section depths and HTML headings 193
10 Tabular baseline 428
11 Tabular HTML column conversions 429
12 Cross-referencing data structures 472
13 Float data structures 483
14 amsthm package — CSS styling of theorems and proofs 612
15 Ntheorem package — CSS styling of theorems and proofs 873
16 Theorem package — CSS styling of theorems and proofs 978

1 Updates

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

For a detailed list of the most recent changes, see the end of the Change History on page 1118.

v0.79: MathJax, nested `tabular`.

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

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

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

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

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

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

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

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

v0.76: MathJax, updates for LaTeX 2019/10 release.

docs
- Docs: Expanded documentation regarding the use of multiple projects in the same directory. See section 5.17.

MathJax
- MathJax: Updated to v2.7.6.

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

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

v0.75: keyfloat, wrapfig

- `\minipage`: Fix for `\linewidth`.

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

v0.74: Docs, SVG math, *lwarpmk*, HTML `alt` and `title` text, lyluatex

docs
- Added to the tutorial the section What next?. See section 5.19.
- Added documentation about localization options. See section 7.1.
- Added documentation about accessibility options. See section 7.2.

HTML `alt` text
- Renamed and updated HTML `alt` text macros:

⚠ changed names

Old	New
(hard coded as “image”)	`\ImageAltText`
`\mathimagename`	`\MathImageAltText`
`\pacakgediagramname`	`\PackageDiagramAltText`

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

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

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

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

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

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

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

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

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

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

⚠ images

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

  Enter ⇒ **`lwarpmk cleanlimages`**

  and

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

packages

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

v0.71: Error handling, multimedia, `tabular`.

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

packages

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

multimedia

- Added media9, movie15, multimedia.

v0.70: Error handling, MathJax, mathtools.

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

packages

- textcomp, xunicode: Fix for `\textinterrobang`.
- mhchem: Works with MathJax. See section 332.
- changes: Updated to v3.1.2.
- Added autonum, changelayout, inputtrc, mathtools, metalogox.

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

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

packages

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

v0.68: Error handling, tabulars, footnotes.

lwarpmk

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

packages

- longtable: Improved flexibility for `\endhead`, etc. Improved error reporting if `\endhead`, etc. incorrect for lwarp.
- threeparttable: Fix for caption type.
- hyperref: Fix for options with braces.
- morefloats: Fix to be loaded early for print output.
- listings: Updated for v1.7.
- Added bigfoot, fnpara, footnotebackref, manyfoot, tablefootnote, threeparttablex.
- Added layouts, niceframe, perpage, showtags.
- Prevented alg, algorithmic, pdfcprot, fncylab.

v0.67: Filename generation, symbol fonts.

docs

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

filenames

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

packages

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

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

⚠ Reset the configuration

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

`lateximage`

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

⚠ existing projects

- To reuse existing `lateximage` directories, add lwarp options

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

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

filenames

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

⚠ Possible filename changes

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

Windows

- Fix for **`lwarpmk cleanlimages`** with Windows.

floats

- Fixes for floats in the home page.

lists, table notes

- Improved CSS for definition lists, table notes.

`tabular`

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

indexing

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

`minipage`

- `\minipagefullwidth`: Fix for global changes.
- Added `\UseMinipageWidths` and `\IgnoreMinipageWidths`. See section 8.3.3.

colors

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

HTML

- Improved HTML output formatting.

docs

- Added discussion regarding invalid HTML. See section 8.1.1.
- Added discussion regarding math in section names, `\imagegraphics` `scale` option. See section 6.
- Added discussion regarding international languages in section names. See section 8.14.

packages

- caption: Fix for options clash.
- xr, xr-hyper: Now compatible.
- subcaption: Improved horizontal spacing.
- multicol: Fix for minipage inside `multicols`.
- multicolrule: Updated for v1.2.
- tocbasic: Minor update.
- acronym: Fix for acronym in float caption.
- kotexutf: Patch with *pdflatex* and new lwarp labels.
- extramarks, fancyhdr: Updated for v3.10.
- memoir: Added docs regarding version numbers. See section 8.13.

- zref: No longer required.
- Added ar, ed, indentfirst, nameauth, truncate.
- Verified to work as-is with changelog.
- Prevented colortab, epsf, hyper, picinpar, picins, sistyle, ucs.

v0.65: css layout, `alt` tags, Japanese.

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

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

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

fixes
- Fix for `tabular*`.
- Fix for tabular border colors.
- Fixes `\quad`, `\enskip`, and figure captions to pass validation.

Japanese
- Added ltj* classes, bounddvi, gentombow, lltjext, plarydshln, plext, plextarydshln, plextcolortbl, pxatbegshi, pxeveryshi, pxftnright, pxjahyper, tascmac.
- Verified to work with plarray, plautopatch, plextarray, plextdelarray, pxgentombow, plsiunitx, pxpdfpages, pxpgfrcs, pxpgfmark.

packages
- Added support for fontspec `\textsi` and `\sishape`.
- Added multicol's `\docolaction`.
- Added embrac, footnoterange, multicolrule, versonotes.

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

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

Chinese
- Fix for biblatex with CTEX and other classes.

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

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

v0.63: mdframed, Chinese, Japanese, Korean

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

fixes
- Fix: `\LinkHome` for print output.

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

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

Chinese
- Fixes for xeCJK.
- Added xpinyin, zhlineskip.

- Verified to work with cjkpunct, upzhkinsoku, zhspacing.

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

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

v0.62: MiKTeX docs, HTML title, CTeX, xeCJK, bitpattern.

docs
- Docs: Setting a UTF-8 locale. See section 9.8.

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 7.6.

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

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

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

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

docs
- Split index into multiple indexes.
- Improved documentation regarding font selection. See section 7.4.
- Added documentation regarding debugging options. See section 35.
- Added documentation regarding HTML entities inside program listings. See section 8.2.1.

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

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

Old	New
`\StartDynamicMath`	`\inlinemathother`
`\StopDynamicMath`	`\inlinemathnormal`

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

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

v0.60: Fixes for longtable, listings.

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

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

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

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

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

- Added
  ```
  lwarpmk epstopdf <list-of-EPS-files>
  ```
  to quickly convert a document's EPS images to PDF or SVG. See section 8.8.

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

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

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

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

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

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

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

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

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

glossaries
- *lwarpmk*: Now uses *makeglossaries* for glossary generation, allowing the processing of multiple glossaries at once.
- Added lwarp option `GlossaryCmd` to specify the shell command used by **`lwarpmk printglossary`** and **`lwarpmk htmlglossary`**. Defaults to **`makeglossaries`**.

index and glossary
- Docs: Extra indexing options. See section 8.6.14.
- Added support for *makeindex*. (Previously supported only *xindy*.) Also added indexing packages listed below.
- Added lwarp options `PrintIndexCmd`, `HTMLIndexCmd`, and `LatexmkIndexCmd` to specify shell commands used by **`lwarpmk printindex`**, **`lwarpmk htmlindex`**, and *latexmk*. May be preset with the `makeindex` or `xindy` lwarp options. See section 7.5.
- Added lwarp options `makeindex` and `xindy` to set `PrintIndexCmd`, `HTMLIndexCmd`, and `LatexmkIndexCmd` to sensible values for a typical single index. See section 7.5.

- Added lwarp option `makeindexStyle` to tell *lwarpmk* to use a custom style instead of `lwarp.ist`. See section 8.6.20.
- Fix for index entries with `\see`, `\seealso`, `\emph`, `\textbf`, etc.

misc. fixes
- Replaced each `\csuse` with `\@nameuse` for improved error detection.
- Additional internal print/HTML macro selection improvements.
- Fix: `\printindex` finishes pending `\index` writes first.

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 8.7.7.

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 8.7.8.

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

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

  See section 7.6.
- Fix: Improved print/HTML macro selection.

misc. fixes
- Fix: `\href` text catcodes.
- Fix: `\subref` text.
- Fixes: Colored `\rule` and `\boxframe`.

packages
- float, rotfloat: Adds support for float styles `ruled` and `boxed`.
- float: Fix: Do not create `\l@<type>` until `\listof` is used.
- marginnote: Fix: Long optional argument.
- ellipsis: Adds `\midwordellipsis`.
- breakurl: Fix for text catcodes.
- Added algorithm2e, register, ltablex, xltabular, xellipsis, trimclip, errata, vowel, xpiano.
- Prevents glossary.
- Tested to work as-is with gauss, phonrule, piano, SIunits, tikzcodeblocks.

v0.56: Shell escape, tabular packages.

lwarpmk
- Added
  ```
  lwarpmk pdftosvg <list-of-PDF-files>
  ```
  to quickly convert a document's PDF images to SVG, for use with HTML. See section 8.8.
- Added support for `--shell-escape`. See section 7.3.

`tabular`
- Added support for array `w` and `W` columns.
- Fix: `\multicolumn` parameter handling.
- Added support for double `\hlines`, `\midrules`, and vertical rules.
- Added support for arydshln dashed lines with HTML `tabular`, but reverts to plain rules for `lateximage` and SVG math `array`.

misc. fixes
- Fix: `\thinspace`.
- Fix: paralist compact environments.

packages
- Added parnotes, quoting, lua-check-hyphen, tocenter, underscore.
- Added bibunits.
- Tested to work as-is with babelbib, bodegraph, fast-diagram, nicematrix, structmech.

v0.55: Various fixes.

misc fixes
- Fix: Extraneous space in file links, which also prevented *Calibre* EPUB conversions.
- Fix: Float optional argument regression.
- Fix: `\ForceHTMLTOC` with `\phantomsection`.
- Fix: Overfull boxes in `lateximages`.
- Fix: QED symbols in `lateximage`.

packages
- koma-script: Fix: Figure with `\centering`, etc.
- Added clrdblpg.

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

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

lwarpmk
- `lwarpmk limages` checks for the presence of the HTML version of the document and valid image references before attempting to create the `lateximages`.
- *lwarpmk*: Improved error message if configuration file does not exist.

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

polyglossia
- Added documentation regarding polyglossia. See section 8.15.4.

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

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

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

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

floats with `\centering`, etc.

- Floats now honor `\centering`, `\raggedright`, `\raggedleft`, and their ragged2e equivalents, when placed directly after:

```
\begin{floattype}
\centering
```

misc. fixes

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

⚠

- Removed `\citetitle`, adjusted `\attribution`.

packages

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

v0.53: Improved image checks.

lwarpmk

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

v0.52: Improved footnotes, SVG math.

documentation

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

SVG math

- Fix: Limit the number of background tasks when generating `lateximages`.
- Added user-adjustable SVG math font scaling. See section 80.3.
- Added warnings if `lwarp_baseline_marker.png` is not present, or if graphicx or graphics is not loaded.
- Improved `\ensuremath` hashing expansion.

- Fix: `equation*` with `split`.
- `tabbing` now works inside a `lateximage`. Use for math in `tabbing`.

MathJax
- Fix: MathJax script was not executing in some conditions.
- Added `\CustomizeMathJax` to add custom functions. See section 8.7.

footnotes
- Fix: Footnote numbering when using `HTMLDebugComments`.
- Fix: Footnote paragraph tags.
- Fix: `FootnoteDepth` defaults to `\subsubsection`.

misc. fixes
- Fix: `\kill` in a `lateximage`.
- Fix: `\FileDepth`, misc. others, when input encoding is not `utf8`.
- Fix: `\texorpdfstring` in a section name.

packages
- hyperref emulation: Fix for #, %, &, ~, _ characters in URLs.
- fancybox, fancyvrb: Initial support for `\VerbatimFootnotes`.
- nicefrac: Added with fix for `\ensuremath`.
- graphicx: Fix for option defaults. Added v1.1a/b options.
- endfloat: Updated for v2.6.
- url: Fixes for active characters.

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 8.6.1.
- Docs: tikz with matrices. See section 8.8.1.

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 8.7.9.
- 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 6.

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 8.13.

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: Lateximages 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 7.6.

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. (See page 2.)

word-processor conversion
- Improved assitance for word-processor conversions when boolean `FormatWP` is set `true`. See section 11.

⚠ 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 mini-pages, 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 11.

`tabbing`
- Add: `tabbing` environment.

overpic
- Add: overpic package. See section 367.

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 `\textinterrobangdown`.
- 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 8.10.1.
- 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 346.2 on page 852.
- 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 8.8.

- grffile is now directly supported instead of emulated.

bigdelim
- Fix for bigdelim, and improved documentation. See section 144.

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 66.8. authblk is added, and should be loaded before titling. See section 66.

⚠ \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 346.2.

- Tabular special cases:
  - Added \TabularMacro to mark custom macros inside tabular data cells, avoiding row corruption. See section 8.10.1.
  - Added \ResumeTabular for use when a tabular environment is defined inside another environment. See section 8.10.1.

⚠ macros inside tabular
⚠ tabular defined inside another environment

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 8.6.2.

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 8.6.
- 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 8)
- 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 8.3.3.

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 8.3.5. `\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 8.6.21.
- 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.

2 Introduction

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

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

---

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

As a package running directly in LaTeX, lwarp has some advantages over other methods of HTML conversion. TeX itself is still used, allowing a wider range of TeX trickery to be understood. Lua expressions are still available with LuaTeX. Entire categories of LaTeX packages work as-is when used with lwarp: definitions, file handling, utilities, internal data structures and calculations, specialized math-mode typesetting for various fields of science and engineering, and anything generating plain-text output. Blocks of PDF output may be automatically converted to SVG images while using the same font and spacing as the original print document, directly supporting 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%.

2.1 Typesetting conventions

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

Table 1: Typesetting conventions

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

2.2 Supported packages and features

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

Package names are colored according to their support level:

name: Supported as-is.

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

name: Emulated for HTML output.

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

MJ: Emulated for MathJax, subject to limitations.

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.

lwarp Supported Functions — continued

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

lwarp Supported Functions — continued

Category	Status
Footnotes:	Adds `FootnoteDepth` to print footnotes at section breaks. bigfoot, dblfnote, endheads, endnotes, fixfoot, fnbreak, fnpara, fnpos, footmisc, footnote, footnotebackref, footnoterange, footnpag, manyfoot, marginnote, nccfoots, pagenote, parnotes, pdfcolfoot, pfnote, sepfootnotes, sidenotes, tablefootnote.
Math:	Converted to SVG images with HTML `<alt>` tags containing the LaTeX source for the math expression. MathJax supported as an alternative. amsmath[MJ]: $\mathcal{AMS}$ environments are supported. User-defined macros are available during converson, due to native LaTeX processing.
Theorems:	Native LaTeX theorems, amsthm, ntheorem, theorem.
Additional math:	Math fonts via SVG images, amscd, autonum, backnaur, bm[MJ], braket[MJ], breqn, cases, delarray, guass, jkmath, mathspec, mathtools[MJ], nicematrix, pb-diagram, resizegather, unicode-math[MJ], witharrows[MJ], xfakebold[MJ], xy. Many others work as-is.
Display math with `\displaymathother`:	Complicated math objects in display math, such as tikz-cd, etc.
Units and fractions:	nicefrac[MJ], SIunits, siunitx[MJ], units[MJ], unitsdef, xfrac.
Floats:	Appear where declared. capt-of, caption, cutwin, dblfloatfix, endfloat, fix2col, flafter, float, floatflt, floatrow, fltrace, ftcap, hypcap, keyfloat, morefloats, multicap, newfloat, nonfloat, placeins, rotfloat, stfloats, subcaption, subfig, subfigure, subfloat, topcapt, trivfloat, wrapfig.
Tabular:	`tabular` environment, array[MJ], arydshln[MJ], bigdelim[MJ], bigstrut[MJ], booktabs[MJ], colortbl, ctable, diagbox, longtable, ltablex, ltxtable, multirow[MJ], supertabular, tabularx, tabulary, threeparttable, threeparttablex, widetable, xltabular, xtab.

lwarp Supported Functions — continued

Category	Status
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, curves, datatool, eepic, epsfig, epstopdf, figsize, fitbox, grffile, media9, movie15, multimedia, overpic, psfrag, psfragx, pst-eps, pstool, pstricks, rviewport, svg, svg-extract, 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, eqlist, hang, listliketab, paralist.
Environments:	Standard LaTeX environments.
`minipage`, `\parbox`:	Some HTML5-imposed limitations. Nested minipages are supported. eqparbox, minibox, pbox, shapepar.
Quotations:	copyrightbox, csquotes, epigraph, quoting, verse.
Verbatim:	fancyvrb, moreverb, shortvrb, verbatim.
Frames:	boxedminipage2e, fancybox, framed, mdframed, niceframe, 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, normalcolor, pdfcol, pdfcolmk, pdfrender, realscripts, relsize, scalefnt, soul, soulpos, soulutf8, stackengine, textfit, thinsp, trimclip, truncate, ulem, umoline, underscore, uspace, xellipsis.
Acronyms:	acro, acronym.
Ordinals:	engord, fmtcount, nth.

lwarp Supported Functions — continued

Category	Status
Text ligatures:	Ligatures for symbols are supported. Ligatures for f, q, t are intentionally turned off because many simpler browsers do not display them correctly. Modern full-featured browsers re-create these ligatures on-the-fly.
Horizontal space:	HTML output for thin-unbreakable, unbreakable, `\enskip`, `\quad`, `\qquad`, `\hspace`.
Rules:	`\rule` with width, height, raise, text color.
HTML reserved characters:	`\&`, `\textless`, and `\textgreater` are converted to HTML entities.
Fonts:	Used as-is. Appear in SVG math expressions or embedded image environments. fontaxes, nfssext-cfr, slantsc, tabfigures. Tested to work as-is: Special font macros in cfr-lm and others which use nfssext-cfr.
Symbols:	Native LaTeX diacriticals, academicons, bbding, chemgreek, dingbat, euro, eurosym, fontawesome, fontawesome5, gensymb, marvosym, metalogo, metalogox, pifont, textalpha, textcomp, textgreek, typicons, xunicode.
Files:	attachfile, attachfile2, hyperxmp, inputtrc, intopdf, pdfpages, pdfx, xmpincl.
Science and engineering:	algorithm2e, algorithmicx, ar[MJ], askmaps, axodraw2, bitpattern, blochsphere, bodegraph, bohr, bytefield, chemfig, chemformula, chemgreek, chemmacros, chemnum, circuitikz, elements, engtlc, fast-diagram, hepnicenames, heppennames, karnaughmap, karnaugh-map, listings, linop, mhchem[MJ], pgfgantt, phfqit, physics[MJ], qcircuit, register, simpler-wick, slashed[MJ], steinmetz[MJ], structmech, struktex. tikz-karnaugh, tikzcodeblocks
Arts and humanities:	foreign, forest, lyluatex, musicography, nameauth, octave, phonrule, piano, schemata, semantic-markup, tikz-dependency, vowel, xpiano
Admonitions:	awesomebox, notes.
Editorial:	changebar, changelog, changes, easy-todo, easyReview, ed, errata, fixme, fixmetodonotes, pdfcomment, pdfmarginpar, todo, todonotes, tram, xechangebar.

lwarp Supported Functions — continued

Category	Status
Accessibility:	accessibility, accsupp, axessibility.
Package handling:	catoptions.
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.

3 Alternatives

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

3.1 internet class

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

3.2 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.

3.3 Translators

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

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/

3.4 AsciiDoc and Asciidoctor

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

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

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

3.4.1 Asciidoctor-LaTeX

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

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

3.5 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.)

3.6 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*.

3.7 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* ™.

3.8 Comparisons

AsciiDoc, Pandoc, and various other markup languages typically have a syntax which tries to be natural and human-readable, but the use of advanced features tends to require many combinations of special characters, resulting in a complicated mess of

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

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

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

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

4 Installation

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

Detailed installation instructions follow.

Table 3: Required software programs

Provided by your LaTeX distribution:

From TeXLive: http://tug.org/texlive/.

LaTeX: *pdflatex*, *xelatex*, or *lualatex*.

The lwarp package: This package.

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

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

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

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

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

Poppler PDF utilities:

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

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

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

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

From Poppler: poppler.freedesktop.org.

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

Enter ⇒ **`brew install poppler`**

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

***Perl*:**

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

strawberryperl.com (recommended), perl.org

Automatically downloaded from the internet as required:

MathJax: Optionally used to display math. From: mathjax.org

4.1 Installing the lwarp package

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

Pre-installed: Try entering into a command line:

Enter ⇒ **`kpsewhich lwarp.sty`**

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

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

Enter ⇒ **`tlmgr install lwarp`**

MiKTEX:

1. For newer versions of MiKTEX, install or update lwarp using the *MiKTeX Console* program.
2. For older versions of MiKTEX, to install lwarp the first time, use the *MiKTeX Package Manager (Admin)*. To update lwarp, use *MiKTeX Update (Admin)*.
3. Either way, also update the package miktex-misc, which will install and update the *lwarpmk* executable.

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

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

1. See http://ctan.org/pkg/lwarp for the lwarp package.
2. Download the TDS archive: `lwarp.tds.zip`
3. Find the TEX `local` directory:

   TEX Live:

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

   MiKTEX:

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

   This should be something like:

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

4. Unpack the archive in the TDS local directory.
5. Renew the cache:

   Enter ⇒ **`mktexlsr`**

   — or —

   Enter ⇒ **`texhash`**

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

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

1. See http://ctan.org/pkg/lwarp for the lwarp package.
2. Download the zip archive `lwarp.zip` into your own `lwarp` directory.
3. Unpack `lwarp.zip`.
4. Locate the contents `lwarp.dtx` and `lwarp.ins`
5. Create the `.sty` files:

   Enter ⇒ **`pdflatex lwarp.ins`**
6. Create the documentation:

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

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

   `<texlocal>/tex/latex/local/lwarp`
8. Copy `lwarp_baseline_marker.png` and `lwarp_baseline_marker.eps` to the same place as the `.sty` files.
9. Copy the documentation `lwarp.pdf` to a `source` directory in the local tree, such as:

   `<texlocal>/doc/local/lwarp`
10. Renew the cache:

    Enter ⇒ **`mktexlsr`**

    — or —

    Enter ⇒ **`texhash`**

    Or, for WINDOWS MiKTEX, start the program called *MiKTeX Settings (Admin)* and click on the button called **Refresh FNDB**.
11. See section 4.2.1 to generate your local copy of *lwarpmk*.
12. Once the local version of `lwarpmk.lua` is installed, it may be made available system-wide as per section 4.2.

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

Just testing!

4.2 Installing the *lwarpmk* utility

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

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

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

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

   Otherwise, continue with the following:

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

   Enter ⇒ **kpsewhich lwarpmk.lua**

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

4. Create *lwarpmk*:

   Unix: Create a symbolic link and make it executable:

   (a) Locate the TEX Live binaries:

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

   This will be something like:

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

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

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

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

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

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

   (c) Make the link executable:

   Enter ⇒ **chmod 0755 lwarpmk**

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

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

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

   WINDOWS MiKTEX: Create a new `lwarpmk.bat` file:

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

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

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

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

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

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

4.2.1 Using a local copy of *lwarpmk*

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

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

   ```
   \usepackage[lwarpmk]{lwarp}
   ```

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

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

   Unix:

   (a) Make `lwarpmk.lua` executable:
   Enter ⇒ **`chmod 0755 lwarpmk.lua`**

   (b) Compile documents with
   Enter ⇒ **`./lwarpmk.lua html`**
   Enter ⇒ **`./lwarpmk.lua print`**
   etc.

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

   WINDOWS:

   Compile documents with either of the following, depending on which command shell is being used:
   Enter ⇒ **`texlua lwarpmk.lua html`**
   Enter ⇒ **`texlua lwarpmk.lua print`**
   etc.

   Or:
   Enter ⇒ **`lwarpmk html`**
   Enter ⇒ **`lwarpmk print`**
   etc.

4.3 Installing additional utilities

To test for the existence of the additional utilities:

Enter the following in a command line. If each programs' version is displayed, then that utility is already installed. See table 3 on page 75.

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 75.
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.

5 Tutorial

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

Need help?

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

5.1 Starting a new project

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

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.

5.2 Compiling the print version with *lwarpmk*

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

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

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 8.6.12.

[7] Also see section 8.6.15 for index options.

5.3 Compiling the HTML version with *lwarpmk*

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

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 5.4 and 5.5.

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 8.6.12.

[11] Also see section 8.6.15 for index options.

5.4 Generating the SVG images

math as SVG images

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

`picture` and Ti*k*z

lwarp uses the same mechanism for `picture` and Ti*k*z environments.

1. Create the SVG images:

   Enter ⇒ **`lwarpmk limages`**

   Enter ⇒ **`lwarpmk html`**

2. Move to the tutorial's HTML math page and reload the document in the browser.
3. The math images are displayed using the same font and formatting as the printed version.
4. Copy/paste a math expression into a text editor to see the LaTeX source.

⚠ adding/removing

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

recompile first

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

⚠ HTML instead of images

If HTML appears where an SVG image should be, recompile the document one more time to get the page numbers back in sync, then remake the images one more time.

⚠ page counter

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

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

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

⚠ Lots of files!

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

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

5.5 Using MathJax for math

math with MathJax

Math may also be represented using the MathJax JavaScript project.

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

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

2. Recompile

   Enter ⇒ **`lwarpmk html`**

3. Reload the math page.

⚠ MathJax requirements

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

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

5.6 Changing the css style

For a formal css style, add to the preamble:

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

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

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

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

5.7 Customizing the HTML output

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

See section 7.6 for more information.

5.8 Using *latexmk*

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

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

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

2. Recompile the printed version of the document.

   Enter ⇒ **`lwarpmk print`**

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

3. Recompile the document.

   Enter ⇒ **`lwarpmk print`**

   and/or

   Enter ⇒ **`lwarpmk html`**

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

forced single-pass recompile

Enter ⇒ **`lwarpmk print1`**

and/or

Enter ⇒ **`lwarpmk html1`**

5.9 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`**

5.10 Using DVI LaTeX

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

To convert EPS files to PDF:

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

To convert PDF files to SVG:

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

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

5.11 Using a glossary

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

5.11.1 gloss package

See section 8.6.11.

5.11.2 glossaries package

To process the glossary for the print version:

Enter ⇒ **`lwarpmk printglossary`**

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

To process the glossary for the HTML version:

Enter ⇒ **`lwarpmk htmlglossary`**

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

Enter ⇒ **`lwarpmk html1`**

Enter ⇒ **`lwarpmk html`**

Enter ⇒ **`lwarpmk print1`**

Enter ⇒ **`lwarpmk print`**

See section 8.6.12 to set options for processing glossaries.

5.12 Cleaning auxiliary files

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

Enter ⇒ **`lwarpmk clean`**

5.13 Cleaning auxiliary and output files

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

Enter ⇒ **`lwarpmk cleanall`**

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

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

Enter ⇒ **`lwarpmk cleanlimages`**

5.15 Converting PDF or EPS images to SVG

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

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

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

Be sure to always provide SVG files for HTML output.

5.16 Creating HTML from an incomplete compile

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

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

5.17 Processing multiple projects in the same directory

⚠ xr, xr-hyper, xcite

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

To create each project:

Enter ⇒ **pdflatex project_a**

Enter ⇒ **pdflatex project_b**

Each project is given its own configuration file:

`project_a.lwarpmkconf, project_b.lwarpmkconf`

To compile each project with `lwarkmk`:

Enter ⇒ **lwarpmk print -p project_a**

Enter ⇒ **lwarpmk print -p project_b**

Enter ⇒ **lwarpmk html -p project_a**

Enter ⇒ **lwarpmk html -p project_b**

To generate each project's images:

Enter ⇒ **lwarpmk limages -p project_a**

Enter ⇒ **lwarpmk limages -p project_b**

To clean each project's images:

Enter ⇒ **lwarpmk cleanlimages -p project_a**

Enter ⇒ **lwarpmk cleanlimages -p project_b**

To clean each project's auxiliary files:

Enter ⇒ **lwarpmk cleanall -p project_a**

Enter ⇒ **lwarpmk cleanall -p project_b**

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

Enter ⇒ **bibtex project_a_html**

5.18 Using the *make* utility

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

```
lwarpmk pdftohtml [-p project]
```

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

5.19 What next?

How do I do something? See the General Index.

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

Package options: See section 28, Package options.

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

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

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

Shell escape: See section 7.3, Shell escape.

CSS customization: See section 7.7, Customizing the CSS.

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

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

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

EPUB conversion: See section 10, EPUB conversion.

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

6 Converting an existing document

To convert an existing document for use with lwarp:

1. Arrange the document in the following order:
   (a) Declare the `\documentclass`.
   (b) Load text fonts.
   (c) Load inputenc or inputenx, fontenc, or fontspec.
   (d) Load lwarp.
   (e) Load remaining packages.
2. Modify the document:

⚠ math in section names

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

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

⚠ scale

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

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

   to:

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

⚠ tabular

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

⚠ package options

   (d) If using braces in package options, such as with caption, see section 8.1.
   (e) Possible option clashes with memoir. See section 8.13.

⚠ indexes

   (f) If using indexes, see section 8.6.15.
   (g) If using many indexes, glossaries, `.aux` files, etc., see section 8.6.15 regarding morewrites. If morewrites is already used, be sure to add the setup with `allocate=10`.
   (h) Other changes as per Special cases and limitations, section 8.
3. Convert any PDF images to SVG. See section 8.8.
4. Manually compile the print version with *latex*, *pdflatex*, *lualatex*, or *xelatex*.
5. `lwarpmk print` to finish the print version.
6. `lwarpmk html` to create the HTML version.
7. `lwarpmk limages` to create the SVG images of any SVG math, `lateximage`, Ti*k*z, etc.

Need help?

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

7 Additional details

7.1 Localization

Regional localization is supported by lwarp via the following package options and macros:

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

`\abstractname`: This macro is honored by lwarp.

`\linkhomename`: Displayed by the link to the homepage.

`\sidetocname`: Displayed at the head of the sideTOC.

HTML settings: See table 6 and section 7.6 for details.

`\HTMLLanguage`: The language to declare for each web page.

`\ImageAltText`, `\MathImageAltText`, `\PackageDiagramAltText`, `\AltTextOpen`, `\AltTextClose`: The defaults used for HTML `alt` text for images. See section 7.2.

`\CSSFilename`: The name of the CSS file to use.

`\MathJaxFilename`: The name of the MathJax script to use.

Package options:

`ImagesName` and `ImagesDirectory`: These options control the filenames used by lwarp when it automatically generates images. See table 5 and section 7.5.

`xindyStyle`, `xindyLanguage`, `xindyCodepage`: When using *xindy*, these options may be set according to local use. See section 8.6.21.

`pdftotextEnc`: To adjust the encoding of *pdftotext*.

7.2 Accessibility

lwarp provides several methods for improving access to the document using tools such as text-only browsers, copy/paste, text-to-speech readers, or Braille readers. HTML provides the `alt` text attribute for images, which is used by lwarp as describe below. HTML also provides the `title` attribute, which usually generates a pop-up text. lwarp can add this to a reference or hyperlink. MathJax also has provisions for improved accessibility as well.

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

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

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

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

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

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

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

custom `alt` text

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

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

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

references and links

`\ThisAltText` may also be used to assign an HTML `title` to the next reference or hyperlink.

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

See section 7.6.

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

7.3 Shell escape

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*.

7.4 Font and UTF-8 support

⚠ type 3 bitmapped fonts

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

vector fonts
Computer Modern
⚠ *pdflatex*
⚠ DVI *latex*

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.

Pkg cm-super

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*:

     Pkg lmodern
     Pkg fontenc
     Pkg inputenc
     Pkg inputenx
     Pkg newunicodechar
     File glyphtounicode.tex

     (a) `\usepackage{lmodern}`, or other font-related packages
     (b) `\usepackage[T1]{fontenc}`
     (c) `\usepackage[utf8]{inputenc}`, or `latin1`, etc. Or use inputenx.
     (d) `\usepackage{newunicodechar}` along with related definitions.
     (e) To assist with the PDF-HTML conversion:
        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 7.5) 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 9.8.

7.4.1 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.

7.5 lwarp package loading and options

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

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

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

```
\lwarpsetup{⟨key=value, . . . ⟩}
```

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 8.7.

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 7.6.

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 7.6.

Opt HTMLFilename
Default: {}

HTMLFilename: See section 7.6.

Opt ImagesName
Default: image-

ImagesName: The prefix for the images automatically generated by lwarp for objects such as SVG math and `latexmages`.

Opt ImagesDirectory
Default: \jobname-images

ImagesDirectory: The directory for the images automatically generated by lwarp for objects such as SVG math and `latexmages`. By default, these images will appear in a directory named `<jobname>-images`, and the images will be named and numbered `image-<nn>`.

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*

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.
ImagesName	A prefix for the filenames of generated images.
ImagesDirectory	The directory used to hold generated images.
PrintLatexCmd	The shell commands for **lwarpmk print**.
HTMLLatexCmd	The shell commands for **lwarpmk html**.
For indexing (section 8.6.15) and glossaries (section 8.6.12):	
makeindex	Use *makeindex* to generate indices.
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.

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 8.6.15.

Examples:

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

automatic setting

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

⚠ xindy

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

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

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 8.6.15.

⚠ filenames

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

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

automatic setting

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

⚠ xindy

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

As with PrintIndexCmd, to generate complicated indexes it may be 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 8.6.15.

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

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

automatic setting

The use of the makeindex or xindy options sets LatexmkIndexCmd to 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 8.6.12.

Opt OSWindows

OSWindows: lwarp attempts to automatically sense WINDOWS, but it may be forced with this option. See section 7.8.

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.

7.6 Customizing the HTML output

⚠ Placement!

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

⚠ Changes!

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

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

   Enter ⇒ **`lwarpmk 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 7.6.1 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 7.6.1 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. Redefine with `\renewcommand`.

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.
\ImageAltText	\includegraphics and other images' alt tag.
\ThisAltText {⟨text⟩}	Assigns an alt/title tag for the next image or link.
\MathImageAltText	The SVG math image lateximage alt tag.
\PackageDiagramAltText	The suffix for a package's lateximage alt tags.
\AltTextOpen	Start an HTML alt tag.
\AltTextClose	End an HTML alt tag.
\CSSFilename	The CSS for the following files.
\MathJaxFilename	The MathJax script 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 16 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 browser display is wide enough. When using a narrow web browser window, "responsive web design" is used to show the sideTOC at the top of the page, as well as a link back to **Home** at the top and bottom.

It is recommended to set:

```
SideTOCDepth = FileDepth
```

or

```
SideTOCDepth = FileDepth+1
```

⚠ inaccessible pages

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

\sidetocname
Default: Contents

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

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

FileSectionNames: If true, web page filenames are derived from a sanitized version of the section names. If false, web pages are numbered. Either way, the HTMLFilename option is used as a prefix. See section 7.6.1 for examples of naming and numbering HTML files. The user must ensure that filenames are unique after 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.

⚠ Unique filename!

\FilenameLimit
Default: 80

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

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". Redefine with \renewcommand.

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

\CSSFilename
Default: lwarp.css

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

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

\MathJaxFilename
Default: lwarp_mathjax.txt

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

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

\HTMLLanguage
Default: en-US

\HTMLLanguage: {⟨*langauge*⟩} The HTML file's HTML lang meta tag. Defaults to en-US.

\HTMLTitle
Default: \thetitle

\HTMLTitle: {⟨*title*⟩} Overrides \title for the HTML header's meta title. De-

faults 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:

\ImageAltText
Default: image

\ImageAltText: Redefine with \renewcommand. \includegraphics and other images are assigned an HTML alt tag according to \ImageAltText along

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

\ThisAltText

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

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

\MathImageAltText
Default: math image

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

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

\PackageDiagramAltText
Default: diagram

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

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

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

\AltTextOpen
Default: (

\AltTextOpen: Redefine with \renewcommand.

\AltTextClose
Default:)

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

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.

7.6.1 Example HTML file naming

Examples of ways to name or number HTML files:

Numbered HTML nodes:

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

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

Named HTML sections, no prefix:

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

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

Named HTML sections, with prefix:

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

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

7.7 Customizing the css

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

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

```
...
\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 --- ) */
```

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

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

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

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

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

7.8 Selecting the operating system

Prog Unix
Prog Mac OS
Prog Linux

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.

Prog MS-Windows
Prog Windows
Opt OSWindows

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.

7.9 Selecting actions for print, HTML, or MATHJAX output

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

For most of built-in LaTeX and many additional packages there is user-level source code support or emulation, so no special handling will be required. For those cases which lwarp does not handle by itself, the following environments and macros may be used to isolate sections of code for print-only or HTML-only.

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

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.

Env warpMathJax Anything which is to be done only while using HTML output with MathJax is surrounded by a warpMathJax environment. Usually, this is \CustomizeMathJax, used to add emulation macros.

7.10 Commands to be placed into the warpprint environment

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

- Paragraph formatting: \parindent \parskip
- Manual page positions such as the textpos package, which is emulated but only in a limited way.
- Anything changing the page counter. lwarp requires that the page counter not be adjusted during HTML output.

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

Also see section 13: Troubleshooting.

7.11 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 66.8.

`\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*⟩}

⚠ HTML corrupted ⚠ newlines Avoid newlines in the `\title`; these will interfere with the file break and CSS detection. Use a `\subtitle` command instead (section 66.8). The title will appear in the document `\maketitle` as a heading `<h1>`. The HTML meta `title` tag will also have this title, unless `\HTMLTitle` is used to set the meta title to something else instead.

`\author` {⟨*author*⟩}

In `\author`, `\protect` may be needed before some formatting commands. In HTML, the author will appear in a `<div>` of class `author` in the `\maketitle`. If the titling package is used, the author will also appear in a HTML meta tag, but `\HTMLAuthor` may be necessary to create a plain list of names if `\author` had affiliations added. `\affiliation` is a new addition to lwarp.

`\date` {⟨*date*⟩}

`\date` works as expected. In HTML, this will appear in a `<div>` class `titledate`.

`\thanks` {⟨*text*⟩}

`\thanks` are allowed in the titlepage fields, and will be rendered as HTML notes at the bottom of the title page.

7.12 HTML page meta descriptions

\HTMLDescription {⟨*A description of the web page.*⟩}
Default: (none)

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{}
```

7.13 HTML homepage meta title

\HTMLTitle {⟨*title*⟩}
Default: \HTMLtitle{\thetitle}

Sets the contents of the web page <meta name="title"> element. May be set empty to cancel the meta title tag.

See section 7.6 for \HTMLTitleBeforeSection and \HTMLTitleAfterSection, used to set the title for HTML subpages.

7.14 HTML page meta author

\HTMLAuthor {⟨*author*⟩}
Default: \HTMLAuthor{\theauthor}

Sets the contents of the web page <meta name="author"> element. May be set empty to cancel the meta author tag.

\author may be used to create a list of authors and their affiliations, in several formats if using authblk, and these may not successfully parse properly into a sensible list for \theauthor. \HTMLAuthor may be used to set the meta tag to a simple list of names.

8 Special cases and limitations

Some commonly-used LaTeX expressions should be modified as follows to allow for a smooth conversion to both HTML and print-formatted outputs.

Need help?

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

8.1 Things to avoid

In the document, avoid the following:

⚠ options with braces

Package options: Package options may cause problems with lwarp, especially if they include curley braces.

If selecting options with braces in `\usepackage` does not work:

```
\usepackage[font={it,small}]{caption}% does not work
```

... try instead selecting the package options before loading lwarp:

```
\PassOptionsToPackage{font={it,small}}{caption}
...
\usepackage{lwarp}
...
\usepackage{caption}
```

... or try setting package options after the package has been loaded:

```
\usepackage{caption}
\captionsetup{font={it,small}}
```

page counter: Do not adjust the page counter. If doing so is required for the print version, place the adjustment inside a `warpprint` environment.

Custom math environment macros: Do not use expressions such as `\beq` as a replacement for `\begin{equation}`.

Custom macros in section, figure, table names: Custom macros which appear in sectioning commands or float captions then appear in the `.toc`, `.lof`, and `.lot` lists, and should be made robust using `\newrobustcmd` or `\robustify` from etoolbox, xparse, etc.

When setting `FileSectionNames` to `true` to name the HTML files from the section names, the file names are created from sanitized versions of the chapter or section names, but the section names must be plain text or something which expands into plain text. Robust macros will not work at the sectioning level which is used for file names, but a robust macro or other complicated name may be used for the manditory argument of `\chapter`, `\section`, etc., if a plain-text version is also included in the optional argument:

```
\chapter[Plain Name]{\ARobustMacro{Fancy Name}}
```

8.1.1 Invalid HTML

Additionally, some objects are valid LaTeX, but invalid HTML. An example is a tabular inside `\textbf`, since HTML does not allow a table inside a span. lwarp will create the table, and the browser may support it, but the result is technically invalid.

8.2 Formatting

8.2.1 Text formatting

⚠ `\bfseries`, etc. — `\textbf`, etc. are supported, but `\bfseries`, etc. work only in some situations.

⚠ HTML special chars — &, <, and > have special meanings in HTML. If `\&`, `\textless`, and `\textgreater` are used, proper HTML entities will be used, but there may be HTML parsing problems if these special characters occur unescaped in program listings or other verbatim text.

program listings — For program listings, the listings package is supported, and its `literate` option is used to convert &, <, and > to proper HTML entities.

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

8.2.2 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.

8.2.3 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}
. . .
```

8.2.4 Accents

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

8.2.5 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 system / browser fonts. In the CSS try referencing fonts which are more complete, but expect to see gaps in coverage.

8.2.6 Superscripts and other non-math uses of math mode

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

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

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

8.2.8 Filenames and URLs in lists or footnotes

filename underscore Escape underscores in the filenames:

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

8.2.9 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.

8.3 Boxes and minipages

8.3.1 Marginpars

`\marginpar` `[`⟨*left*⟩`]` `{`⟨*right*⟩`}` `\marginpar` may contains paragraphs, but in order to re-

main inline with the surrounding text lwarp nullifies block-related macros inside the `\marginpar`. Paragraph breaks are converted to `<br />` tags.

`\marginparBlock` `[`⟨*left*⟩`]{`⟨*right*⟩`}` To include block-related macros, use `\marginparBlock`, which takes the same arguments but creates a `<div>` instead of a `<span>`. A line break will occur in the text where the `\marginBlock` occurs.

8.3.2 Save Boxes

⚠ HTML corrupted

⚠ boxes TEX boxes are placed inline and do not allow line breaks, so boxes with long contents may overflow the line during HTML conversion. lwarp uses methods which help avoid this problem.

⚠ `minipage`, `\parbox` `\savebox` and related do not (yet) support `minipage` or `\parbox`.

8.3.3 Minipages

⚠ inline A line of text with an inline `minipage` or `\parbox` will have the `minipage` or `\parbox` placed onto its own line, because a paragraph is a block element and cannot be made `inline-block`.

placement `minipages` and `\parboxes` will be placed side-by-side in HTML unless you place a `\newline` between them.

side-by-side Side-by-side `minipages` may be separated by `\quad`, `\qquad`, `\enskip`, `\hspace`, `\hfill`, or a `\rule`. When inside a `center` environment, the result is similar in print and HTML. Paragraph tags are suppressed between side-by-side `minipages` and these spacing commands, but not at the start or end of the paragraph.

⚠ minipage in a span There is limited support for `minipages` inside an HTML `<span>`. An HTML `<div>` cannot appear inside a `<span>`. While in a `<span>`, `minipages`, and `\parboxes`, and any enclosed lists have limited HTML tags, resulting in an "inline" format, without markup except for HTML breaks. Use `\newline` or `\par` for an HTML break.

⚠ minipage size When using `minipage`, `\parbox`, and `fminipage`, a virtual 6×9 inch text area is used for `\linewidth`, `\textwidth`, and `\textheight`, both for sizing the minipage, and also for its contents.

if width is `\linewidth` If a `minipage` or `\parbox` is assigned a width of exactly `\linewidth`, in HTML it is automatically given no HTML width, thus allowed to fill the line as needed, similar to how it appears in print output.

full-width if HTML A new macro `\minipagefullwidth` requests that, during HTML output, the next single `minipage` or `\parbox` be generated without an HTML `width` attribute, allowing it to be the full width of the display rather than the declared print-output width. This may be useful where the printed version's width makes no sense in HTML.

⚠ `tabular`, `multicols` Inside a `tabular` or `multicols` environment, where the width depends on the browser window, `\minipagefullwidth` is effectively used by default for every `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.

⚠ multicol Inside a multicols, \linewidth is divided by the specified number of columns.

⚠ text alignment Nested minipages adopt their parent's text alignment in HTML, whereas in regular LaTeX PDF output they do not. Use a flushleft or similar environment in the child minipage to force a text alignment.

8.3.4 Side-by-side minipages

Place side-by-side minipages inside a center environment, with horizontal space between them, such as \quad, \qquad, \hspace, or \hfill. The result is similar in print and HTML. Do not use space commands at the start or end of the line.

8.3.5 Framed minipages and other environments

\fbox can only be used around inline 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 85 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 535 for xcolor's \colorbox and \fcolorbox, and lwarp's additional \colorboxBlock and \fcolorboxBlock.

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

⚠ Misplaced alignment tab character &

To frame equations: See section 224 for the fancybox package.

For fancy framed minipages: See packages boxedminipage2e, shadow, fancybox, framed, mdframed.

Custom environments: Use a custom environment to create a sidebar, containing a BlockClass environment with custom css formatting, and \warpprintonly{\hrule} command:

```
\begin{BlockClass}{frameminipage}% ignored in print output
  % use \CSS\ to format div class ``framedminipage''
\warpprintonly{\hrule} % only appears in print output
Contents
\warpprintonly{\hrule} % only appears in print output
\end{BlockClass}
```

8.3.6 fancybox package

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.

8.3.7 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.

8.4 Section names

If using named HTML files, by selecting \booltrue{FileSectionNames}, the generated filenames may be simplified by using \FilenameSimplify and \FilenameNullify:

\FilenameSimplify {⟨*text*⟩}

To remove common short words from the automatically-generated filenames, replacing each with a single hyphen "-", use \FilenameSimplify:

```
\FilenameSimplify*{-in-}
\FilenameSimplify*{A-}
```

The first example removes the word "in" in the middle of a filename, and the second example removes "A" at the start of the filename. The star forces the arguments to be detokenized, which is required for a plain-text comparison. (The unstarred form is used for a token-sensitive comparison, which is seldom required by the user.) After simplfication, repeated hyphen characters will be further simplified to a single hyphen "-". Finally, single hyphens at the start or end of the filename are removed.

\FilenameNullify {⟨*macros*⟩}

⚠ macros in section names

Macro names may appear in the automatically-generated file names. To remove these, create *non-robust* nullified versions of the macros, ensuring that each line ends with a percent character % as shown below. These are placed inside \FilenameNullify, which adds them to the list of macros which are nullfiied during filename generation. Low-level macros such as \begingroup will cause problems when nullfied. Many macros such as \textbf are already nullfied. lwarp also already nullifies built-in symbol and textcomp macros, including if defined by xunicode, but not all xunicode macros. See the definition of \LWR@nullfonts for a complete list.

```
\FilenameNullify{%
  \renewcommand*{\macroname}[1]{#1}%
  \renewcommand*{\anothermacro}{}%
}
```

⚠ duplicate filename

Avoid duplicate file names. Section names at levels which result in HTML file splits must be unique. lwarp will generate an error if a duplicate HTML filename is generated.

Use the optional TOC caption entry parameter for formatting. Remember to `\protect` LaTeX commands which appear in section names and TOC captions.

⚠ math in section names

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

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

8.5 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.

8.5.1 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 478.

8.5.2 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 550 to redefine the message which is printed for page number references.

8.5.3 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.

8.5.4 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.

footnote numbering

To have footnote numbers reset each time footnotes are printed:

```
\setcounter{footnoteReset}{1}
```

For bigfoot, manyfoot, or perpage:

```
\MakePerPage{footnoteX}
```
— or —
```
\MakeSortedPerPage{footnoteX}
```

The footnotes are reset when they are printed, according to section level as set by `FootnoteDepth`, which is not necessarily by HTML page. This is recommended for `\alph`, `\Alph`, or `\fnsymbol` footnotes, due to the limited number of symbols which are available.

footmisc

The footmisc `stable` option is emulated by lwarp.

⚠ sectioning commands

When using footnotes in sectioning commands, to generate consistent results between print and HTML, use the footmisc package with the `stable` option, provide a short TOC entry, and `\protect` the `\footnote`:

```
\usepackage[stable]{footmisc}
. . .
\subsection[Subsection Name]
    {Subsection Name\protect\footnote{A footnote.}}
```

memoir with footmisc
⚠ memoir

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

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

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

fancybox, fancyvrb
⚠ `\VerbatimFootnotes`
⚠ sectioning or displaymath

If using fancybox or fancyvrb with `\VerbatimFootnotes`, and using footnotes in a sectioning command or display math, use `\footnotemark` and `\footnotetext`:

```
\subsection[Subsection Name]
    {Subsection Name\protect\footnotemark}
\footnotetext{A footnote with \verb+verbtim+.}
```

and likewise for equations or display math.

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

pfnote
⚠ pfnote numbers

While emulating pfnote, lwarp is not able to reset HTML footnote numbers per page number to match the printed version, as HTML has no concept of page numbers. lwarp therefore uses continuous footnote numbering even for pfnote.

bigfoot, manyfoot
⚠ verbatim

Verbatim footnotes are not yet supported.

If using the bigfoot package, and possibly also manyfoot, problems may occur with counter allocation because lwarp uses many counters, and there is a difference in how counters numbered 256 and up are handled in pdfLaTeX. With bigfoot this has been known to show up as an error related to one footnote insert being forbidden inside another. Another problem showed up as a input stack error, and which of these problems occurred depended on how many counters were allocated.

As a possible solution, try creating several new counters before defining bigfoot or manyfoot footnotes, hoping to shift the problematic counter above the 256 threshold. It may instead be necessary to use XeLaTeX or LuaLaTeX instead of pdfLaTeX.

8.5.5 xr, xr-hyper, and xcite packages

See section 5.17.

8.6 Front and back matter

8.6.1 Custom classes with multiple authors and affiliations

Some classes allow multiple authors and affiliations. Often it is possible to emulate these using a standard class along with authblk:

```
%\documentclass{customclass} % for print document
\documentclass{article} % for html document

\usepackage{lwarp}
\begin{warpHTML}
\usepackage{authblk}
\let\affiliation\affil % maybe required
\end{warpHTML}
```

8.6.2 Starred chapters and sections

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.

8.6.3 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.

8.6.4 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 66.8.

8.6.5 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.

8.6.6 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.

8.6.7 pagenote package

Pkg pagenote

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

8.6.8 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
```

8.6.9 *BibTeX*

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

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

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

8.6.10 xcite package

See section 5.17.

8.6.11 gloss package

Pkg gloss To process the HTML glossary:

⚠ compiling

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

8.6.12 glossaries package

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

processing glossaries

Opt GlossaryCmd

Default: makeglossaries

Opt [lwarpmk] printglossary

Opt [lwarpmk] htmlglossary

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

⚠ *makeglossaries* not found In some situations it may be required to modify the default command, such as to add the **perl** command in front:

```
\usepackage[
   GlossaryCmd={perl makeglossaries},
] {lwarp}
```

xindy language To set the language to use for processing glossaries with *xindy*:

```
\usepackage[
   GlossaryCmd={makeglossaries -L english},
] {lwarp}
```

Other options for *makeglossaries* may be set as well.

placement and TOC options The glossaries may be placed in a numbered or unnumbered section, given a TOC entry, and placed inline or on their own HTML page:

Numbered section, on its own HTML page:

```
\usepackage[xindy,toc,numberedsection=nolabel]{glossaries}
. . .
\printglossaries
```

Unnumbered section, inline with the current HTML page:

```
\usepackage[xindy,toc]{glossaries}
. . .
\printglossaries
```

Unnumbered section, on its own HTML page:

```
\usepackage[xindy,toc]{glossaries}
. . .
\ForceHTMLPage
\printglossaries
```

⚠ glossary style

The default `style=item` option for glossaries conflicts with lwarp, so the style is forced to `index` instead.

⚠ number list

The page number list in the printed form would become `\namerefs` in HTML, which could become a very long string if many items are referenced. For now, the number list is simply turned off.

print/HTML versions

The print and HTML versions of the glossary differ in their internal page numbers. Separate commands for generating print and HTML glossaries are used, even though the page number is currently ignored.

8.6.13 nomencl package

Pkg nomencl

To process the HTML nomenclature:

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

8.6.14 Indexing overview

There are many ways to process indexes for a LaTeX document, including native LaTeX capabilities, a number of packages and classes, the possible availability of shell escape and *latexmk*, and the need to process print and HTML versions. lwarp attempts to provide easy recompilation of indexes along with the rest of the document, but the various indexing options must be set correctly. Numerous examples are given below. Some differ in minor details, so the important parts are highlighted in red, and options are in green.

Once set up properly, the entire document may be recompiled with **`lwarpmk print`** and **`lwarpmk html`**. In some cases, it will also be necessary to compile the indexes with **`lwarpmk printindex`** and **`lwarpmk htmlindex`**. A recompile may then be forced with **`lwarpmk print1`** and **`lwarpmk html1`**.

manual processing

The user may continue to process indexes manually or by shell script without the use of *lwarpmk*, but adjustments will be required to process HTML indexes as well. In

general, `*.idx` and `*.ind` files will be accompanied by `*_html.idx` and `*_html.ind` files.

custom index style

If using a custom indexing style file, see sections 8.6.20 and 8.6.21.

source code

See section 76 for lwarp's core index and glossary code, section 290 for index, section 457 for splitidx, section 289 for imakeidx, section 495 for tocbibind, and section 555.17 for memoir's indexing patches.

8.6.15 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.

8.6.16 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.

8.6.17 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 555.17 in the lwarp-patch-memoir package for the \@@wrspindexhyp macro as an example.

For multiple indexes using *makeindex* and splitidx:

```
\usepackage[
  makeindex, latexmk,
  PrintIndexCmd={
    splitindex <projectname> -- -s lwarp.ist
  },
  HTMLIndexCmd={
    splitindex <projectname>_html -- -s lwarp.ist
  }
]{lwarp}
\usepackage{splitidx}
. . .
\makeindex
\newindex[Second Index]{secondname}
```

When creating the HTML index, "_html" is automatically appended to each of the names.

Use

Enter ⇒ **lwarpmk printindex**

Enter ⇒ **lwarpmk htmlindex**

to compile the indexes.

For multiple indexes using *xindy* and splitidx:

```
\usepackage[
  xindy, latexmk,
  PrintIndexCmd={
    splitindex -m xindy <projectname> -- -M lwarp.xdy
      -L english -C utf8                          <optional>
  },
  HTMLIndexCmd={
    splitindex -m xindy <projectname>_html -- -M lwarp.xdy
      -L english -C utf8                          <optional>
  }
]{lwarp}
\usepackage{splitidx}
. . .
\makeindex
\newindex[Second Index]{secondname}
```

When creating the HTML index, "_html" is automatically appended to each of the names.

Use

Enter ⇒ **lwarpmk printindex**

Enter ⇒ **lwarpmk htmlindex**

to compile the indexes.

8.6.18 Indexing with imakeidx

Due to the number of methods which may be used to process multiple indexes, the options for style file and *xindy* language and codepage must be specified in one of several different ways. These are described in detail later in this section, but are summarized here.

If shell escape is used, imakeidx will automatically compile the indexes by itself. Options specifying a custom style file and *xindy* language and codepage must be specified for each `\makeindex` command using its `options=` option, which must include lwarp's special `lwarp.ist` or `lwarp.xdy` file, or a file based on them. If using a custom indexing style file, see sections 8.6.20 and 8.6.21. 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.

8.6.19 Indexes with memoir

For a single index with memoir and makeindex:

```
\documentclass{memoir}
\usepackage[makeindex,latexmk]{lwarp}
. . .
\makeindex
```

The usual `.idx` and `.ind` files will be used, along with the `lwarp.ist` style file.

lwarpmk will use *latexmk* if specified, in which case *latexmk* will create the index automatically. Otherwise, use

Enter ⇒ **`lwarpmk printindex`**

Enter ⇒ **`lwarpmk htmlindex`**

to compile the indexes.

For multiple indexes with memoir and makeindex, using *latexmk*:

lwarp's options are used, and are passed to *latexmk*.

```
\documentclass{memoir}
\usepackage[makeindex,latexmk]{lwarp}
. . .
\makeindex
\makeindex[secondname]
```

lwarpmk will use *latexmk* to create the indexes automatically when the user executes **`lwarpmk print`** and **`lwarpmk html`**.

For multiple indexes with memoir and makeindex, *without* latexmk:

lwarpmk must be told how to create the indexes:

```
\documentclass{memoir}
\usepackage[
  makeindex,
  PrintIndexCmd={
    makeindex -s lwarp.ist <projectname>.idx ;
    makeindex -s lwarp.ist secondname.idx
  },
  HTMLIndexCmd={
    makeindex -s lwarp.ist <projectname>_html.idx ;
    makeindex -s lwarp.ist secondname_html.idx
  }
]{lwarp}
. . .
\makeindex
\makeindex[secondname]
```

⚠ WINDOWS *For WINDOWS, replace the two ";" characters with "&".*

`<projectname>` is the `\jobname`: if compiling "`name.tex`", use the filenames `name.idx` and `name_html.idx`.

Use

Enter ⇒ **`lwarpmk printindex`**

Enter ⇒ **`lwarpmk htmlindex`**

to compile the indexes.

For a single index with memoir and xindy:

```
\documentclass{memoir}
\usepackage[
    xindy,
    xindyLanguage=english,                    <optional>
    xindyCodepage=utf8,                       <optional>
    latexmk                                   <optional>
]{lwarp}
. . .
\xindyindex
\makeindex
```

The usual `.idx` and `.ind` files will be used, along with the `lwarp.xdy` style file.

lwarpmk will use *latexmk* if specified, in which case *latexmk* will create the index automatically. Otherwise, use

Enter ⇒ **`lwarpmk printindex`**

Enter ⇒ **`lwarpmk htmlindex`**

to compile the indexes.

For multiple indexes with memoir and xindy, using latexmk:

lwarp's options are used, and are passed to *latexmk*.

```
\documentclass{memoir}
\usepackage[
    xindy,
    xindyLanguage=english,                    <optional>
    xindyCodepage=utf8,                       <optional>
    latexmk
]{lwarp}
. . .
\xindyindex
\makeindex
\makeindex[secondname]
```

lwarpmk will use *latexmk* to create the indexes automatically.

For multiple indexes with memoir and xindy, *without* latexmk:

lwarpmk must be told how to create the indexes:

```
\documentclass{memoir}
\usepackage[
  xindy,
  PrintIndexCmd={
    xindy -M lwarp.xdy -L english -C utf8
      <projectname>.idx ;
    xindy -M lwarp.xdy -L english -C utf8
      secondname.idx
  },
  HTMLIndexCmd={
    xindy -M lwarp.xdy -L english -C utf8
      <projectname>_html.idx ;
    xindy -M lwarp.xdy -L english -C utf8
      secondname_html.idx
  }
]{lwarp}
...
\xindyindex
\makeindex
\makeindex[secondname]
```

⚠ WINDOWS *For WINDOWS, replace the four ";" characters with "&".*

`<projectname>` is the `\jobname`: if compiling "`name.tex`", use the filenames `name.idx` and `name_html.idx`.

Use

Enter ⇒ **`lwarpmk printindex`**

Enter ⇒ **`lwarpmk htmlindex`**

to compile the indexes.

8.6.20 Using a custom *makeindex* style file

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.

8.6.21 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.

8.6.22 Additional indexing limitations

⚠ xindy with hyperref

xindy and hyperref may not work well together for print output with “see”, “see also”, reference ranges, or stylized index references. It may be necessary to turn off hyper-referencing for indexes:

```
\usepackage[hyperindex=false]{hyperref}
```

⚠ empty index

If an HTML index is empty, it may be necessary to add the following before lwarp is

loaded:

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

makeindex custom display styles

When using *makeindex*, custom display styles are possible:

```
\begin{warpprint}
\newcommand{\notesstyle}[1]{#1nn}
\end{warpprint}

\begin{warpHTML}
\makeatletter
\newcommand{\notesstyle}[1]{\LWR@doindexentry{#1} notes }
\makeatother
\end{warpHTML}
. . .
A sentence.\index{key|notesstyle}
```

xindy custom display styles

For custom styles with *xindy*, see `lwarp.xdy` for `\textbf` and `\textit` as examples.

8.6.23 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.

8.7 Math

8.7.1 Math in section names

⚠ math in section names

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

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

8.7.2 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 10 regarding EPUB output with MathJax.

8.7.3 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.

8.7.4 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
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

[14] See section 540 regarding fonts and fractions.

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.

8.7.5 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:

```
\begin{warpMathJax}
\CustomizeMathJax{
    \newcommand{\expval}[1]{\langle#1\rangle}
    \newcommand{\abs}[1]{\lvert#1\rvert}
}
\CustomizeMathJax{\newcommand{\arcsinh}{\text{arcsinh}}}
\CustomizeMathJax{\newcommand{\arccosh}{\text{arccosh}}}
\CustomizeMathJax{\newcommand{\NN}{\mathbb{N}}}
\end{warpMathJax}
```

lwarp already provides MathJax customizations for some packages.

siunitx When using siunitx, a similar process may be used to add custom units:

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\myunit}{\mathrm{WXYZ}}}
\CustomizeMathJax{\newcommand{\umyunit}{\mathrm{\micro\myunit}}}
\end{warpMathJax}
```

8.7.6 MathJax limitations

MathJax limitations Limitations when using MathJax include:

Prog MathJax

⚠ \multicolumn, multirow

- MathJax does not support \multicolumn or multirow. These may be used in text tabulars or SVG math, but in MathJax math arrays they are emulated. \multicolumn only fills a single cell, resulting in a short row. \multirow simply prints its text on the first line.

⚠ subequations

- MathJax 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. lwarp provides a basic emulation for MathJax.

⚠ references in math

- Inside a MathJax expression, references to equations work within the same HTML web page, but do not work when referring to an equation in a different HTML web page. Outside of a MathJax expression, in the text body, references work as expected.

lateximage

- Math appearing inside a lateximage, and therefore also inside a Ti*k*z or picture

environment, is rendered as SVG math even if MathJax is used in the rest of the document.

siunitx
- For siunitx, see siunitx package, section 8.7.12.

physics
- For physics, see physics package, section 8.7.14.

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 directly supported by MathJax, including `\ensuremath` and bigdelim, and occasionally-used macros such as `\relax`. While using MathJax, lwarp provides emulation for footnotes, bm, mathtools, nicefrac, siunitx, and units.

8.7.7 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.

8.7.8 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.

8.7.9 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.

8.7.10 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.

8.7.11 mathtools package

Pkg `mathtools`

⚠ equation numbering

`showonlyrefs` is disabled, as it conflicts with cleveref, which is used by lwarp. Equation numbers may not match the print version.

⚠ italic correction

`mathic` is not emulated for HTML.

⚠ MathJax

If using MathJax:

- Starred macros are not available. Starred environments do work.
- mathtools `disallowspaces` does not work for MathJax. Protect brackets which are not optional arguments, such as:

```
\begin{gathered}{}
[p]=1 . . .
\end{gathered}
```

- `showonlyrefs` does not work in MathJax, and will result in a difference in equation numbering compared to the print version.
- Due to MathJax limitations, the following do not render well: `\overbracket`, `\underbracket`, `\overbrace`, `\underbrace`, `rcases`, `drcases`, `\Aboxed`, and `\ArrowBetweenLines`.
- For the new `cases`-like environments, `\text` must be used to set the normal roman font if desired.
- `alignat` in MathJax requires math mode, but in LaTeXit doesn't. It may be required to use `warpHTML` and `warpprint` to isolate a version for each mode.

- `\DeclareParedDelimiter` and related must be in the preamble before `\begin{document}`. The starred versions of each macro are not created.

8.7.12 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.

⚠ 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`.

⚠ MathJax

For math mode with SVG display, the original siunitx code is used while generating the SVG image. For text mode, lwarp uses an emulation which provides a very effective HTML interpretation of siunitx. For math expressions while using MathJax, a limited emulation is used. Most functions work reasonably well, but many options cannot be emulated. siunitx macros with more than one optional value cannot absorb the second optional value, and complicated parsing such as for `\ang` is not supported. The result usually looks fine, and otherwise is enough to get the meaning across.

lwarp's MathJax emulation for siunitx is meant to be a stop-gap measure until an extension is included in MathJax. As of this writing, the third-party siunitx extension for MathJax is not currently hosted at any public CDN, thus siunitx is not usable with this extension unless a local copy of this extension is created first. See `\MathJaxFilename` to select a custom MathJax script, but lwarp's emulation would have to be diabled as well.

Document modifications required for MathJax:

custom units

- Custom units may be added with `\CustomizeMathJax`. See the lwarp-siunitx code for examples.

⚠ unit spacing

- Units work better using ~ between units instead of using periods.

⚠ `\square, \cubic`

- To square or cube compound units, enclose the following compound units in braces:

  ```
  \cubic{\centi\meter}
  ```

  Single units do not require braces.

Also see MathJax option, section 8.7.4.

8.7.13 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`.

8.7.14 physics package

Pkg physics

physics works as-is for HTML with SVG math.

For MathJax, emulation is provided via MathJax macros. These are not the same as the third-party MathJax extension.

- The `notrig` option is honored.
- ⚠ Starred macros are not yet detected.
- ⚠ Most macros don't work with `\big`, etc.
- ⚠ Macros do not auto-detect variable numbers of mandatory arguments. Provide empty `{}` arguments for those which are not used.
- ⚠ Many of the macros do not work with auto-detected delimiters. Use the delimiter-specific versions instead. Some macros do not even consider the following arguments, so they may work as expected.
- ⚠ For `\Re` and `\Im`, the arguments must be in braces.
- For `\functionalderivative`, for the example in the manual with $(E - TS)$, enclose the parens in braces.
- ⚠ `\expectationvalue` requires and uses two mandatory arguments, unlike the third-party MathJax physics extension.
- Each of `\matrixquantity`, `\smallmatrixquantity`, and `\matrixdeterminant` work, while `\identitymatrix` and the following simply print a place-holder, and must be replaced by hand.

8.7.15 newtxmath package

Pkg newtxmath

⚠ loading sequence

The proper load order is:

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

8.8 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 according to `\ImageAltText`.

`\rotatebox`

`\rotatebox` accepts the optional `origin` key.

⚠ browser support

`\rotatebox`, `\scalebox`, and `\reflectbox` depend on modern browser support. The CSS3 standard declares that when an object is transformed the whitespace which they occupied is preserved, unlike LaTeX, so expect some ugly results for scaling and rotating.

8.8.1 tikz package

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 `\&`.

8.8.2 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.

8.8.3 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.

8.8.4 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.

8.8.5 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.

8.8.6 pstricks package

Pkg pstricks

⚠ use `pspicture`

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

8.8.7 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`**

8.8.8 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.

8.8.9 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**

.

8.8.10 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
```

8.8.11 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.

8.8.12 Multimedia packages

Pkg multimedia

Pkg movie15

Pkg media9

The packages multimedia, movie15, and media9 are supported.

HTML5 <audio> and <video> objects are created for .mp3 and .mp4 files.

HTML5 <embed> objects are created for http and ftp links.

\href links are created for other media types. (Unfortunately, there is not much overlap between the file types supported for print output and the file types supported by HTML5.)

For media9, a multimedia object is inserted for each addresource=, as well as each flashvars source= and src=. This may result in duplicate objects.

Undesired objects may be nullified by placing them inside \warpprintonly or the warpprint environment.

Each HTML multimedia object includes the poster text, except for <embed> objects. For movie15, the text option is supported to specify the poster text.

The width, height, and totalheight options are supported. The HTML object is scaled according to the display width, correctly compensating for either tall or wide viewports.

Other options are ignored.

media9 \addmediapath is supported. It is assumed that the same path structure will exist for the HTML document.

HTML5 media controls are always specified for each <audio> and <video> object.

media9 slideshows are not supported.

\hyperlinkmovie, \movieref, and \mediabutton are not supported.

3D objects are not supported.

If using a YouTube™ video, use an "embedded" URL with .../embed/... instead of .../v/...

8.9 Tabbing

The tabbing environment works, except that SVG math and lateximages do not yet work inside the environment.

⚠ math in tabbing

If math is used inside tabbing, place tabbing inside a lateximage environment, which will render the entire environment as a single SVG image.

8.10 Tabular

8.10.1 tabular environment

Tabular mostly works as expected, but pay special attention to the following, especially if working with environments, macros inside tabulars, multirows, siunitx S columns, or the packages multirow, longtable, supertabular, or xtab.

Defining macros and environments:

⚠ Misplaced alignment tab character &

- When defining environments or macros which include tabular and in-

stances 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:

⚠ 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:

@ 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. When given an optional width, each cell is styled to create the custom border. Without an optional width, the entire row is given a class to assign the standard border.

combined rules

- If you wish to use \cmidrule followed by \bottomrule, it may be necessary to use:
  ```
  \cmidrule{2-3} \\[-2ex]
  \bottomrule
  ```
  The optional -2ex is ignored in HTML, but improves the visual formatting in the print output.

⚠ \warpprintonly
⚠ Misplaced \noalign

- For \toprule and \bottomrule, when combined with a warpprint or warpHTML environment, if a "Misplaced \noalign" error occurs, change
  ```
  This & That \endhead
  ```
  to
  ```
  \warpprintonly{This & That \endhead}
  ```
  and likewise with the other \end headings. Keep the \endfirsthead row unchanged, as it is still relevent to HTML output.

Other:

- tabularx ignores the width, but X columns do produce paragraph columns or multicolumns.

longtable headings

- For longtable, place headings and footings which do not apply to HTML inside \warpprintonly{}.

⚠ S columns

- For S columns (from the siunitx package), while producing print output, anything non-numeric must be placed inside { } braces, including commands such as \multirow. While producing HTML output, though, anything placed inside braces is not seen by lwarp's tabular handling algorithm. To resolve this problem, make a copy of the row, with one version for print output, containing the extra braces, and another version for HTML output, without the extra braces, such as:
  ```
  \warpprintonly{1 & 2 & {\multirow{2}{2cm}{Text}} & 3 \\}
  \warpHTMLonly{1 & 2 & \multirow{2}{2cm}{Text} & 3 \\}
  ```

⚠ tabular inside a

- In LaTeX, a tabular may be placed inside a minipage, but in HTML a <table> may not be inside a . If this situation is detected, a warning is printed instructing the user to isolate the using \warpprintonly or the warpprint environment.

8.10.2 multirow package

vposn

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

multirow cells

- For multirow, insert \mrowcell into any empty multi-row cells. This will be a null function for the print output, and is a placeholder for parsing the table for HTML output. An error is generated if this is missed.
  ```
  . . . &  \multirow{2}{.5in}{text}  & . . .
  . . . &  \mrowcell                 & . . .
  ```

colored cells

- The multirow documentation regarding colored cells recommends using a negative number of rows. This will not work with lwarp, so \warpprintonly and \warpHTMLonly must be used to make versions for print and HTML.

with \multicolumn

- See section 346.2 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,

⚠ empty cells

\mcolrowcell must be used for each cell of each column and each row to be skipped. An error is generated if this is missed.

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

⚠ MathJax

- MathJax does not support multirow, so it is emulated to only print its text on the first row. \multirow works as expected in text tabulars or SVG math.

8.10.3 longtable package

Pkg longtable

Use one of either \endhead or \endfirsthead for both print and HTML, and use a \warpprintonly macro to disable the other head phrase, and also the \endfoot and \endfirstfoot phrases. (See section 8.10.4 if using threeparttablex.)

```
\begin{longtable}{ [column specifiers] }
[ . . . ] \endfirsthead   % or \endhead, for print and HTML
\warpprintonly{          % not used in HTML
  [ . . . ] \endhead     % or \endfirsthead
  [ . . . ] \endfoot
  [ <lastfoot macros> ] \endlastfoot
}
. . .  table contents . . .
\warpHTMLonly{
  [ <lastfoot macros> ]   % HTML last footer, without \endfoot
                                                % or \endlastfoot.
}
\end{longtable}
```

⚠ Misplaced \noalign

Use the \warpprintonly macro instead of the warpprint environment. Doing so helps avoid “Misplaced \noalign.” when using \begin{warpprint}.

⚠ \kill

\kill is ignored, place a \kill line inside

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

or place it inside \warpprintonly.

⚠ leximage

longtable is not supported inside a \leximage.

8.10.4 threeparttablex package

Pkg threeparttablex

threeparttablex is used with longtable and booktabs as follows:

```
\begin{longtable}{ [column specifiers] }
[ . . . ] \endfirsthead   % or \endhead, for print and HTML
\warpprintonly{          % not used in HTML
  [ . . . ] \endhead     % or \endfirsthead
  [ . . . ] \endfoot
  \bottomrule \insertTableNotes \endlastfoot
}
. . .  table contents . . .
\warpHTMLonly{   % HTML last footer
  \bottomrule
  \UseMinipageWidths     % optional
  \insertTableNotes
  \endlastfoot
}
\end{longtable}
```

table width

The table notes are created using a `\multicolumn`. By default the width is not specified to the browser, so long table notes can cause the table to be spread out horizontally. For HTML output, lwarp guesses the width of the table depending on the number of columns, then restricts its guess to a min/max range. To use this guess for the width of the table notes, use `\UseMinipageWidths` before `\insertTableNotes`. The width is then specified, and in many cases the result is an improvement in overall table layout.

8.10.5 supertabular and xtab packages

Pkg supertabular

Pkg xtab

⚠ Misplaced alignment tab character &

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

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

See section 8.10.1.

⚠ lateximage

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

8.10.6 colortbl package

Pkg colortbl

⚠ row/cell color

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

colortbl ignores the overhang arguments.

8.10.7 ctable package

⚠ Misplaced alignment tab character &

Use `\StartDefiningTabulars` before one or more `\ctables`, and `\StopDefiningTabulars`

after. These change the meaning of the ampersand & character.

8.10.8 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{\{}{3}{.25in}[left ] & c & d \\
\mrowcell & e & f \\
\mrowcell & g & h \\
<empty> & i & j \\
\end{tabular}
```

<-> a b

left { c d
e f
g h

<-> i j

For MathJax, limited emulation is provided which merely prints the delimter and optional text in the first row.

8.11 Floats

8.11.1 Float contents alignment

⚠ figure & table alignment

\centering, etc. are honored in a figure or table if they are the first command inside the float:

```
\begin{table*}
\centering
\caption{A Table}
. . .
```

8.11.2 float, trivfloat, and/or algorithmicx together

Pkg float
Pkg trivfloat
Pkg algorithmicx

If using \newfloat, trivfloat, and/or algorithmicx together, see section 506.1.

8.11.3 caption and subcaption packages

⚠ package conflicts

Pkg caption
Pkg subcaption

Package options may cause problems with lwarp, especially if they include curley braces.

If selecting options with braces in \usepackage does not work:

```
\usepackage[font={it,small}]{caption}% does not work
```

... try instead selecting the package options before loading lwarp:

```
\PassOptionsToPackage{font={it,small}}{caption}
. . .
\usepackage{lwarp}
. . .
\usepackage{caption}
```

... or try setting package options after the package has been loaded:

```
\usepackage{caption}
\captionsetup{font={it,small}}
```

⚠ numbering To ensure proper float numbering, set caption positions such as:

```
\captionsetup[figure]{position=bottom}
\captionsetup[subfigure]{position=bottom}
\captionsetup[table]{position=top}
\captionsetup[subtable]{position=top}
```

Similarly for longtable. These positions depend on where the user places the \caption command inside each float.

8.11.4 subfig package

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.

8.11.5 floatrow package

Pkg floatrow Use \StartDefiningTabulars and \StopDefiningTabulars before and after defining macros using \ttabbox with a tabular inside. See section 8.10.1.

⚠ Misplaced alignment tab character &
⚠ subfig package When combined with the subfig package, while inside a subfloatrow \ffigbox and \ttabbox must have the caption in the first of the two of the mandatory arguments.

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

- Use fixed lengths. lwarp will scale the HTML lengths appropriately.

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

8.11.6 keyfloat package

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.)

8.12 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.

8.13 MEMOIR class

Cls memoir

⚠ options clash

While emulating memoir, lwarp pre-loads a number of packages (section 555.1). 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.

8.14 International languages

⚠ section and file names

If using *pdflatex* with the setting \booltrue{FileSectionNames}, non-ASCII text in section names can result in corrupted HTML file names. *pdflatex* may be used if setting \boolfalse{FileSectionNames}, in which case HTML file numbers will be generated.

For correct HTML file names, use *xelatex*, *lualatex*, or dedicated document classes / engines.

(As of this writing, this warning is only relevent to the kotex package.)

8.15 Miscellaneous packages

8.15.1 verse and memoir

Pkg verse

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

Cls memoir
\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.

8.15.2 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}
. . .
```

8.15.3 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.

8.15.4 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.

8.15.5 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.

8.15.6 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}
```

8.15.7 chemfig package

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

```
\begin{lateximage}[-chemfig-~\PackageDiagramAltText]
\chemfig{. . . }
\polymerdelim[. . . ]{. . . }
\end{lateximage}
```

8.15.8 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.

8.15.9 mhchem package

See section 332.

8.15.10 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}
```

8.15.11 kotex package

Pkg kotex

See section 8.14 regarding *pdflatex* and Korean section names.

⚠ Korean section names

9 Compiling using custom shell commands

lwarp and *lwarpmk* try to make it easy to process print and HTML compilation tasks in most situations. Depending on the operating system, command-line options, TEX engine, and lwarp options, the commands **`lwarpmk print`** and **`lwarpmk html`** are automatically set up to correctly recompile the project. These actions may be overridden using lwarp options, thus allowing the use of packages such as perltex and pythontex.

9.1 Command options

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`.

9.2 Literal character macros

The lwarp package options are parsed by TEX, and so some characters require the use of a special macro to represent them. See table 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
      }
}
```

9.3 *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.

9.4 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 8.7.9.

9.5 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.

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

9.6 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.

9.7 *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`**.

9.8 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.

10 EPUB conversion

lwarp does not produce EPUB documents, but it may be told to modify its HTML output to greatly assist in the conversion. An external program may then be used to finish the conversion to EPUB.

<meta> author

To assign the author's name for regular lwarp HTML files, and also for the EPUB, use `\HTMLAuthor {⟨name⟩}`. This assigns the name to the `<meta>` author element. It may be set empty, and it defaults to `\theauthor`.

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

`FormatEPUB`

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.

11 Word-processor conversion

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

11.1 Activating word-processor conversion

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

`FormatWP`

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.

11.2 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
\book				
\part	<h2>	<h1>	<h2>	<h1>
\chapter	<h3>	<h2>	—	—
\section	<h4>	<h3>	<h3>	<h2>
\subsection	<h5>	<h4>	<h4>	<h3>
\paragraph	<h6>	<h5>	<h5>	<h4>
\subparagraph		<h6>	<h6>	<h5>

* For default depths when not FormatWP, see table 9 on page 193.

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 177.

11.3 Recommendations

TOC, LOF, LOT

For use with *LibreOffice Writer*, it is recommended to:

1. Set \booltrue{FormatWP}
2. Set \booltrue{WPMarkTOC} and \boolfalse{WPMarkLOFT}
3. Use lwarp to generate the HTML document.
4. Copy/paste from the HTML document into an empty *LibreOffice Writer* document.
5. Manually insert a LibreOffice TOC in the LibreOffice document.
6. Manually add frames around each float, adding a caption which is cut/pasted from each float's simulated caption.
7. Manually create cross references.

This process yields a document with an actual LibreOffice Table of Contents, but a simulated List of Figures and List of Tables.

siunitx

For siunitx, remember to adjust the preamble as mentioned above.

LO view border options

LibreOffice has options in the **View** menu to turn on/off the display of thin borders around table cells and text objects.

11.4 Limitations

Floats and captions are not explicitly converted to LibreOffice floats with their own captions. Floats are surrounded by a thin frame in the LibreOffice editor, and may be marked with `WPMarkFloats`, but are not given a proper LibreOffice object frame. Captions are given an explicit italic formatting, but not a proper LibreOffice paragraph style.

Cross references are not actual LibreOffice linked cross references.

The List of Figures and List of Tables are not linked. The pasted pseudo LOF and LOT match the numbering of the LaTeX and HTML versions.

Equation numbering is not automatic, but the equation numbers in SVG math will match the LaTeX and HTML output. SVG math is recommended when using the $\mathcal{AMS}$ environments, which may have multiple numbered equations per object.

As of when last checked, LibreOffice ignores the following:

- Minipage alignment.
- Tabular cell vertical alignment.
- Image rotation and scaling.
- Rounded border corners, which are also used by:
  - `\textcircled`
  - booktabs trim
- `\hspace` and rules, also used by algorithmic.
- Coloring of text decorations, used by soul and ulem.
- Overline text decoration, used by romanbar.

LibreOffice also has limitations with frames and backgrounds:

- Multiple lines in an object are framed individually instead of as a whole.
- Nested frames are not handled correctly.
- Images inside boxes are not framed correctly.
- Spans with background colors and frames are not displayed correctly.

12 Modifying lwarp

locating something

To quickly find the source for a package in `lwarp.dtx`, search for `*packagename`, such as `*siunitx`.

Likewise, to quickly find the source for a file in `lwarp.dtx`, search for `*filename`, such as `*lwarp.css`.

Purely text-based packages probably will work as-is when generating HTML.

Look to existing code for ideas on how to expand into new code.

image of TEX output

An environment may be converted to a `lateximage` then displayed with an image of the resulting LATEX output. See section 88 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
```

12.1 Creating a development system

The following creates a local development system for lwarp on a TeXLive system in a UNIX-like environment. Doing so allows anything requesting lwarp to use the development version instead of whichever version is installed in TeXLive.

Create a development directory:

Place into this directory `lwarp.dtx` and `lwarp.ins`.

To create `lwarp.sty`, execute

Enter ⇒ **`pdflatex lwarp.ins`**

which creates `lwarp.sty` and several hundred additional `lwarp-*.sty` files for the various packages which are supported.

To create the 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}
```

12.2 Modifying a package for lwarp

If a class loads additional packages, it will be required to modify the class for lwarp, since lwarp must be loaded before most other packages.

To work with lwarp, a class must first set up anything which replicates the functions of the basic LaTeX classes, load any required fonts, then load lwarp, then finally load and adjust any other required packages.

When creating HTML, lwarp redefines the `\usepackage` and `\RequirePackage` macros such that it first looks to see if a `lwarp-<packagename>.sty` version exists. If so, the lwarp version is used instead. This modular system allows users to create their own versions of packages for lwarp to use for HTML, simply by creating a new package with a `lwarp-` prefix. If placed in the local directory along with the source code, it will be seen by that project alone. If placed alongside the other `lwarp-` packages where TeX

can see it, then the user's new package will be seen by any documents using lwarp. (Remember `mktexlsr` or `texhash`.)

An `lwarp-<packagename>.sty` package is only used during HTML generation. Its purpose is to pretend to be the original package, while modify anything necessary to create a successful HTML conversion. For many packages it is sufficient to simply provide nullified macros, lengths, counters, etc. for anything which the original package does, while passing the raw text on to be typeset. See the pre-existing `lwarp-` packages for examples.

Anything the user might expect of the original package must be replaced or emulated by the new `lwarp-` package, including package options, user-adjustable counters, lengths, and booleans, and conditional behaviors. In many of these packages, most of the new definitions have a "local" prefix according to the package name, and `@` characters inside the name, which hides these names from the user. In most cases these macros will not need to be emulated for HTML output. Only the "user-facing" macros need to be nullified or emulated.

Each `lwarp-*` package should first call either of:

```
\LWR@ProvidesPackageDrop
  -or-
\LWR@ProvidesPackagePass
```

If "`Drop`"ped, the original print-version package is ignored, and only the `lwarp-` version is used. Use this where the original print version is useless for HTML. If "`Pass`"ed, the original package is loaded first, with the user-supplied options, then the `lwarp-` version continues loading as well. See section 365 (ntheorem) for an example of selectively disabling user options for a package. Use this when HTML output only requires some modifications of the original package. For a case where the original package is usable without changes, there is no need to create a `lwarp-` version.

12.2.1 Adding a package to the lwarp.dtx file

When adding a package to `lwarp.dtx` for permanent 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 20.1. This causes lwarp to stop with an error if packagename is loaded before lwarp. Finally, add an entry in table 2, Supported packages and features, and also the Updates section.

12.3 Modifying a class for lwarp

If a class loads additional packages, it will be required to modify the class for lwarp, since lwarp must be loaded before most other packages.

To work with lwarp, a class must first set up anything which replicates the functions of the basic LaTeX classes, load any required fonts, then load lwarp, then finally load and adjust any other required packages.

12.4 Testing lwarp

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.

12.5 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`.

13 Troubleshooting

13.1 lwarp package error conditions and warnings

lwarp tests for a number of error conditions and prints appropriate warnings. The following is a summary of these conditions.

13.1.1 Configuration file `lwarpmk.conf`

File does not exist: The configuration file must exist for lwarpmk.

Incorrect `Unix` /`Windows` selection: The operating system which was detected by lwarp. So far only `Unix` and `Windows` are supported.

Incorrect delimiter characters. Older versions of *lwarpmk* used a different delimiter.

Source name is set to `lwarp`: lwarp has recently been recompiled in this directory, which overwrote the project's configuration files. This also occurs if *lwarpmk* is executed in lwarp's source directory.

Incorrect operating system: The configuratio file was set for a different operating system, perhaps due to sharing in a collaborative project.

Outdated configuration files: lwarp has been updated since this projects was last compiled. If there appears to be a valid print command in the file, lwarpmk displays this to instruct the user how to recompile the print version, which then updates the configuration files.

The designated source file does not exist: For whatever reason…

Unknown engine: lwarp cannot determing which engine is being used. Supported are DVI LaTeX, PDF LaTeX, XeLaTeX, LuaLaTeX, and upLaTeX.

13.1.2 Image generation with `lwarpmk limages`

"Wait a moment for the images to complete before reloading page.":

Images are generated by background tasks. If the document is reloaded before these tasks are complete, some images may not yet be generated. *lwarpmk* tries to wait for background tasks to complete before exiting.

HTML version does not exist: Images are extracted from the HTML version, which must be compiled before images are generated.

`*-images.txt` does not exist: This file tells which images to extract from the HTML file. If the file does not exist, it may be that no SVG math or `lateximages` were used. If so, **`lwarpmk limages`** is not necessary.

Cross references are not correct: The document must have up-to-date cross references to locate the images to extract. A number of conditions may cause incorrect cross references.

"WARNING: Images will be incorrect.": An image reference was not found. Recompile.

`lwarpmk epstopdf *` or `lwarpmk pdftosvg *`: Errors if filenames are not found.

13.1.3 Default bitmapped font

lwarp requires the use of a vector font. If lwarp detects that the document uses the default COMPUTER MODERN font, and the cm-super package is not installed, it is assumed that the font is bitmapped. An error is generated, along with the recommendation to install cm-super or use lmodern.

13.1.4 Packages

Loaded before lwarp: Some packages and classes must be loaded before lwarp. These include input and font encoding, morewrites and newclude, and a number of CJK-related packages and classes.

Loaded after lwarp: Most packages which are modified by lwarp must be loaded after lwarp.

Loaded never: Some packages do not work with lwarp. An error is generated, along with a list of alternatives to consider.

Specific packages: Some packages enforce a specific load order vs. certain other packages.

Patching error: lwarp tries to patch some packages using xpatch. If the original package has been updated more recently than lwarp, a patch may not work. It may be necessary to use an older version of the package until lwarp is updated.

longtable: lwarp's longtable package issues detailed error messages regarding the use of the table headers and footers.

polyglossia: If used, an informative message is printed to instruct the user to be sure to set a language, without which an error will occur.

babel or polyglossia: An informative message is printed to note that not all langauges are supported by cleveref.

13.1.5 Compiling

`SideTOCDepth` < `FileDepth`: A warning is displayed if these counters are set such that the sideTOC will not be able to access all pages of the website.

Duplicate filenames: lwarp may generate file names from section names. While doing so, the filenames are simplified and special characters are removed. If this process generates a duplicate filename, and error is generated, describing the filename and which section name generated it.

⚠ HTML corrupted **Multirow:** When `\multirow` or `\multicolrow` are used, `\mrowcell` or `\mcolrowcell` must be placed in the appropriate cells to avoid corrupted HTML output.

(width,height) missing a comma: `\makebox` and `\framebox` can accept a parenthesis-delimited width and height, which must be separated by a comma.

"Load graphicx or graphics for improved SVG math baselines.": SVG math baselines are improved if either of these packages are used.

"Load graphicx or graphics for improved XeTeX logo.": If these packages are loaded, the XeLaTeX logo can use the reversed "E".

"It is recommended to use `[width=xx\linewidth]` instead of `[scale=yy]` ": Browser support of `scale` does not have the same effect as in LaTeX.

13.2 Using the lwarp package

The following address problems which may occur, and possible solutions to each.

Also see:
Section 7.10: Commands to be placed into the `warpprint` environment
Section 8: Special cases and limitations

⚠ HTML corrupted **Text is not converting correctly / corrupted HTML tags:**

- Font-related UTF-8 information must be embedded in the PDF file. See section 7.4 regarding bitmapped vs. vector fonts.
- See section 8.2.1 regarding HTML entities and the characters &, <, and >.

Undefined HTML settings:

- See the warning regarding the placement of the HTML settings at section 7.6.

Tabular problems: See section 8.10.1.

Obscure error messages:

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

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

Options clash: If using memoir, see section 8.13.

"Missing `\begin{document}`.": Some packages require that their options be specified before lwarp is loaded, or via the package's setup macro, especially if these options include the use of braces. See section 8.1.

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

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

"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 8.6.9.

⚠ polyglossia

"! Undefined control sequence. . . . \@begindocumenthook":
See section 8.15.4 if using polyglossia.

⚠ custom macros for environments

"\begin{equation} ended by \end{document}": Do not use custom macros such as `\beq` and `\eeq` to replace

```
\begin{equation}
. . .
\end{equation}
```

⚠ `\LWR@formatted`

"Misplaced \omit": If using `\LWR@formatted` to define new macros for print and HTML modes, see section 36 regarding `\LWR@expandableformatted`.

⚠ "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 8.7.9.

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

Incorrect MATHJAX: Some objects do not convert to MATHJAX. Use `\displaymathother` before these objects, then `\displaymathnormal` to return to "normal" display math. See section 8.7.9.

Missing sections: See section 7.6 regarding the `FileDepth` and `SideTOCDepth` counters, and the use of `\tableofcontents` in the home page.

Misnumbered footnotes from section headings: See section 8.5.4.

Missing HTML files:

- See the warning regarding changes to the HTML settings at section 7.6.
- Ensure that the filenames are unique after math and short words are removed. See `FileSectionNames` at section 7.6.

Missing / incorrect cross-references:

- Use `lwarpmk again` followed by `lwarpmk html` or `lwarpmk print` to compile the document one more time.
- labels
  ⚠ label characters
  `\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 550 to redefine the message which is printed for page number references.

BibTeX errors with `\etalchar`: See section 8.6.9.

Malformed URLs: Do not use the `%` character between arguments of `\hyperref`, etc., as this character is among those which is neutralized for inclusion in HTML URLs.

Em-dashes or En-dashes in listing captions and titles:

Use 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.

recompile first

Before attempting to create the SVG image files, *lwarpmk* verifies that the HTML version of the document exists and has correct internal image references.[15] If it is necessary to recompile the document's HTML version one more time, *lwarpmk* usually will inform the user with an error message, but there are some conditions which cannot be detected, so the user should watch for the LaTeX recompile warnings.

⚠ HTML instead of images

If HTML appears where an SVG image should be, recompile the document one more time to get the page numbers back in sync, then remake the images one more time.

⚠ page counter

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

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

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

⚠ Lots of files!

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

Plain-looking document:

- The document's CSS stylesheet may not be available, or may be linked incorrectly. Verify any `\CSSFilename` statements point to a valid CSS file.

⚠ HTML corrupted

Broken fragments of HTML:

- Check the PDF file used to create HTML to see if the tags overflowed the margin. (This is why such large page size and margins are used.)

Changes do not seem to be taking effect:

- Be sure to `lwarpmk clean`, recompile, then start by reloading the home page. You may have been looking at an older version of the document. If you changed a section name, you may have been looking at the file for the old name.
- See the warning regarding changes to the HTML settings at section 7.6.
- Verify that the proper CSS is actually being used.
- The browser may compensate for some subtle changes, such as automatically generating ligatures, reflowing text, etc.

Un-matched conditional compiles:

- Verify the proper `begin/end` of `warpprint`, `warpHTML`, and `warpall` environments.

13.2.1 Debug tracing output

\tracinglwarp When \tracinglwarp is used, lwarp will add extra tracing messages to the .log file. The last several messages may help track down errors.

Place \tracinglwarp just after \usepackage{lwarp} to activate tracing.

13.3 Compiling the lwarp.dtx file

lwarp_tutorial.tex: Copy or link lwarp_tutorial.txt from the TDS doc directory to the source directory, or wherever you wish to compile the documentation. This file is included verbatim in the documentation, but is in the doc directory so that it may be found by *texdoc* and copied by the user.

Illogical error messages caused by an out-of-sync lwarp.sty file:

1. Delete the lwarp.sty file.
2. Enter **pdflatex lwarp.ins** to generate a new lwarp.sty file.
3. Enter **pdflatex lwarp.dtx** to recompile the lwarp.pdf documentation.

Un-nested environments:

Be sure to properly nest:

- \begin{macrocode} and \end{macrocode}
- \begin{macro} and \end{macro}
- \begin{environment} and \end{environment}

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

14 Trademarks

- TEX is a trademark of American Mathematical Society.
- ADOBE® and ADOBE *Framemaker*® are either registered trademarks or trademarks of ADOBE SYSTEMS INCORPORATED in the United States and/or other countries.
- LINUX® is the registered trademark of Linus Torvalds in the U.S. and other countries.
- MAC OS® is a trademark of APPLE INC.
- MADCAP FLARE™ is the property of MADCAP SOFTWARE, INC.
- MATHJAX is copyright 2009 and later. The MATHJAX CONSORTIUM is a joint venture of the AMERICAN MATHEMATICAL SOCIETY (AMS) and the SOCIETY FOR INDUSTRIAL AND APPLIED MATHEMATICS (SIAM) to advance mathematical and scientific content on the web.
- MICROSOFT®, ENCARTA, MSN, and WINDOWS® are either registered trademarks or trademarks of MICROSOFT CORPORATION in the United States and/or other countries.
- UNIX® is a registered trademark of THE OPEN GROUP.
- YOUTUBE™ is trademark of GOOGLE LLC.

File 1 **lwarp.sty**

15 Implementation

This package is perhaps best described as a large collection of smaller individual technical challenges, in many cases solved through a number of ~~crude hacks~~ clever tricks. Reference sources are given for many of the solutions, and a quick internet search will provide additional possibilities.

Judgement calls were made, and are often commented. Improvements are possible. The author is open to ideas and suggestions.

Packages were patched for re-use where they provided significant functionality. Examples include xcolor with its color models and conversion to HTML color output, and siunitx which provides many number and unit-formatting options, almost all of which are available in pure-text form, and thus easily used by *pdftotext*.

Packages were emulated where their primary purpose was visual formatting which is not relevent to HTML output. For example, packages related to sectioning are already patched by numerous other packages, creating a difficult number of combinations to try to support, and yet in HTML output all of the formatting is thrown away, so these packages are merely emulated.

Packages with graphical output are allowed as-is, but must be nested inside a `latexmage` 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	`<div class = "book">`
part	-1	`<h2>`
chapter	0	`<h3>`
section	1	`<h4>`
subsection	2	`<h5>`
subsubsection	3	`<h6>`
paragraph	4	`<span class = "paragraph">`
subparagraph	5	`<span class = "subparagraph">`
listitem	7	new for this package, used for list items

* If `FormatWP` is true, section headings may be adjusted, depending on `WPTitleHeading`. See table 8 on page 177.

16 Section depths and HTML headings

Stacks are created to track depth inside the LaTeX document structure. This depth is translated to HTML headings as shown in table 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.

17 Source code

This is where the documented source code for lwarp begins, continuing through the following sections all the way to the change log and index at the end of this document.

The following sections document the actual implementation of the lwarp package.

line numbers — The small numbers at the left end of a line refer to line numbers in the `lwarp.sty` file.

subjects — Blue-colored tags in the left margin aid in quickly identifying the subject of each paragraph.

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 —

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

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

Detects XƎTEX and LuaLATEX:

```
\RequirePackage{iftex}[2019/11/07]

\newif\ifxetexorluatex

\ifXeTeX
    \xetexorluatextrue
\else
    \ifLuaTeX
        \xetexorluatextrue
    \else
        \xetexorluatexfalse
    \fi
\fi
```

19 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}
```

20 Package load order

Several packages must never be used with lwarp, others should only be loaded before lwarp, and others should only be loaded after. The lwarp core checks most of these cases. In some lwarp-* packages, \LWR@loadbefore is used to trigger an error if they are loaded after lwarp, while additional code provides necessary patches for when they are loaded before.

Packages which must be loaded after lwarp are enfoced by a large number of \LWR@loadafter statements, below. Some packages are emulated by memoir, and

so these are tested by \LWR@notmemoirloadafter, which does not cause an error if memoir is used.

\LWR@checkloadfilename is used to check each filename to see if it must never be loaded, or must always be loaded before lwarp.

20.1 Tests of package load order

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

```
\newcommand*{\LWR@loadafter}[1]{%
\@ifpackageloaded{#1}
{
    \PackageError{lwarp}
        {%
            Package #1,\MessageBreak
            or one which uses #1,\MessageBreak
            must be loaded after lwarp
        }
        {Move \detokenize{\usepackage}{#1} after
        \detokenize{\usepackage}{lwarp}.\MessageBreak
        Package #1 may also be loaded by something else,\MessageBreak
        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\MessageBreak
    by lwarp's HTML conversion.\MessageBreak
    Package(s) #2 may be useful instead%
}
{%
    Package #1 might conflict with lwarp in some way,\MessageBreak
    or is superceded by another package.\MessageBreak
    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}{%
        \LWR@loadnever{#1}{#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\MessageBreak
    by lwarp's HTML conversion.\MessageBreak
    Class(es) #2 may be useful instead
}
{%
    Class #1 might conflict with lwarp in some way,\MessageBreak
    or is superceded by another class.\MessageBreak
    For a possible alternative, see class(es) #2.
}
}{}%
}
```

20.2 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{alg}{algorithm2e, algorithmicx}
\LWR@earlyloadnever{algorithmic}{algorithm2e, algorithmicx}
\LWR@earlyloadnever{bitfield}{bytefield}
\LWR@earlyloadnever{boxedminipage}{boxedminipage2e}
\LWR@earlyloadnever{caption2}{caption}
% \LWR@earlyloadnever{ccaption}{caption}% might be preloaded by memoir
\LWR@earlyloadnever{colortab}{colortbl}
\LWR@earlyloadnever{doublespace}{setspace}
\LWR@earlyloadnever{epsf}{graphicx}
\LWR@earlyloadnever{fancyheadings}{fancyhdr}
\LWR@earlyloadnever{fncylab}{cleveref}
\LWR@earlyloadnever{glossary}{glossaries}
```

```
\LWR@earlyloadnever{hyper}{hyperref}
\LWR@earlyloadnever{newthm}{ntheorem}
\LWR@earlyloadnever{pdfcprot}{microtype}
\LWR@earlyloadnever{picinpar}{floatflt, wrapfig}
\LWR@earlyloadnever{picins}{floatflt, wrapfig}
\LWR@earlyloadnever{rplain}{fancyhdr}
\LWR@earlyloadnever{si}{siunitx}
\LWR@earlyloadnever{sistyle}{siunitx}
\LWR@earlyloadnever{t1enc}{fontenc, inputenc, inputenx}
\LWR@earlyloadnever{ucs}{inputenc, inputencx}
\LWR@earlyloadnever{wasysym}{textcomp, amssymb, amsfonts, mnsymbol, fdsymbol}
```

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}
```

20.3 Enforcing package loading after lwarp

Packages which should only be loaded after lwarp are tested here to trip an error of they have already been loaded.

The following packages must be loaded after lwarp:

```
\LWR@loadafter{2in1}
\LWR@loadafter{2up}
\LWR@loadafter{a4}
\LWR@loadafter{a4wide}
\LWR@loadafter{a5comb}
\LWR@notmemoirloadafter{abstract}
\LWR@loadafter{academicons}
\LWR@loadafter{accessibility}
\LWR@loadafter{accsupp}
\LWR@loadafter{acro}
\LWR@loadafter{acronym}
\LWR@loadafter{adjmulticol}
\LWR@loadafter{addlines}
\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{autonum}
\LWR@loadafter{awesomebox}
\LWR@loadafter{axessibility}
\LWR@loadafter{axodraw2}
\LWR@loadafter{backnaur}
\LWR@loadafter{backref}
\LWR@loadafter{balance}
\LWR@loadafter{bbding}
\LWR@loadafter{bigdelim}
\LWR@loadafter{bigfoot}
\LWR@loadafter{bigstrut}
\LWR@loadafter{bitpattern}
\LWR@loadafter{blowup}
\LWR@loadafter{bm}
\LWR@loadafter{booklet}
\LWR@loadafter{bookmark}
\LWR@notmemoirloadafter{booktabs}
\LWR@loadafter{bophook}
\LWR@loadafter{bounddvi}
\LWR@loadafter{boxedminipage2e}
\LWR@loadafter{braket}
\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@loadafter{changelayout}
\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{ctable}
\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{eqlist}
\LWR@loadafter{eqparbox}
\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{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{fnpara}
\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{footnotebackref}
\LWR@loadafter{footnotehyper}
\LWR@loadafter{footnoterange}
\LWR@loadafter{footnpag}
\LWR@loadafter{foreign}
\LWR@loadafter{forest}
\LWR@loadafter{framed}
\LWR@loadafter{ftcap}
\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{hypbmsec}
\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
% inputtrc may be loaded before lwarp
\LWR@loadafter{intopdf}
\LWR@loadafter{karnaugh-map}
\LWR@loadafter{keyfloat}
\LWR@loadafter{layaureo}
\LWR@loadafter{layout}
\LWR@loadafter{layouts}
\LWR@loadafter{leading}
\LWR@loadafter{letterspace}
\LWR@loadafter{lettrine}
\LWR@loadafter{lineno}
\LWR@loadafter{lips}
\LWR@loadafter{listings}
\LWR@loadafter{listliketab}
\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{lyluatex}
\LWR@loadafter{magaz}
\LWR@notmemoirloadafter{makeidx}
\LWR@loadafter{manyfoot}
\LWR@loadafter{marginfit}
\LWR@loadafter{marginfix}
\LWR@loadafter{marginnote}
\LWR@loadafter{marvosym}
\LWR@loadafter{mathtools}
\LWR@loadafter{mcaption}
\LWR@loadafter{mdframed}
\LWR@loadafter{media9}
\LWR@loadafter{memhfixc}
\LWR@loadafter{metalogo}
\LWR@loadafter{metalogox}
\LWR@loadafter{mhchem}
\LWR@loadafter{microtype}
\LWR@loadafter{midfloat}
\LWR@loadafter{midpage}
\LWR@loadafter{minibox}
\LWR@loadafter{minitoc}
% morefloats must be allowed early for print mode
\LWR@notmemoirloadafter{moreverb}
% morewrites must be loaded before lwarp
\LWR@notmemoirloadafter{movie15}
\LWR@notmemoirloadafter{mparhack}
\LWR@loadafter{multicap}
%\LWR@loadafter{multicol}% loaded by ltxdoc
```

```
\LWR@loadafter{multicolrule}
\LWR@loadafter{multimedia}
\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{niceframe}
\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{parallel}
\LWR@loadafter{parcolumns}
\LWR@loadafter{parnotes}
\LWR@notmemoirloadafter{parskip}
\LWR@loadafter{pbox}
\LWR@loadafter{pdfcol}
\LWR@loadafter{pdfcolfoot}
\LWR@loadafter{pdfcolmk}
\LWR@loadafter{pdfcolparallel}
\LWR@loadafter{pdfcolparcolumns}
\LWR@loadafter{pdfcomment}
\LWR@loadafter{pdfcrypt}
\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{perpage}
\LWR@loadafter{pfnote}
\LWR@loadafter{phfqit}
\LWR@loadafter{physics}
\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@loadafter{shapepar}
\LWR@notmemoirloadafter{showidx}
\LWR@loadafter{showkeys}
\LWR@loadafter{showtags}
\LWR@loadafter{sidecap}
\LWR@loadafter{sidenotes}
```

```
\LWR@loadafter{SIunits}
\LWR@loadafter{siunitx}
\LWR@loadafter{slantsc}
\LWR@loadafter{slashed}
\LWR@loadafter{soul}
\LWR@loadafter{soulpos}
\LWR@loadafter{soulutf8}
\LWR@loadafter{splitidx}
\LWR@loadafter{srcltx}
\LWR@loadafter{srctex}
\LWR@loadafter{stabular}
\LWR@loadafter{stackengine}
\LWR@loadafter{steinmetz}
\LWR@notltjloadafter{stfloats}
\LWR@loadafter{struktex}
\LWR@loadafter{subcaption}
\LWR@loadafter{subfig}
\LWR@loadafter{subfigure}
\LWR@loadafter{supertabular}
\LWR@loadafter{svg}
\LWR@loadafter{t1inc}
\LWR@loadafter{tabfigures}
\LWR@loadafter{tabls}
\LWR@loadafter{tablefootnote}
\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{threeparttablex}
\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{tocdata}
\LWR@loadafter{tocenter}
\LWR@notmemoirloadafter{tocloft}
\LWR@loadafter{tocstyle}
\LWR@loadafter{todo}
\LWR@loadafter{todonotes}
\LWR@loadafter{topcapt}
\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{unicode-math}
\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{widetable}
\LWR@loadafter{widows-and-orphans}
\LWR@loadafter{witharrows}
\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{xr}
\LWR@loadafter{xr-hyper}
\LWR@loadafter{xtab}
% xunicode must be loaded before lwarp
\LWR@loadafter{xurl}
\LWR@loadafter{xy}
\LWR@loadafter{zwpagelayout}
```

21 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
```

22 pdfLaTeX T1 and UTF-8 encoding

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

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

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
```

23 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}% Here, the first arguments are ligatures.
\newunicodechar{fi}{fi}
\newunicodechar{fl}{fl}
\newunicodechar{ffi}{ffi}
\newunicodechar{ffl}{ffl}
\newunicodechar{—}{---}
\newunicodechar{–}{--}
\fi

\fi
```

24 Avoid a bitmapped font

If DVI or PDF LaTeX, and if the default Computer Modern is the selected font family, ensure that cm-super or lmodern is used to provide a vector font.

```
\ifxetexorluatex
\else
    \ifdefstring{\f@family}{cmr}{
        \IfFileExists{type1ec.sty}% found in cm-super
        {}
        {% cm-super not installed
            \IfFileExists{lmodern.sty}{
```

```
                \PackageInfo{lwarp}{cm-super not installed, loading lmodern}
                \RequirePackage{lmodern}
            }{
                \PackageError{lwarp}
                {%
                    Lwarp requires a vector font.\MessageBreak
                   Install and load cm-super, lmodern, or another\MessageBreak
                    Type-1 vector font before loading lwarp%
                }
                {%
                    Install cm-super or lmodern.\MessageBreak
                    If lmodern, load it before lwarp:\MessageBreak
                        \space\space\protect\usepackage{lmodern}\MessageBreak
                        \space\space\protect\usepackage{lwarp}%
                }
            }
        }% cm-super not installed
    }{}% f@family
\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

26.1 Lengths and units

`\LWR@providelength` {⟨\lengthname⟩} Provides the length if it isn't defined yet.

Used to provide source compatibility for lengths which will be ignored, but might or might not be already provided by other packages.

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

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

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

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

\LWR@printpercentlength {⟨*smaller*⟩} {⟨*larger*⟩}

Prints a percent ratio of the two lengths.

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

26.2 Counters

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

Locally add to a counter.

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

26.3 Patching

\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.}
}
```

26.4 Chinese text isolation

\LWR@isolate {⟨*text*⟩} Isolates Chinese characters from the surrounding text. This is required to avoid extra spaces on either side of the Chinese characters, especially when written to a file.

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

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

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

26.5 Inserting vertical space

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

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

26.6 Argument selection

\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}
```

26.7 Global boxes

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

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

```
\DeclareRobustCommand\LWR@gsavebox[1]{%
  \@ifnextchar(%)
    {\LWR@@gsavepicbox#1}{\@ifnextchar[{\LWR@@gsavebox#1}{\LWR@gsbox#1}}}%
\long\def\LWR@gsbox#1#2{\global\setbox#1\hbox{%
  \color@setgroup#2\color@endgroup}}
\def\LWR@@gsavebox#1[#2]{%
  \@ifnextchar [{\LWR@@igsavebox#1[#2]}{\LWR@@igsavebox#1[#2][c]}}
\long\def\LWR@@igsavebox#1[#2][#3]#4{%
  \LWR@gsbox#1{\@imakebox[#2][#3]{#4}}}
\def\LWR@@gsavepicbox#1(#2,#3){%
  \@ifnextchar[%]
    {\LWR@@igsavepicbox#1(#2,#3)}{\LWR@@igsavepicbox#1(#2,#3)[]}}
\long\def\LWR@@igsavepicbox#1(#2,#3)[#4]#5{%
  \LWR@gsbox#1{\@imakepicbox(#2,#3)[#4]{#5}}}
```

Env LWR@glrbox {⟨*macroname*⟩}

```
\def\LWR@glrbox#1{%
  \edef\reserved@a{%
    \endgroup
    \global\setbox#1\hbox{%
      \begingroup\aftergroup}%
        \def\noexpand\@currenvir{\@currenvir}%
        \def\noexpand\@currenvline{\on@line}}%
  \reserved@a
    \@endpefalse
    \color@setgroup
      \ignorespaces}
\let\LWR@endglrbox\LWR@endlrbox
```

26.8 Converting a macro name to a cs name

\macrotocsname {⟨*macro name with backslash*⟩}

Results in the macro name without the leading backslash.

Ref: https://tex.stackexchange.com/questions/42318/removing-a-backslash-from-a-character-sequen

```
\newcommand*{\macrotocsname}[1]{%
  \ifcat\relax\noexpand#1%
    \expandafter\expandafter\expandafter\@gobble\expandafter\string
  \fi
  #1%
}
```

27 Operating-System portability

Prog Unix
Prog Mac OS
Prog Linux

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.

Prog MS-Windows

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

Prog Windows
Opt 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}%
```

```
807 }
```

Opt mathjax

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

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

812 }
```

Opt BaseJobname
Default: \jobname

Option BaseJobname sets the \BaseJobname for this document.

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.

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

Opt ImagesDirectory
Default: \jobname-images

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

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

Opt ImagesName
Default: image-

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

```
815 \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.

```
816 \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 "}")
```

```
817 \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.

```
818 \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.

```
819 \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 warpHTML For HTML output:

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

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

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

Env warpMathJax Only if MathJax is being used along with HTML.

```
\begin{warpprint}
\excludecomment{warpMathJax}
\end{warpprint}

\begin{warpHTML}
\ifbool{mathjax}
    {\includecomment{warpMathJax}}
    {\excludecomment{warpMathJax}}
\end{warpHTML}
```

Env LWR@creatinglwarpmk 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}
```

If geometry has not yet been loaded, use the preexising page and text sizes to be preserved for later reuse. These will be replaced by lwarp `\AtBeginDocument` with a very large page size to reduce HTML tag overflow off the page.

```
\@ifpackageloaded{geometry}
{}{
    \RequirePackage[
        reset,
        paperwidth=\paperwidth,
        paperheight=\paperheight,
        textwidth=\textwidth,
        textheight=\textheight,
        left=\oddsidemargin,
        top=\topmargin,
        marginparsep=\marginparsep,
        marginparwidth=\marginparwidth,
    ]{geometry}
}
```

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

```
\LetLtxMacro\LWR@origgeometry\geometry
\LetLtxMacro\LWR@orignewgeometry\newgeometry
\LetLtxMacro\LWR@origrestoregeometry\restoregeometry
\LetLtxMacro\LWR@origsavegeometry\savegeometry
\LetLtxMacro\LWR@origloadgeometry\loadgeometry
```

Bool `LWR@allowanothergeometry`

geometry may be loaded by the user before lwarp, after lwarp, or not at all. If before lwarp, it will have already been loaded by now and its page layout has already been saved. If geometry is loaded after lwarp, its layout will be set at that time and the user macros nullified. `\AtEndPreamble` this layout will be saved. If the user never loads geometry, lwarp-geometry will be loaded `\AtBeginDocument`, but it should not change the page layout set here. This is controlled by the boolean `LWR@allowanothergeometry`. Geometry may be adjusted throughout the preamble until `\AtEndPreamble`, when this boolean is set `false`.

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

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

```
\AtEndPreamble{
```

Whatever geometry choices the user has made in the preamble, either before or after lwarp was loaded, are now saved for possible temporary reuse, such as by lyluatex.

See the lwarp-geometry section for what happens if geometry is loaded after lwarp.

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

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

```
\newlength{\LWR@userspaperwidth}
\setlength{\LWR@userspaperwidth}{\paperwidth}

\newlength{\LWR@userspaperheight}
\setlength{\LWR@userspaperheight}{\paperheight}

\newlength{\LWR@usersmarginparwidth}
\setlength{\LWR@usersmarginparwidth}{\marginparwidth}

\newlength{\LWR@userstextwidth}
\setlength{\LWR@userstextwidth}{\textwidth}

\newlength{\LWR@userstextheight}
\setlength{\LWR@userstextwidth}{\textheight}
```

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

```
\LWR@origgeometry{
    reset,
    paperheight=190in,
    paperwidth=20in,
    left=2in,
    right=6in,
    top=1in,
    bottom=1in,
    heightrounded,%
}
```

The lwarp page geometry is saved for future restore:

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

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

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

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

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

\@twosidefalse
\@mparswitchfalse
}% \AtEndPreamble

\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 while creating the print version.

Recent versions of LaTeX (as of Fall 2019) now include the functionality of the filecontents package, but with a new optional argument used to specify whether to force the overwriting of an existing file. If an older LaTeX kernel is used, the original filecontents package is used, but it is patched to throw away the new optional argument.

```
\@ifundefined{filec@ntents@opt}{% older kernel, discard optional args

    \RequirePackage{filecontents}

```

```
    \LetLtxMacro\LWR@orig@filec@ntents\filec@ntents

    \@ifpackagelater{filecontents}{2011/10/08}
    {
```

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

```
        \renewcommand*{\filec@ntents}[1][]{\LWR@orig@filec@ntents}
    }
    {% patch older package for morewrites
```

For an older version of filecontents, discard the optional argument, and also patch to work with morewrites, per https://tex.stackexchange.com/questions/312830/does-morewrites-not-support-filecontents-and-can-i-write-body-of-environment-us/312910

```
        \newwrite\fcwrite
        \renewcommand*{\filec@ntents}[1][]{%
            \def\chardef##1\write{\let\reserved@c\fcwrite}%
            \LWR@orig@filec@ntents%
        }
    }

}% older kernel
{% newer kernel
```

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

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

\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
\LetLtxMacro\LWR@origRequirePackageWithOptions\RequirePackageWithOptions
```

\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}{cm-super, lmodern}
\LWR@checkloadnever{#1}{aecc}{cm-super, lmodern}
\LWR@checkloadnever{#1}{aecompl}{cm-super, lmodern}
\LWR@checkloadnever{#1}{alg}{algorithm2e, algorithmicx}
\LWR@checkloadnever{#1}{algorithmic}{algorithm2e, algorithmicx}
\LWR@checkloadnever{#1}{bitfield}{bytefield}
\LWR@checkloadnever{#1}{boxedminipage}{boxedminipage2e}
\LWR@checkloadnever{#1}{caption2}{caption}
\LWR@checkloadnever{#1}{ccaption}{caption}% might be preloaded by memoir
```

```
\LWR@checkloadnever{#1}{colortab}{colortbl}
\LWR@checkloadnever{#1}{doublespace}{setspace}
\LWR@checkloadnever{#1}{epsf}{graphicx}
\LWR@checkloadnever{#1}{fancyheadings}{fancyhdr}
\LWR@checkloadnever{#1}{fncylab}{cleveref}
\LWR@checkloadnever{#1}{glossary}{glossaries}
\LWR@checkloadnever{#1}{hyper}{hyperref}
\LWR@checkloadnever{#1}{newthm}{ntheorem}
\LWR@checkloadnever{#1}{pdfcprot}{microtype}
\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}
\LWR@checkloadnever{#1}{bxcjkjatype}{upLaTeX, bxjsarticle, ujarticle, utarticle}
\LWR@checkloadnever{#1}{hangul}{kotex, xetexko, luatexko}
```

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}{nfssext-cfr}
\LWR@checkloadbefore{#1}{fontaxes}
\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}{}%
    \ifdefstring{\LWR@strresulttwo}{changes}{\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]}%
        {\LWR@origRequirePackage{\LWR@parsedrequirepackagenames}}%
}%
}
\LetLtxMacro\usepackage\RequirePackage
```

```
\end{warpall}
```

for HTML output:

```
\begin{warpHTML}
```

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

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

Uses the original package, including options.

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

If using catoptions, an error occurs if a package is loaded with an option then loaded again with no options. lwarp does this if a package is preloaded then later patched. To avoid an error while using catoptions, if a package has already been loaded, it is loaded again with its original options.

```
    \@ifpackageloaded{#1}{%
        \edef\LWR@tempone{\csuse{opt@#1.sty}}%
        \IfValueTF{#2}%
            {%
                \expandafter\LWR@origRequirePackage%
                    \expandafter[\LWR@tempone]{#1}[#2]%
            }%
            {%
                \expandafter\LWR@origRequirePackage%
                    \expandafter[\LWR@tempone]{#1}%
            }%
    }{%
        \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@ProvidesPackageDropA` {⟨*name*⟩} {⟨*date or -NoValue-*⟩}

Declares the package. Factored for reuse.

```
\newcommand*{\LWR@ProvidesPackageDropA}[2]{%
    \PackageInfo{lwarp}{%
        Replacing package ‘#1’ with the lwarp version,\MessageBreak
        and discarding options,%
    }%
    \IfValueTF{#2}
    {\ProvidesPackage{lwarp-#1}[#2]}
```

```
1249     {\ProvidesPackage{lwarp-#1}}
1250 }
```

\LWR@ProvidesPackageDropB Nullifies then processes the options.

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

```
1251 \newcommand*{\LWR@ProvidesPackageDropB}{%
1252 % \ProcessOptions\relax% original LaTeX code
1253 \let\ds@\@empty%         from the original \ProcessOptions
1254 \edef\@curroptions{}%    lwarp modification to \ProcessOptions
1255 \@process@ptions\relax%  from the original \ProcessOptions
1256 }
```

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

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

```
1257 \NewDocumentCommand{\LWR@ProvidesPackageDrop}{m o}{
```

Declare the package:

```
1258 \LWR@ProvidesPackageDropA{#1}{#2}
```

Ignore all options:

```
1259 \DeclareOption*{}
```

Process the options:

```
1260 \LWR@ProvidesPackageDropB
1261 }
```

```
1262 \end{warpHTML}
```

31 Additional required packages

for HTML output:

```
1263 \begin{warpHTML}
```

Pkg caption

```
1264 \LWR@origRequirePackage{caption}
1265 \AtBeginDocument{\RequirePackage{lwarp-caption}}%
```

```
1266 \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*⟩} Modified to load `_html_inc` files.

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

```
\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@orignewpage% changed from clearpage
\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@orignewpage% changed from clearpage
    \@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 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

Default: false

Add comments in HTML about closing s, sections, etc.

```
\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` Changes HTML output for easy EPUB conversion via an external program. Removes per-file headers, footers, and nav. Adds footnotes per chapter/section.
Default: `false`

```
\newbool{FormatEPUB}
\boolfalse{FormatEPUB}
```

Bool `FormatWP` 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.
Default: `false`

```
\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.

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.

[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 lists are printed instead.*

```
\newbool{WPMarkLOFT}
\boolfalse{WPMarkLOFT}
```

Bool `WPMarkMath` While formatting for word processors, prints math as LaTeX code instead of creating
Default: `false` SVG images or MathJax. This is useful for cut/paste into the *LibreOffice Writer TeXMaths* extension.

```
\newbool{WPMarkMath}
\boolfalse{WPMarkMath}
```

Bool `WPTitleHeading` While formatting for word processors, `true` sets the document title to `<h1>`, which
Default: `false` is expected for HTML documents, but also causes the lower-level section headings to start at **Heading 2** when imported into LibreOffice. Set to `false` to cause the title to be plain text, and the section headings to begin at **Heading 1**.

See table 8 on page 177.

```
\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@origttfamily\ttfamily

\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


\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 latexmage.

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 `latexi``mage` 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}
    {
        \PackageInfo{lwarp}{%
            Jobname with -pstool is found.\MessageBreak
            Not generating lwarp configuration files,%
        }
    }
    {
        \PackageInfo{lwarp}{Generating lwarp 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*}[overwrite]{lwarp.css}
/*
  CSS stylesheet for the LaTeX Lwarp package
  Copyright 2016-2020 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.tocbook {display: inline-block ; margin-left: 0em ;
    font-weight: bold ; margin-top: 1ex ; margin-bottom: 1ex ; }
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; }
```

```
1904
1905 span.textmd, div.textmd { font-weight: normal; }
1906
1907 span.texteb { font-weight: bolder; }
1908
1909 span.textlg { font-weight: lighter; }
1910
1911 span.textup, div.textup {
1912     font-style: normal;
1913     font-variant: normal;
1914     font-variant-numeric: normal ;
1915 }
1916
1917 span.textsc, div.textsc {
1918     font-variant: small-caps;
1919     font-variant-numeric: oldstyle-nums ;
1920 }
1921
1922 span.textulc, div.textulc {
1923     font-variant: normal ;
1924     font-variant-numeric: normal ;
1925 }
1926
1927 span.textsl, div.textsl { font-style: oblique; }
1928
1929 span.textrm, div.textrm {
1930     font-family: "DejaVu Serif", "Bitstream Vera Serif",
1931     "Lucida Bright", Georgia, serif;
1932 }
1933
1934 span.textsf, div.textsf {
1935      font-family: "DejaVu Sans", "Bitstream Vera Sans",
1936         Geneva, Verdana, sans-serif ;
1937 }
1938
1939 span.textcircled { border: 1px solid black ; border-radius: 1ex ; }
1940
1941 span.underline {
1942     text-decoration: underline ;
1943     text-decoration-skip: auto ;
1944 }
1945
1946 span.overline {
1947     text-decoration: overline ;
1948     text-decoration-skip: auto ;
1949 }
1950
1951
1952 /* for vertical text: */
1953 div.verticalrl { writing-mode: vertical-rl }
1954 div.horizontaltb { writing-mode: horizontal-tb }
1955
1956
1957 /* for diagbox */
1958 div.diagboxtitleN { border-bottom: 1px solid gray }
```

```
1959 div.diagboxtitleS { border-top: 1px solid gray }
1960
1961 div.diagboxE {
1962     padding-left: 2em ;
1963     text-align: right ;
1964 }
1965
1966 div.diagboxW {
1967     padding-right: 2em ;
1968     text-align: left ;
1969 }
1970
1971
1972
1973 /* For realscripts */
1974 .supsubscript {
1975     display: inline-block;
1976     text-align:left ;
1977 }
1978
1979 .supsubscript sup,
1980 .supsubscript sub {
1981     position: relative;
1982     display: block;
1983     font-size: .5em;
1984     line-height: 1;
1985 }
1986
1987 .supsubscript sup {
1988     top: .5em;
1989 }
1990
1991 .supsubscript sub {
1992     top: .5em;
1993 }
1994
1995 div.attribution p {
1996     text-align: right ;
1997     font-size: 80%
1998 }
1999
2000 span.poemtitle {
2001   font-size: 120% ; font-weight: bold;
2002 }
2003
2004 pre.tabbing {
2005     font-family: "Linux Libertine Mono O", "Lucida Console",
2006         "Droid Sans Mono", "DejaVu Mono", "Bitstream Vera Mono",
2007         "Liberation Mono", "FreeMono", "Andale Mono",
2008         "Nimbus Mono L", "Courier New", monospace;
2009 }
2010
2011 blockquote {
2012     display: block ;
2013     margin-left: 2em ;
```

```
2014     margin-right: 2em ;
2015 }
2016
2017 /* quotchap is for the quotchap package */
2018 div.quotchap {
2019     display: block ;
2020     font-style: oblique ;
2021     overflow-x: auto ;
2022     margin-left: 2em ;
2023     margin-right: 2em ;
2024 }
2025
2026 blockquote p, div.quotchap p {
2027     line-height: 1.5;
2028     text-align: left ;
2029     font-size: .85em ;
2030 }
2031
2032 /* qauthor is for the quotchap package */
2033 div.qauthor {
2034   display: block ;
2035   text-align: right ;
2036   margin-left: auto ;
2037   margin-right: 2em ;
2038   font-size: 80% ;
2039   font-variant: small-caps;
2040 }
2041
2042 div.qauthor p {
2043   text-align: right ;
2044 }
2045
2046 div.epigraph, div.dictum {
2047   line-height: 1.2;
2048     text-align: left ;
2049     padding: 3ex 1em 0ex 1em ;
2050 /*     margin: 3ex auto 3ex auto ; */ /* Epigraph centered */
2051     margin: 3ex 1em 3ex auto ; /* Epigraph to the right */
2052 /*    margin: 3ex 1em 3ex 1em ; */ /* Epigraph to the left */
2053     font-size: .85em ;
2054     max-width: 27em ;
2055 }
2056
2057 div.epigraphsource, div.dictumauthor {
2058     text-align:right ;
2059     margin-left:auto ;
2060 /*     max-width: 50% ; */
2061     border-top: 1px solid #A0A0A0 ;
2062     padding-bottom: 3ex ;
2063     line-height: 1.2;
2064 }
2065
2066 div.epigraph p, div.dictum p { padding: .5ex ; margin: 0ex ;}
2067 div.epigraphsource p, div.dictumauthor p { padding: .5ex 0ex 0ex 0ex ; margin: 0ex ;}
2068 div.dictumauthor { font-style:italic }
```

```
2069
2070
2071 /* copyrightbox package */
2072 div.copyrightbox { margin: .5ex .5em }
2073 div.copyrightbox p {margin: 0px .5em ; padding: 0px}
2074 div.copyrightboxnote {text-align: left ; font-size: 60%}
2075
2076
2077 /* lettrine package: */
2078 span.lettrine { font-size: 4ex ; float: left ; }
2079 span.lettrinetext { font-variant: small-caps ; }
2080
2081 /* ulem, soul, umoline packages: */
2082 span.uline {
2083     text-decoration: underline ;
2084     text-decoration-skip: auto ;
2085 }
2086
2087 span.uuline {
2088     text-decoration: underline ;
2089     text-decoration-skip: auto ;
2090     text-decoration-style: double ;
2091 }
2092
2093 span.uwave {
2094     text-decoration: underline ;
2095     text-decoration-skip: auto ;
2096     text-decoration-style: wavy ;
2097 }
2098
2099 span.sout {
2100     text-decoration: line-through ;
2101 }
2102
2103 span.oline {
2104     text-decoration: overline ;
2105     text-decoration-skip: auto ;
2106 }
2107
2108 span.xout {
2109     text-decoration: line-through ;
2110 }
2111
2112 span.dashuline {
2113     text-decoration: underline ;
2114     text-decoration-skip: auto ;
2115     text-decoration-style: dashed ;
2116 }
2117
2118 span.dotuline {
2119     text-decoration: underline ;
2120     text-decoration-skip: auto ;
2121     text-decoration-style: dotted ;
2122 }
2123
```

```
span.letterspacing { letter-spacing: .2ex ; }

span.capsspacing {
    font-variant: small-caps ;
    letter-spacing: .1ex ;
}

span.highlight { background: #F8E800 ; }




html body {
  margin: 0 ;
  line-height: 1.2;
}


body div {
  margin: 0ex;
}


div.book, 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,
        "Times New Roman", serif;
    font-style: normal ;
    font-weight: bold ;
    text-align: left ;
}

h1 {    /* title of the entire website, used on each page */
    text-align: center ;
    font-size: 2.5em ;
    padding: .4ex 0em 0ex 0em ;
}

div.book {
    text-align: center ;
    font-size: 2.325em ;
    padding: .4ex 0em 0ex 0em ;
}

h2 { font-size: 2.25em }
h3 { font-size: 2em }
h4 { font-size: 1.75em }
h5 { font-size: 1.5em }
h6 { font-size: 1.25em }
span.paragraph {font-size: 1em ; font-variant: normal ;
    margin-right: 1em ; }
span.subparagraph {font-size: 1em ; font-variant: normal ;
    margin-right: 1em ;  }
```

```
2179
2180 div.minisec {
2181     font-family: "DejaVu Sans", "Bitstream Vera Sans",
2182         Geneva, Verdana, sans-serif ;
2183     font-style: normal ;
2184     font-weight: bold ;
2185     text-align: left ;
2186 }
2187
2188 h1 {
2189   margin: 0ex 0em 0ex 0em ;
2190   line-height: 1.3;
2191   text-align: center ;
2192 }
2193
2194 h2 {
2195   margin: 1ex 0em 1ex 0em ;
2196   line-height: 1.3;
2197   text-align: center ;
2198 }
2199
2200 h3 {
2201   margin: 3ex 0em 1ex 0em ;
2202   line-height: 1.3;
2203 }
2204
2205 h4 {
2206   margin: 3ex 0em 1ex 0em ;
2207   line-height: 1.3;
2208 }
2209
2210 h5 {
2211   margin: 3ex 0em 1ex 0em ;
2212   line-height: 1.3;
2213 }
2214
2215 h6 {
2216   margin: 3ex 0em 1ex 0em ;
2217   line-height: 1.3;
2218 }
2219
2220
2221 div.titlepage {
2222   text-align: center ;
2223 }
2224
2225 .footnotes {
2226     text-align: left ;
2227     font-size: .85em ;
2228     margin: 3ex 2em 0ex 2em ;
2229     border-top: 1px solid silver ;
2230 }
2231
2232 .marginpar, .marginparblock {
2233     max-width: 50%;
```

```
    float: right ;
    clear: both ;
    text-align: left ;
    margin: 1ex 0.5em 1ex 1em ;
    padding: 1ex 0.5em 1ex 0.5em ;
    font-size: 85% ;
    border-top: 1px solid silver ;
    border-bottom: 1px solid silver ;
    overflow-x: auto ;
}

.marginpar br { margin-bottom: 2ex ; }

div.marginblock, div.marginparblock {
    max-width:50%;
    min-width: 10em; /* room for caption */
    float:right;
    text-align:left;
    margin: 1ex 0.5em 1ex 1em ;
    padding: 1ex 0.5em 1ex 0.5em ;
    overflow-x: auto;
}

div.marginblock div.minipage,
div.marginparblock div.minipage {
    display: inline-block ;
    margin: 0pt auto 0pt auto ;
}

div.marginblock div.minipage p ,
div.marginparblock div.minipage p
    { font-size: 85%}

div.marginblock br ,
div.marginparblock br
    { margin-bottom: 2ex ; }

div.bodycontainer {
    float: left ;
    width: 80% ;
}

div.bodywithoutsidetoc div.bodycontainer {
    float: none ;
    width: 100% ;
}

section.textbody div.footnotes{
    margin: 3ex 2em 0ex 2em ;
    border-bottom: 2px solid silver ;
}

.footnoteheader {
    border-top: 2px solid silver ;
    margin-top: 3ex ;
```

```
    padding-top: 1ex ;
    font-weight: bold ;
}

.mpfootnotes {
    text-align: left ;
    font-size: .85em ;
    margin-left: 1em ;
    border-top: 1px solid silver ;
}

/* Remove footnote top border in the title page. */
div.titlepage div.mpfootnotes {
    border-top: none ;
}



ul, ol {
  margin: 1ex 1em 1ex 0em;
  line-height: 1.2;
}

body dir, body menu {
  margin: 3ex 1em 3ex 0em;
  line-height: 1.2;
}

li { margin: 0ex 0em 1ex 0em; }

html {
  margin: 0;
  padding: 0;
}

.programlisting {
  font-family: "DejaVu Mono", "Bitstream Vera Mono", "Lucida Console",
        "Nimbus Mono L", "Liberation Mono", "FreeMono", "Andale Mono",
        "Courier New", monospace;
  margin: 1ex 0ex 1ex 0ex ;
  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 ;
}

div.abstract {
  margin: 2em 5% 2em 5% ;
  padding: 1ex 1em 1ex 1em ;
/*  font-weight: bold ; */
  font-size: 90% ;
    text-align: left ;
}

div.abstract dl {line-height:1.5;}
div.abstract dt {color:#304070;}

div.abstracttitle{
    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-weight:bold;
    font-size:1.25em;
    text-align: center ;
}

span.abstractrunintitle{
    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-weight:bold;
}


.verbatim {
    overflow-x: auto ;
}

.alltt {
    overflow-x: auto ;
}


.bverbatim {
    margin: 1ex 0pt 1ex 0pt ;
    padding: .5ex 0pt .5ex 0pt ;
    overflow-x: auto ;
}

.lverbatim {
    margin: 1ex 0pt 1ex 0pt ;
    padding: .5ex 0pt .5ex 0pt ;
    overflow-x: auto ;
}

.fancyvrb {
    font-size:.85em ;
    margin: 3ex 0pt 3ex 0pt
}

.fancyvrblabel {
    font-size: .85em ;
    text-align: center ;
    font-weight: bold ;
    margin-top: 1ex ;
    margin-bottom: 1ex ;
}


.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;
}

.inlineminipage {
    display: inline-block ;
    text-align: left
}

/* 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 ;
     border-radius: 0px ;
    padding: .4ex .3em .4ex .3em ;
    margin: 0pt .3ex 0pt .3ex ;
  display: inline-block ;
}

```

```
.doublebox {
   border: 3px double black;
     border-radius: 0px ;
    padding: .4ex .3em .4ex .3em ;
    margin: 0pt .3ex 0pt .3ex ;
  display: inline-block ;
}

.ovalbox, .Ovalbox {
   border: 1px solid black;
     border-radius: 1ex ;
    padding: .4ex .3em .4ex .3em ;
    margin: 0pt .3ex 0pt .3ex ;
  display: inline-block ;
}

.Ovalbox { border-width: 2px ; }

.framebox {
   border: 1px solid black;
     border-radius: 0px ;
    padding: .3ex .2em 0ex .2em ;
    margin: 0pt .1ex 0pt .1ex ;
  display: inline-block ;
}


.mdframed {
    padding: 0ex ;
    margin: 2ex 0em 2ex 0em ;
}

.mdframed p { padding: 0ex .5em 0ex .5em ; }

.mdframed dl { padding: 1ex .5em 0ex .5em ; }

.mdframedtitle {
    padding: .5ex 0pt 0pt 0pt ;
    border-radius: 10pt 10pt 0pt 0pt ;
    display: block ;
    margin-bottom: 1ex ;
}

.mdframedsubtitle {
    display: block ;
}

.mdframedsubsubtitle {
    display: block ;
}

.mdtheorem {
    padding: 0ex .5em 0ex .5em ;
    margin: 3ex 5% 3ex 5% ;
}
```

```
2564
2565
2566 /* framed package */
2567 .framed, pre.boxedverbatim, fcolorbox {
2568     margin: 3ex 0em 3ex 0em ;
2569    border: 1px solid black;
2570      border-radius: 0px ;
2571     padding: .3ex 1em 0ex 1em ;
2572   display: block ;
2573 }
2574
2575 .shaded {
2576     margin: 3ex 0em 3ex 0em ;
2577     padding: .3ex 1em .3ex 1em ;
2578     display: block ;
2579 }
2580
2581 .snugframed {
2582     margin: 3ex 0em 3ex 0em ;
2583    border: 1px solid black;
2584      border-radius: 0px ;
2585   display: block ;
2586 }
2587
2588 .framedleftbar {
2589     margin: 3ex 0em 3ex 0em ;
2590    border-left: 3pt solid black;
2591      border-radius: 0px ;
2592     padding: .3ex .2em .3ex 1em ;
2593   display: block ;
2594 }
2595
2596 .framedtitle {
2597     margin: 0em ;
2598     padding: 0em ;
2599     font-size: 130%
2600 }
2601
2602 .framedtitle p { padding: .3em }
2603
2604
2605 /* For the niceframe package: */
2606
2607 div.niceframe, div.curlyframe, div.artdecoframe, div.generalframe {
2608     padding: 1ex ;
2609     margin: 2ex auto ;
2610     border-radius: 2ex ;
2611 }
2612
2613 div.niceframe {
2614     border: 6px groove black ;
2615 }
2616
2617 div.curlyframe {
2618     border-left: 3px dotted black ;
```

```
    border-right: 3px dotted black ;
    border-radius: 6ex ;
}

div.artdecoframe {
    border-left: 10px double black ;
    border-right: 10px double black ;
    border-radius: 6ex ;
}

div.generalframe {
    border: 6px groove black ;
}



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 {
```

```
2784     border: none ;
2785     background: none ;
2786 }
2787
2788
2789 /* For packages such as float, rotfloat, and algorithm2e: */
2790
2791 figure.boxed, figure.boxruled {
2792     border: 1px solid black ;
2793 }
2794
2795 figure.ruled {
2796     border-top: 1px solid black ;
2797     border-bottom: 1px solid black ;
2798     border-left: 0px ;
2799     border-right: 0px ;
2800     border-radius: 0px ;
2801     background: none ;
2802     box-shadow: none ;
2803 }
2804
2805 figure.ruled div.figurecaption, figure.boxruled div.figurecaption {
2806     border-top: 1px solid silver ;
2807     border-bottom: 1px solid silver ;
2808 }
2809
2810
2811 table {
2812     margin: 1ex auto 1ex auto ;
2813     border-collapse: separate ;
2814     border-spacing: 0px ;
2815     line-height: 1.3 ;
2816     }
2817
2818 table > tbody > tr.hline > td {border-top: 1px solid #808080 ; margin-top: 0ex ;
2819     margin-bottom: 0ex ; } /* for \hline */
2820
2821 tr.tbrule td {border-top: 1px solid black ; margin-top: 0ex ;
2822     margin-bottom: 0ex ; } /* for \toprule, \bottomrule */
2823
2824 td {padding: .5ex .5em .5ex .5em ;}
2825
2826 table td.tdl { text-align: left ; vertical-align: middle ; }
2827 table td.tdc { text-align: center ; vertical-align: middle ; }
2828 table td.tdat { text-align: center ; vertical-align: middle ; padding: 0px ; margin: 0px ; }
2829 table td.tdbang { text-align: center ; vertical-align: middle ; }
2830 table td.tdr { text-align: right ; vertical-align: middle ; }
2831 table td.tdp { text-align: left ; vertical-align: bottom ; }
2832 table td.tdm { text-align: left ; vertical-align: middle ; }
2833 table td.tdb { text-align: left ; vertical-align: top ; }
2834 table td.tdP { text-align: center ; vertical-align: bottom ; }
2835 table td.tdM { text-align: center ; vertical-align: middle ; }
2836 table td.tdB { text-align: center ; vertical-align: top ; }
2837
2838 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 ;
}
```

```
2894
2895 .minipage .tnotes {
2896     margin: 0pt ;
2897     padding: 0pt ;
2898 }
2899
2900 .tnotes dl dt p {margin-bottom:0px;}
2901
2902 .tnoteitemheader {margin-right: 1em;}
2903
2904
2905 /* for colortbl and cell color */
2906 div.cellcolor {
2907     width: 100% ;
2908     padding: .5ex .5em .5ex .5em ;
2909     margin: -.5ex -.5em -.5ex -.5em ;
2910 }
2911
2912
2913 /* for lyluatex */
2914 span.lyluatex {
2915     display: inline-block ;
2916 }
2917
2918 div.lyluatex p span.lateximagesource img {
2919     display: block ;
2920     margin-top: 3ex ;
2921     margin-bottom: 3ex ;
2922 }
2923
2924
2925 /* for bigdelim */
2926 .ldelim, .rdelim { font-size: 200% }
2927
2928
2929 /* center, flushleft, flushright environments */
2930 div.center{text-align:center;}
2931 div.center table {margin-left:auto;margin-right:auto;}
2932 div.flushleft{text-align:left;}
2933 div.flushleft table {margin-left:0em ; margin-right:auto;}
2934 div.flushright{text-align:right;}
2935 div.flushright table {margin-left:auto ; margin-right: 0em ;}
2936
2937
2938 /* Fancybox */
2939 div.Btrivlist table tr td {
2940     padding: .2ex 0em ;
2941 }
2942
2943
2944 /* program listing callouts: */
2945 span.callout {
2946      font-family: "DejaVu Sans", "Bitstream Vera Sans",
2947         Geneva, Verdana, sans-serif ;
2948     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 {
```

```
3004     margin: 1ex auto 0ex auto ;
3005     background: none ;
3006 }
3007
3008 div.author table tbody tr td { padding: .25ex ; }
3009 */
3010
3011 span.affiliation {font-size: .85em ; font-variant: small-caps; }
3012
3013 div.titledate {
3014     text-align: center ;
3015     font-size: .85em ;
3016     font-style: italic;
3017     margin: 1ex 0em 1ex 0em ;
3018 }
3019
3020
3021 nav.topnavigation{
3022     text-align: left ;
3023     padding: 0.5ex 1em 0.5ex 1em ;
3024 /*     margin: 2ex 0em 3ex 0em ; */
3025     margin: 0 ;
3026     border-bottom: 1px solid silver ;
3027     border-top: 1px solid silver ;
3028     clear:both ;
3029 }
3030
3031 nav.botnavigation{
3032     text-align: left ;
3033     padding: 0.5ex 1em 0.5ex 1em ;
3034 /*     margin: 3ex 0em 2ex 0em ; */
3035     margin: 0 ;
3036     border-top: 1px solid silver ;
3037     border-bottom: 1px solid silver ;
3038     clear:both ;
3039 }
3040
3041
3042 header {
3043     line-height: 1.2 ;
3044     font-size: 1em ;
3045     border-bottom: 1px solid silver ;
3046     margin: 0px ;
3047     padding: 2ex 1em 2ex 1em ;
3048     text-align:left ;
3049 }
3050
3051
3052 footer {
3053     font-size: .85em ;
3054     line-height: 1.2 ;
3055     margin-top: 1ex ;
3056     border-top: 1px solid silver ;
3057     padding: 2ex 1em 2ex 1em ;
3058     clear:both ;
```

```
3059     text-align:left ;
3060 }
3061
3062
3063 a.linkhome { font-weight:bold ; font-size: 1em ;}
3064
3065
3066 div.lateximagesource { padding: 0px ; margin: 0px ; display: none; }
3067
3068 img.lateximage{
3069     padding: 0pt ;
3070     margin: 0pt ;
3071     box-shadow: none ;
3072     border: none ;
3073     background: none ;
3074     max-width: 100% ;
3075     border-radius: 0ex ;
3076     border: none ;
3077 }
3078
3079
3080 /* The -1px right margin compensates for the 1px right border. */
3081 /* Without this -1px margin, the body container appears below instead */
3082 /* of floating to the side. */
3083 div.sidetoccontainer {
3084     font-family: "DejaVu Serif", "Bitstream Vera Serif",
3085         "Lucida Bright", Georgia, serif;
3086     float: left ;
3087     width: 20%;
3088     margin: 0pt -1px 3ex 0pt ;
3089     border-right: 1px solid silver;
3090     border-bottom: 1px solid silver;
3091     background: #FAF7F4 ;
3092     font-size:.9em ;
3093     border-radius: 0px 0px 20px 0px ;
3094 }
3095
3096 div.sidetoccontents {
3097     overflow-y: auto ;
3098     width: 100% ;
3099     text-align: left ;
3100 }
3101
3102
3103 nav.sidetoc p {line-height:1.2 ; margin: 1ex .5em 1ex .5em ;
3104     text-indent: 0 ; }
3105
3106 nav.sidetoc p a {color:black ; font-size: .7em ;}
3107
3108 div.sidetoctitle {font-size: 1.2em; font-weight:bold; text-align:center;
3109     border-bottom: 1px solid silver ;    }
3110
3111 nav.sidetoc a:hover {text-decoration: underline ; }
3112
3113
```

```
3114
3115 section.textbody { margin: 0ex 1em 0ex 1em ;}
3116
3117
3118 div.multicolsheading { -webkit-column-span: all;
3119     -moz-column-span: all; column-span: all; }
3120 div.multicols { -webkit-columns: 3 380px ;
3121     -moz-columns: 3 380px ; columns: 3 380px ; }
3122 div.multicols p {margin-top: 0ex}
3123
3124
3125 /* Used for xfrac and nicefrac: */
3126 span.numerator {
3127     font-size: 60% ;
3128     vertical-align: .4em ;
3129 }
3130
3131 span.denominator {
3132     font-size: 60%
3133 }
3134
3135
3136 /* Used for algorithm2e: */
3137 div.alg2evline{
3138     margin-left: 1em ;
3139     padding-left: 1em ;
3140     border-left: 1px solid black ;
3141     border-radius: 0px 0px 0px 1ex ;
3142 }
3143
3144 div.alg2evsline{
3145     margin-left: 1em ;
3146     padding-left: 1em ;
3147     border-left: 1px solid black ;
3148 }
3149
3150 div.alg2enoline{
3151     margin-left: 1em ;
3152     padding-left: 1em ;
3153 }
3154
3155 span.alg2elinenumber{
3156     margin-right: .5em ;
3157     font-size: 50% ;
3158     color: red ;
3159 }
3160
3161
3162 /* Used for algorithmicx: */
3163 span.floatright { float: right ; }
3164
3165
3166 /* keyfloat and tocdata: */
3167 .floatnotes {
3168     margin: 0ex 5% 0ex 5% ;
```

```
    padding: 0ex 1em 0ex 1em ;
    font-size:.80em ;
    text-align: left ;
}

.authorartist{
    display:block ;
    font-size:.70em ;
    font-style: italic;
}
nav .authorartist{ display:inline; }



/* Native LaTeX theorems: */

.theoremcontents { font-style: italic; margin-top: 3ex ; margin-bottom: 3ex ; }
.theoremlabel { font-style: normal; font-weight: bold ; margin-right: .5em ; }


/* theorem, amsthm, and ntheorem packages */

span.theoremheader,
span.theoremheaderplain,
span.theoremheaderdefinition,
span.theoremheaderbreak,
span.theoremheadermarginbreak,
span.theoremheaderchangebreak,
span.theoremheaderchange,
span.theoremheadermargin
{
    font-style:normal ; font-weight: bold ; margin-right: 1em ;
}

span.amsthmnameplain,
span.amsthmnamedefinition,
span.amsthmnumberplain,
span.amsthmnumberdefinition
{
    font-style:normal ; font-weight: bold ;
}


span.amsthmnameremark,
span.amsthmnumberremark
{font-style:italic ; font-weight: normal ; }


span.amsthmnoteplain,
span.amsthmnotedefinition
{font-style:normal ;}


span.theoremheaderremark,
```

```
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 backnaur package: */
div.backnaur {
    display: block ;
    margin: 2ex 2em 2ex 2em ;
}

div.backnaur p {
    margin: .25ex 0ex .25ex 0ex ;
}

div.backnaurprod {
    display: inline-block ;
    min-width: 8em ;
    text-align:right ;
}

div.backnaurdesc {
    display: inline-block ;
```

```
}


/* 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;
}
```

```
3334
3335 .latexlogotwoe {
3336   text-transform: none ;
3337   font-variant-numeric: oldstyle-nums ;
3338 }
3339
3340 .latexlogotwoesub {
3341   font-style:italic ;
3342   vertical-align: -0.27ex;
3343   margin-left: -0.11em;
3344   margin-right: -0.1em;
3345   font-size: 1em;
3346 }
3347
3348 .xelatexlogo {
3349     font-family: "Linux Libertine O", "Nimbus Roman No 9 L",
3350         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
3351     letter-spacing: .03em ;
3352     font-size: 1.1em;
3353 }
3354
3355 .xelatexlogosub {
3356   vertical-align: -0.27ex;
3357   margin-left: -0.0667em;
3358   margin-right: -.05em;
3359   font-size: 1em;
3360   letter-spacing: .03em ;
3361 }
3362
3363 .amslogo {
3364     font-family: "TeXGyreChorus","URW Chancery L",
3365         "Apple Chancery","ITC Zapf Chancery","Monotype Corsiva",
3366         "Linux Libertine O", "Nimbus Roman No 9 L",  "FreeSerif",
3367         "Hoefler Text", Times, "Times New Roman", serif ;
3368     font-style: italic ;
3369 }
3370
3371 .lyxlogo {
3372     font-family: "URW Classico", Optima, "Linux Biolinum O",
3373         "DejaVu Sans", "Bitstream Vera Sans", Geneva,
3374         Verdana, sans-serif ;
3375 }
3376
3377
3378 /* Only display top and bottom navigation if a small screen: */
3379 /* Hide the sidetoc if a small screen: */
3380 nav.topnavigation { display:none; }
3381 nav.botnavigation { display:none; }
3382
3383 /* Only display the sidetoc's webpage title if a small screen */
3384 span.sidetocthetitle { display: none }
3385
3386 @media screen and (max-width: 50em) {
3387     div.sidetoccontainer {
3388         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*}[overwrite]{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;}


div.book, 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 ;
    text-shadow: 2px 2px 3px #808080;
}

h1 {    /* title of the entire website, used on each page */
    font-variant: small-caps ;
    color: #304070 ;
    text-shadow: 2px 2px 3px #808080;
    background-color: #F7F7F0 ;
    background-image: linear-gradient(to bottom, #F7F7F0, #C0C0C4);
}

h1 {
  border-bottom: 1px solid #304070;
/*  border-top: 2px solid #304070; */
}

h2 {
  border-bottom: 1px solid #304070;
/*  border-top: 2px solid #304070; */
    background-color: #F7F7F0 ;
    background-image: linear-gradient(to bottom, #F7F7F0, #DAD0C0);
}


div.abstract {
    background: #f5f5eb ;
    background-image: linear-gradient(to bottom, #f5f5eb, #C8C8B8);

  border: 1px solid silver;
    border-radius: 1em ;
}

div.abstract dl {line-height:1.5;}
div.abstract dt {color:#304070;}
```

```
3492
3493 div.abstracttitle{
3494     font-family: "URW Classico", Optima, "Linux Biolinum O",
3495         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
3496         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
3497     font-weight:bold;
3498     font-variant: small-caps ;
3499     font-size:1.5em;
3500     border-bottom: 1px solid silver ;
3501     color: #304070 ;
3502     text-align: center ;
3503     text-shadow: 1px 1px 2px #808080;
3504 }
3505
3506 span.abstractrunintitle{
3507     font-family: "URW Classico", Optima, "Linux Biolinum O",
3508         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
3509         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
3510     font-weight:bold;
3511 }
3512
3513
3514 div.epigraph, div.dictum {
3515     background: #f5f5eb ;
3516     background-image: linear-gradient(to bottom, #f5f5eb, #C8C8B8);
3517
3518     border: 1px solid silver ;
3519     border-radius: 1ex ;
3520     box-shadow: 3px 3px 3px #808080 ;
3521 }
3522
3523
3524 .example {
3525     background-color: #f5f5eb ;
3526     background-image: linear-gradient(to bottom, #f5f5eb, #C8C8B8);
3527
3528 }
3529
3530 div.exampletitle{
3531     font-family: "URW Classico", Optima, "Linux Biolinum O",
3532         "Linux Libertine O", "Liberation Serif", "Nimbus Roman No 9 L",
3533         "FreeSerif", "Hoefler Text", Times, "Times New Roman", serif;
3534     font-weight:bold;
3535     font-variant: small-caps ;
3536     border-bottom: 1px solid silver ;
3537     color: #304070 ;
3538     text-align: center ;
3539     text-shadow: 1px 1px 2px #808080;
3540 }
3541
3542
3543 .sidebar {
3544     background-color: #f5f5eb ;
3545     background-image: linear-gradient(to bottom, #f5f5eb, #C8C8B8);
3546
```

```
}

div.sidebartitle{
    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-weight:bold;
    font-variant: small-caps ;
    border-bottom: 1px solid silver ;
    color: #304070 ;
    text-align: center ;
    text-shadow: 1px 1px 2px #808080;
}


.fancyvrblabel {
    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-weight:bold;
    font-variant: small-caps ;
    font-size: 1.5em ;
    color: #304070 ;
    text-align: center ;
    text-shadow: 1px 1px 2px #808080;
}

div.minipage {
    background-color: #eeeee7 ;
    border: 1px solid silver ;
    border-radius: 1ex ;
}

table div.minipage { background: none ; border: none ; }

div.framebox div.minipage {border:none ; background:none}

section.textbody > div.minipage {
    box-shadow: 3px 3px 3px #808080 ;
}

div.fboxBlock div.minipage { box-shadow: none ; }

.framed .minipage , .framedleftbar .minipage {
    border: none ;
    background: none ;
    padding: 0ex ;
    margin: 0ex ;
}

figure.figure .minipage, div.figurecaption .minipage { border: none; }

div.marginblock div.minipage ,
div.marginparblock div.minipage
    { border: none; }
```

```
3602
3603 figure , div.marginblock {
3604     background-color: #eeeee7 ;
3605     border: 1px solid silver ;
3606     border-radius: 1ex ;
3607     box-shadow: 3px 3px 3px #808080 ;
3608 }
3609
3610 figure figure {
3611     border: 1px solid silver ;
3612     margin: 0em ;
3613     box-shadow: none ;
3614 }
3615
3616 /*
3617 div.figurecaption {
3618     border-top: 1px solid silver ;
3619     border-bottom: 1px solid silver ;
3620     background-color: #e8e8e8 ;
3621 }
3622 */
3623
3624
3625 div.table {
3626     box-shadow: 3px 3px 3px #808080 ;
3627 }
3628
3629 /*
3630 .tnotes {
3631     background: #e8e8e8;
3632     border: 1px solid silver;
3633 }
3634 */
3635
3636
3637 nav.topnavigation{
3638     background-color: #b0b8b0 ;
3639     background-image: linear-gradient(to bottom,#e0e0e0,#b0b8b0) ;
3640 }
3641
3642 nav.botnavigation{
3643     background-color: #b0b8b0 ;
3644     background-image: linear-gradient(to top,#e0e0e0,#b0b8b0) ;
3645 }
3646
3647
3648
3649 header{
3650     background-color: #F7F7F0 ;
3651     background-image: linear-gradient(to top, #F7F7F0, #b0b8b0);
3652 }
3653
3654 footer{
3655     background-color: #F7F7F0 ;
3656     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*}[overwrite]{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;
}
```

```
3703
3704 span.textrm {
3705     font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
3706         "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
3707         "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
3708         "Times New Roman", serif;
3709 }
3710
3711 span.textsf {
3712      font-family: "DejaVu Sans", "Bitstream Vera Sans",
3713         Geneva, Verdana, sans-serif ;
3714 }
3715
3716
3717
3718 div.book, h1, h2, h3, h4, h5, h6, span.paragraph, span.subparagraph
3719 {
3720     font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
3721         "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
3722         "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
3723         "Times New Roman", serif;
3724     color: #800000 ;
3725     text-shadow: none ;
3726 }
3727
3728 h1, h2 {
3729     background-color: #fffcf5 ;
3730     background-image: none ;
3731     border-bottom: 1px solid #808080;
3732 /*    border-top: 2px solid #808080; */
3733 }
3734
3735 div.abstracttitle {
3736     font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
3737         "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
3738         "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
3739         "Times New Roman", serif;
3740     color: black ;
3741     text-shadow: none ;
3742 }
3743
3744 span.abstractrunintitle {
3745     font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
3746         "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
3747         "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
3748         "Times New Roman", serif;
3749     color: black ;
3750     text-shadow: none ;
3751 }
3752
3753 div.abstract { font-size: 100% }
3754
3755 .sidebar {
3756     background: #fffcf5;
3757     background-image: none ;
```

```
  margin: 2em 5% 2em 5%;
  padding: 0.5em 1em;
  border: none ;
  border-top : 1px solid silver;
  border-bottom : 1px solid silver;
  font-size: 90% ;
}

div.sidebartitle{
    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;
    color: #800000 ;
    text-shadow: none ;
    border: none ;
}

.example {
    background: #fffcf5;
    background-image: none ;
  margin: 2em 5% 2em 5%;
  padding: 0.5em 1em;
  border: none ;
  border-top : 1px solid silver;
  border-bottom : 1px solid silver;
}

div.exampletitle{
    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;
    color: #800000 ;
    text-shadow: none ;
    border: none ;
}

div.fancyvrblabel{
    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;
    color: #800000 ;
    text-shadow: none ;
    border: none ;
}



figure {
    margin: 5ex 5% 5ex 5% ;
    padding: 1ex 1em 1ex 1em ;
    background-color: #fffcf5 ;
    overflow-x: auto ;
```

```
3813     border: none ;
3814 /*     border-top: 1px solid silver; */
3815 /*     border-bottom: 1px solid silver; */
3816 }
3817
3818
3819 div.figurecaption , .lstlisting {
3820     border: none ;
3821 /*     border-top: 1px solid silver ; */
3822 /*     border-bottom: 1px solid silver ; */
3823     background-color: #fffcf5 ;
3824 }
3825
3826 .tnotes {
3827     background: #fffcf5 ;
3828     border-top: 1px solid silver ;
3829     border-bottom: 1px solid silver ;
3830 }
3831
3832 .theorem {
3833         background: none ;
3834 }
3835
3836 .minipage {
3837     background-color: #fffcf5 ;
3838     border: none ;
3839 }
3840
3841 div.floatrow figure { border: none ; }
3842
3843 figure figure { border: none ; }
3844
3845
3846 nav.toc, nav.lof, nav.lot, nav.lol {
3847     font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
3848         "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
3849         "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
3850         "Times New Roman", serif;
3851 }
3852
3853 div.sidetoccontainer {
3854     font-family: "Linux Libertine O", "Hoefler Text", "Garamond",
3855         "Bembo", "Janson", "TeX Gyre Pagella", "Palatino",
3856         "Liberation Serif", "Nimbus Roman No 9 L",  "FreeSerif", Times,
3857         "Times New Roman", serif;
3858     background-image: linear-gradient(to bottom, #fffcf5, #C0C0C0);
3859 }
3860
3861 div.sidetoctitle{
3862     color: #800000 ;
3863 }
3864
3865 header{
3866     background-color: #e0e0e0 ;
3867     background-image: linear-gradient(to top, #fffcf5, #b0b0b0);
```

```
    text-align:center ;
}

footer{
    background-color: #e0e0e0 ;
    background-image: linear-gradient(to bottom, #fffcf5, #b0b0b0);
    padding: 2ex 1em 2ex 1em ;
    text-align:left ;
}

nav.botnavigation {
    background: #dedcd5 ;
    border-top: 1px solid black ;
}
\end{filecontents*}
% \end{Verbatim}% for syntax highlighting
\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*}[overwrite]{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*}[overwrite]{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*}[overwrite]{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 `lateximages` when using WINDOWS.

This must be present when compiling the project, but does not need to be present when distributing the resulting HTML files.

The arguments are each of the three fields from `<project>-images.txt`, and also the base name of the source file.

MiKTeX does not allow file `lwarp_one_limage.cmd` to be created directly by *lwarpmk*, so `lwarp_one_limage.txt` is created instead, then copied to `lwarp_one_limage.cmd` by *lwarpmk*. This occurs each time *lwarpmk* used to create `lateximages`.

Config file:

```
\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

The default MathJax script used by lwarp when using MATHJAX. A recent version of MathJax is used, as served by the recommended repository. Adjustments are made to allow LaTeX to control the equation tags.

`\MathJaxFilename` determines which script file is copied into the HTML pages, and defaults to `lwarp_mathjax.txt`. The script files must be present when compiling the project, but do not need to be present when distributing the resulting HTML files.

custom script

To generate a custom script, such as to use a local repository, copy `lwarp_mathjax.txt` to a new file, make changes while keeping lwarp's adjustments for equation numbering, and use `\MathJaxFilename` to select new filename.

Config file:

```
\begin{LWRwriteconf}
\begin{filecontents*}[overwrite]{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"
    }
    });

    PARSE.Augment({
    mySection: function (name) {
        seteqsection.num = AMS.number;
        var n = this.GetArgument(name);
        if (n === "") {
            seteqsection = seteqsectionDefault;
        } else {
            if (!seteqsections["_"+n])
                seteqsections["_"+n] = {name:n, num:0};
            seteqsection = seteqsections["_"+n];
        }
        AMS.number = seteqsection.num;
    },
    mySetEqNumber: function (name) {
        var n = this.GetArgument(name);
        if (!n || !n.match(/^ *[0-9]+ *$/))
            n = "";
        else
            n = parseInt(n)-1;
        <!-- $ syntax highlighting -->
        if (n === "" || n < 1)
            TEX.Error
            ("Argument to "+name+" should be a positive integer");
        AMS.number = n;
    }
    });
    MathJax.Hub.Config({
    TeX: {
        equationNumbers: {
            formatTag: function (n) {
```

```
                <!-- if not numeric, don't include the chapter -->
                if (!n.match(/^ *[0-9]+ *$/ ))
                <!-- $ syntax highlighting -->
                    return "("+(n).replace(/^\./,"")+")" ;
                else
                  return "("+(seteqsection.name+"."+n).replace(/^\./,"")+")" ;
            },
            formatID: function (n) {
                n = (seteqsection.name+'.'+n).replace
                    (/[:"'<>&]/g,"").replace(/^\./,"");
                return 'mjx-eqn-' + n;
            }
        }
    }
    });
});
</script>

<!-- http://docs.mathjax.org/en/latest/options/ThirdParty.html -->
<script type="text/x-mathjax-config">
  MathJax.Ajax.config.path["Contrib"] =
    "https://cdn.mathjax.org/mathjax/contrib";
</script>

<script type="text/x-mathjax-config">
MathJax.Hub.Config({
    TeX: {
        extensions: ["autoload-all.js"] ,
        equationNumbers: {
            autoNumber: "AMS"
        }
    }
});
</script>

<!-- Alternative CDN provider: -->
<script type="text/javascript" async
src="https://cdnjs.cloudflare.com/ajax/libs/mathjax/2.7.6/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 `lateximages` and SVG math images are generated using multiple processes in parallel. For UNIX and LINUX, every 32 images the `wait` command is issued to wait for the previous batch of images to finish processing before starting a new batch. For WINDOWS, every 32 images one task is dispatched with

```
START /B /WAIT /BELOWNORMAL
```

which causes the operating system to wait until this lesser-priority tasks finishes, hopefully also waiting for the normal priority tasks which were already in progress to also complete. Afterwards, the next batch of images is started.

The following is only generated if the `lwarpmk` option was given to lwarp.

```
\begin{LWR@createlwarpmk}

\begin{filecontents*}[overwrite]{lwarpmk.lua}
#!/usr/bin/env texlua

-- Copyright 2016-2020 Brian Dunn


printversion = "v0.79"
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*,
```

```
    *_html.pdf, *_html.html, *_html.sidetoc
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 splitfilename ( pathandfilename )
--
-- Separates out the path and extension from a filename.
-- Returns path, filename with extension, and extension.
-- Ex: thispath, thisfilename, thisextension = splitfilename ("path/to/filename.ext")
--
-- https://www.fhug.org.uk/wiki/wiki/doku.php?id=plugins:code_snippets:
--      split_filename_in_to_path_filename_and_extension
--
    if lfs.attributes(pathandfilename,"mode") == "directory" then
      local strPath = pathandfilename:gsub("[\\/]$","") -- $ (syntax highlighting)
        return strPath.."\\","",""
    end
    pathandfilename = pathandfilename.."."
    return pathandfilename:match("^(.-)([^\\/]-)%.([^\\/%.]-)%.?$")
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 printhowtorecompile ()
-- Tells the user how to recompile to regenerate the configuration files.
  print ("lwarpmk: The configuration files lwarpmk.conf and "..sourcename..".lwarpmkconf" )
    print ("lwarpmk:   must be updated.  To do so, recompile" )
    print ("lwarpmk:   " , sourcename..".tex" )
    if ( printlatexcmd == "" ) then
        print ("lwarpmk:   using xe/lua/pdflatex," )
    else
        print ("lwarpmk:   using the command:")
        print ("lwarpmk:   " , printlatexcmd )
    end
    print ("lwarpmk:   then use lwarpmk again.")
end -- printhowtorecompile


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
```

```
4275         cvalueerror ( line, linenum , cvalue )
4276     end
4277 elseif ( cvarname == "sourcename" ) then sourcename = cvalue
4278 elseif ( cvarname == "homehtmlfilename" ) then homehtmlfilename = cvalue
4279 elseif ( cvarname == "htmlfilename" ) then htmlfilename = cvalue
4280 elseif ( cvarname == "imagesdirectory" ) then imagesdirectory = cvalue
4281 elseif ( cvarname == "imagesname" ) then imagesname = cvalue
4282 elseif ( cvarname == "latexmk" ) then latexmk = cvalue
4283 elseif ( cvarname == "printlatexcmd" ) then printlatexcmd = cvalue
4284 elseif ( cvarname == "HTMLlatexcmd" ) then HTMLlatexcmd = cvalue
4285 elseif ( cvarname == "printindexcmd" ) then printindexcmd = cvalue
4286 elseif ( cvarname == "HTMLindexcmd" ) then HTMLindexcmd = cvalue
4287 elseif ( cvarname == "latexmkindexcmd" ) then latexmkindexcmd = cvalue
4288 elseif ( cvarname == "glossarycmd" ) then glossarycmd = cvalue
4289 elseif ( cvarname == "pdftotextenc" ) then pdftotextenc = cvalue
4290 else
4291     print ("lwarpmk: ===")
4292     print ("lwarpmk: " .. linenum .. " : " .. line ) ;
4293     print (
4294         "lwarpmk: Incorrect variable name \"" .. cvarname .. "\" in " ..
4295         conffile ..".\n"
4296     ) ;
4297     print ("lwarpmk: ===")
4298 --    printconf () ;
4299 os.exit(1) ;
4300 end -- cvarname
4301 end -- do scan lines
4302 io.close(cfile)
4303 end -- file exists
4304 -- Error if sourcename is "lwarp".
4305 -- This could happen if a local copy of lwarp has recently been recompiled.
4306 if sourcename=="lwarp" then
4307     print ("lwarpmk: ===")
4308    print ("lwarpmk: lwarp.sty has recently been recompiled in this directory,")
4309    print ("lwarpmk: and \"lwarpmk.conf\" is no longer set for your own project.")
4310    print ("lwarpmk: Recompile your own project using pdf/lua/xelatex <projectname>.")
4311    print ("lwarpmk: After a recompile, \"lwarpmk.conf\" will be set for your project,")
4312     print ("lwarpmk: and you may again use lwarpmk.")
4313     print ("lwarpmk: ===")
4314     os.exit(1)
4315 end -- sourcename of "lwarp"
4316 -- Select some operating-system commands:
4317 if opsystem=="Unix" then  -- For Unix / Linux / Mac OS:
4318     rmname = "rm"
4319     mvname = "mv"
4320     cpname = "cp"
4321     touchnamepre = "touch"
4322     touchnamepost = ""
4323     newtouchname = "touch"
4324     dirslash = "/"
4325     opquote= "\'"
4326     cmdgroupopenname = " ( "
4327     cmdgroupclosename = " ) "
4328     seqname = " && "
4329     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." )
    printhowtorecompile ()
    print ("lwarpmk: ===")
    os.exit(1)
end
-- Error if the configuration file's version is not current:
if ( confversion ~= requiredconfversion ) then
    print ("lwarpmk: ===")
    printhowtorecompile ()
    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


function refreshdate ()
```

```
os.execute(touchnamepre .. " " .. sourcename .. ".tex " .. touchnamepost)
end



function reruntoget (filesource)
--
-- Scan the LaTeX log file for the phrase "Rerun to get",
-- indicating that the file should be compiled again.
-- Return true if found.
--
local fsource = io.open(filesource)
for line in fsource:lines() do
if ( string.find(line,"Rerun to get") ~= nil ) then
    io.close(fsource)
    return true
end -- if
end -- do
io.close(fsource)
return false
end



function onetime (latexcmd, fsuffix)
--
-- Compile one time, return true if should compile again.
-- fsuffix is "" for print, "_html" for HTML output.
--
print("lwarpmk: Compiling with: " .. latexcmd)
executecheckerror (
    latexcmd ,
    "Compile error."
)
return (reruntoget(sourcename .. fsuffix .. ".log") ) ;
end


function manytimes (latexcmd, fsuffix)
--
-- Compile up to five times.
-- fsuffix is "" for print, "_html" for HTML output
--
if onetime(latexcmd, fsuffix) == true then
if onetime(latexcmd, fsuffix) == true then
if onetime(latexcmd, fsuffix) == true then
if onetime(latexcmd, fsuffix) == true then
if onetime(latexcmd, fsuffix) == true then
end end end end end
end


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.
-- Also removes sourcename_html.pdf, sourcename_html.html,
-- and sourcename_html.sidetoc.
--
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* " ..
    sourcename .. "_html.pdf " ..
    sourcename .. "_html.html " ..
    sourcename .. "_html.sidetoc " ..
    " *_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 ("")
    print ("lwarpmk: ===")
    print ("lwarpmk: The HTML version of the document does not exist.")
    print ("lwarpmk: Enter \"lwarpmk html\" to compile the HTML version.")
    print ("lwarpmk: ===")
    os.exit(1)
end
io.close (htmlpdffile)
end -- checkhtmlpdfexists


function warnlimages ()
--
-- Warning of a missing <sourcename>-images.txt file:
    print ("lwarpmk: ===")
    print ("lwarpmk: \"" .. sourcename .. "-images.txt\" does not exist.")
    print ("lwarpmk: Your project does not use SVG math or other lateximages,")
    print ("lwarpmk: or the file has been deleted somehow.")
    print ("lwarpmk: Use \"lwarpmk html1\" to recompile your project")
    print ("lwarpmk: and recreate \"" .. sourcename .. "-images.txt\".")
   print ("lwarpmk: If your project does not use SVG math or other lateximages,")
   print ("lwarpmk: then \"" .. sourcename .. "-images.txt\" will never exist, and")
    print ("lwarpmk: \"lwarpmk limages\" will not be necessary.")
    print ("lwarpmk: ===")
end -- warnlimages


function warnlimagesrecompile ()
-- Warning if must recompile before creating limages:
    print ("")
    print ("lwarpmk: ===")
    print ("lwarpmk: Cross-references are not yet correct.")
   print ("lwarpmk: The document must be recompiled before creating the lateximages.")
   print ("lwarpmk: Enter \"lwarpmk html1\" again, then try \"lwarpmk limages\" again.")
    print ("lwarpmk: ===")
end --warnlimagesrecompile


function checklimages ()
--
-- Check <sourcename>.txt to see if need to recompile first.
-- If any entry has a page number of zero, then there were incorrect images.
--
print ("lwarpmk: Checking for a valid " .. sourcename .. "-images.txt file.")
local limagesfile = io.open(sourcename .. "-images.txt", "r")
if ( limagesfile == nil ) then
    warnlimages ()
    os.exit(1)
end
-- Track warning to recompile if find a page 0
local pagezerowarning = false
-- Scan <sourcename>.txt
for line in limagesfile:lines() do
    -- lwimgpage is the page number in the PDF which has the image
```

```
4550     -- lwimghash is true if this filename is a hash
4551     -- lwimgname is the lateximage filename root to assign for the image
4552     i,j,lwimgpage,lwimghash,lwimgname = string.find (line,"|(.*)|(.*)|(.*)|")
4553     -- For each entry:
4554     if ( (i~=nil) ) then
4555         -- If the page number is 0, image references are incorrect
4556         --  and must recompile the soure document:
4557         if ( lwimgpage == "0" ) then
4558             pagezerowarning = true
4559         end
4560     end -- if i~=nil
4561 end -- do
4562 -- The last line should be |end|end|end|.
4563 -- If not, the compile must have aborted, and the images are incomplete.
4564 if ( lwimgpage ~= "end" ) then
4565     warnlimagesrecompile()
4566     os.exit(1) ;
4567 end
4568 if ( pagezerowarning ) then
4569     warnlimagesrecompile()
4570     os.exit(1) ;
4571 end -- pagezerowarning
4572 end -- checklimages
4573
4574
4575 function createuniximage ( lwimgfullname )
4576 --
4577 -- Create one lateximage for Unix / Linux / Mac OS.
4578 --
4579 executecheckerror (
4580     cmdgroupopenname ..
4581     "pdfseparate -f " .. lwimgpage .. " -l " .. lwimgpage .. " " ..
4582         sourcename .."_html.pdf " ..
4583         imagesdirectory .. dirslash .."lateximagetemp-%d" .. ".pdf" ..
4584         seqname ..
4585     -- Crop the image:
4586     "pdfcrop  --hires  " .. imagesdirectory .. dirslash .. "lateximagetemp-" ..
4587         lwimgpage .. ".pdf " ..
4588         imagesdirectory .. dirslash .. lwimgname .. ".pdf" ..
4589         seqname ..
4590     -- Convert the image to svg:
4591    "pdftocairo -svg  -noshrink  " .. imagesdirectory .. dirslash .. lwimgname .. ".pdf " ..
4592         imagesdirectory .. dirslash .. lwimgname ..".svg" ..
4593         seqname ..
4594     -- Remove the temporary files:
4595    rmname .. " " .. imagesdirectory .. dirslash .. lwimgname .. ".pdf" .. seqname ..
4596    rmname .. " " .. imagesdirectory .. dirslash .. "lateximagetemp-" .. lwimgpage .. ".pdf" ..
4597     cmdgroupclosename .. " >/dev/null " .. bgname
4598     ,
4599     "File error trying to convert " .. lwimgfullname
4600 )
4601 -- Every 32 images, wait for completion at below normal priority,
4602 --  allowing other image tasks to catch up.
4603 numimageprocesses = numimageprocesses + 1
4604 if ( numimageprocesses > 32 ) then
```

```
4605     numimageprocesses = 0
4606     print ( "lwarpmk: waiting" )
4607     executecheckerror ( "wait" , "File error trying to wait.")
4608 end
4609 end -- createuniximage
4610
4611
4612 function createwindowsimage ( lwimgfullname )
4613 --
4614 -- Create one lateximage for Windows.
4615 --
4616 -- Every 32 images, wait for completion at below normal priority,
4617 --  allowing other image tasks to catch up.
4618 numimageprocesses = numimageprocesses + 1
4619 if ( numimageprocesses > 32 ) then
4620     numimageprocesses = 0
4621     thiswaitcommand = "/WAIT /BELOWNORMAL"
4622     print ( "lwarpmk: waiting" )
4623 else
4624     thiswaitcommand = ""
4625 end
4626 -- Execute the image generation command
4627 executecheckerror (
4628     "start /B " .. thiswaitcommand .. " \"\" lwarp_one_limage " ..
4629     lwimgpage .. " " ..
4630     lwimghash .. " " ..
4631     lwimgname .. " " ..
4632     sourcename .. " <nul >nul"
4633     ,
4634     "File error trying to create image."
4635 )
4636 end -- createwindowsimage
4637
4638
4639 function createonelateximage ( line )
4640 --
4641 -- Given the next line of <sourcename>.txt, convert a single image.
4642 --
4643 -- lwimgpage is the page number in the PDF which has the image
4644 -- lwimghash is true if this filename is a hash
4645 -- lwimgname is the lateximage filename root to assign for the image
4646 i,j,lwimgpage,lwimghash,lwimgname = string.find (line,"|(.*)|(.*)|(.*)|")
4647 -- For each entry:
4648 if ( (i~=nil) ) then
4649     -- Skip if the page number is 0:
4650     if ( lwimgpage == "0" ) then
4651         pagezerowarning = true
4652     -- Skip if the page number is "end":
4653     else if ( lwimgpage == "end" ) then
4654     else
4655         -- Skip is this image is hashed and already exists:
4656        local lwimgfullname = imagesdirectory .. dirslash .. lwimgname .. ".svg"
4657         if (
4658             (lwimghash ~= "true") or
4659             (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 "end"
    end -- not page 0
end -- not nil
end -- createonelateximage


function createlateximages ()
--
-- Create lateximages based on <sourcename>-images.txt:
--
-- See if the document must be recompiled first:
checklimages ()
-- See if the HTML version exists:
checkhtmlpdfexists ()
-- Attempt to create the lateximages:
print ("lwarpmk: Creating lateximages.")
local limagesfile = io.open(sourcename .. "-images.txt", "r")
if ( limagesfile == nil ) then
    warnlimages ()
    os.exit(1)
end
-- Create the lateximages directory, ignore error if already exists
err = os.execute("mkdir " .. imagesdirectory)
-- For Windows, create lwarp_one_limage.cmd from lwarp_one_limage.txt:
if opsystem=="Windows" then
    executecheckerror (
        cpname .. " lwarp_one_limage.txt lwarp_one_limage.cmd" ,
       "File error trying to copy lwarp_one_limage.txt to lwarp_one_limage.cmd"
    )
end -- create lwarp_one_limage.cmd
-- Track the number of parallel processes
numimageprocesses = 0
-- Track warning to recompile if find a page 0
pagezerowarning = false
-- Scan <sourcename>.txt
for line in limagesfile:lines() do
    createonelateximage ( line )
end -- do
io.close(limagesfile)
```

```
print ( "lwarpmk limages: ===")
print ( "lwarpmk limages: Wait a moment for the images to complete" )
print ( "lwarpmk limages:   before reloading the page." )
print ( "lwarpmk limages: ===")
print ( "lwarpmk limages: Done." )
if ( pagezerowarning == true ) then
    print ( "lwarpmk limages: WARNING: Images will be incorrect." )
    print ( "lwarpmk limages:   Enter \"lwarpmk cleanlimages\", then" )
    print ( "lwarpmk limages:   recompile the document one more time, then" )
    print ( "lwarpmk limages:   repeat \"lwarpmk images\" again." )
end -- pagezerowarning
end -- function


function convertepstopdf ()
--
-- Converts EPS files to PDF files.
-- The filenames are arg[argindex] and up.
-- arg[1] is the command "epstopdf".
--
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] .. "\"")
        thispath, thisfilename, thisextension = splitfilename(arg[i])
        if ( thispath == nil ) then
            os.execute ( "epstopdf " .. arg[i] )
        else
            os.execute (
                "epstopdf " ..
                thispath .. thisfilename .. "." .. thisextension .. " " ..
                thispath .. thisfilename .. ".pdf"
            )
        end
    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] .. "\"")
        thispath, thisfilename, thisextension = splitfilename(arg[i])
        if ( thispath == nil ) then
```

```
            os.execute ( "pdftocairo -svg " .. arg[i] )
        else
            os.execute (
                "pdftocairo -svg " ..
                thispath .. thisfilename .. "." .. thisextension .. " " ..
                thispath .. thisfilename .. ".svg"
            )
        end
    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.")


-- lwarpmk print:

if arg[1] == "print" then
loadconf ()
if ( latexmk == "true" ) then
    print ("lwarpmk: Compiling with: " .. printlatexcmd)
    executecheckerror (
        printlatexcmd ,
        "Compile error."
    )
    print ("lwarpmk: Done.")
else -- not latexmk
    verifyfileexists (sourcename .. ".tex") ;
    -- See if up to date:
    if (
        ( lfs.attributes ( sourcename .. ".pdf" , "modification" ) == nil ) or
        (
```

```
4825             lfs.attributes ( sourcename .. ".tex" , "modification" ) >
4826             lfs.attributes ( sourcename .. ".pdf" , "modification" )
4827         )
4828     ) then
4829         -- Recompile if not yet up to date:
4830         manytimes(printlatexcmd, "")
4831         print ("lwarpmk: Done.") ;
4832     else
4833         print ("lwarpmk: " .. sourcename .. ".pdf is up to date.") ;
4834     end
4835 end -- not latexmk
4836
4837
4838 -- lwarpmk print1:
4839
4840 elseif arg[1] == "print1" then
4841     loadconf ()
4842     verifyfileexists (sourcename .. ".tex") ;
4843     onetime(printlatexcmd, "")
4844     print ("lwarpmk: Done.") ;
4845
4846
4847 -- lwarpmk printindex:
4848 -- Compile the index then touch the source
4849 -- to trigger a recompile of the document:
4850
4851 elseif arg[1] == "printindex" then
4852 loadconf ()
4853 os.execute ( printindexcmd )
4854 print ("lwarpmk: -------")
4855 updateanddone ()
4856
4857
4858 -- lwarpmk printglossary:
4859 -- Compile the glossary then touch the source
4860 -- to trigger a recompile of the document:
4861
4862 elseif arg[1] == "printglossary" then
4863 loadconf ()
4864 print ("lwarpmk: Processing the glossary.")
4865
4866 os.execute(glossarycmd .. " " .. sourcename)
4867 updateanddone ()
4868
4869
4870 -- lwarpmk html:
4871
4872 elseif arg[1] == "html" then
4873 loadconf ()
4874 if ( latexmk == "true" ) then
4875     print ("lwarpmk: Compiling with: " .. HTMLlatexcmd)
4876     executecheckerror (
4877         HTMLlatexcmd ,
4878         "Compile error."
4879     )
```

```
4880     pdftohtml ()
4881     print ("lwarpmk: Done.")
4882 else -- not latexmk
4883     verifyfileexists ( sourcename .. ".tex" ) ;
4884     -- See if exists and is up to date:
4885     if (
4886       ( lfs.attributes ( homehtmlfilename .. ".html" , "modification" ) == nil ) or
4887         (
4888             lfs.attributes ( sourcename .. ".tex" , "modification" ) >
4889             lfs.attributes ( homehtmlfilename .. ".html" , "modification" )
4890         )
4891     ) then
4892         -- Recompile if not yet up to date:
4893         manytimes(HTMLlatexcmd, "_html")
4894         pdftohtml ()
4895         print ("lwarpmk: Done.")
4896     else
4897         print ("lwarpmk: " .. homehtmlfilename .. ".html is up to date.")
4898     end
4899 end -- not latexmk
4900
4901
4902 -- lwarpmk html1:
4903
4904 elseif arg[1] == "html1" then
4905     loadconf ()
4906     verifyfileexists ( sourcename .. ".tex" ) ;
4907     onetime(HTMLlatexcmd, "_html")
4908     pdftohtml ()
4909     print ("lwarpmk: Done.")
4910
4911
4912 -- lwarpmk pdftohtml:
4913 elseif arg[1] == "pdftohtml" then
4914     loadconf ()
4915     pdftohtml ()
4916
4917
4918 -- lwarpmk htmlindex:
4919 -- Compile the index then touch the source
4920 -- to trigger a recompile of the document:
4921
4922 elseif arg[1] == "htmlindex" then
4923 loadconf ()
4924 os.execute ( HTMLindexcmd )
4925 print ("lwarpmk: -------")
4926 updateanddone ()
4927
4928
4929 -- lwarpmk htmlglossary:
4930 -- Compile the glossary then touch the source
4931 -- to trigger a recompile of the document.
4932 -- The <sourcename>.xdy file is created by the glossaries package.
4933
4934 elseif arg[1] == "htmlglossary" then
```

```
loadconf ()
print ("lwarpmk: Processing the glossary.")
os.execute(glossarycmd .. " " .. sourcename .. "_html")
updateanddone ()


-- lwarpmk limages:
-- Scan the <sourcename>.txt file to create lateximages.

elseif arg[1] == "limages" then
loadconf ()
print ("lwarpmk: Processing images.")
createlateximages ()
print ("lwarpmk: Done.")


-- lwarpmk again:
-- Touch the source to trigger a recompile.

elseif arg[1] == "again" then
loadconf ()
updateanddone ()


-- lwarpmk clean:
-- Remove project.aux, .toc, .lof, .lot, .log, *.idx, *.ind, *_html_inc.*, .gl*

elseif arg[1] == "clean" then
loadconf ()
removeaux ()
print ("lwarpmk: Done.")


-- lwarpmk cleanall
-- Remove project.aux, .toc, .lof, .lot, .log, *.idx, *.ind, *_html_inc.*, .gl*
--    and also project.pdf, project.dvi, *.html

elseif arg[1] == "cleanall" then
loadconf ()
removeaux ()
os.execute ( rmname .. " " ..
    sourcename .. ".pdf " .. sourcename .. "_html.pdf " ..
    sourcename .. ".dvi " .. sourcename .. "_html.dvi " ..
    "*.html"
    )
print ("lwarpmk: Done.")


-- lwarpmk cleanlimages
-- Remove images from the imagesdirectory.

elseif arg[1] == "cleanlimages" then
loadconf ()
os.execute ( rmname .. " " .. imagesdirectory .. dirslash .. "*" )
print ("lwarpmk: Done.")
```

```
4990
4991 -- lwarpmk epstopdf <list of file names>
4992 -- Convert EPS files to PDF using epstopdf
4993 elseif arg[1] == "epstopdf" then
4994 convertepstopdf ()
4995 print ("lwarpmk: Done.")
4996
4997
4998 -- lwarpmk pdftosvg <list of file names>
4999 -- Convert PDF files to SVG using pdftocairo
5000 elseif arg[1] == "pdftosvg" then
5001 convertpdftosvg ()
5002 print ("lwarpmk: Done.")
5003
5004
5005 -- lwarpmk with no argument :
5006
5007 elseif (arg[1] == nil) then
5008 printhelp ()
5009
5010
5011 -- lwarpmk -h or lwarpmk --help :
5012
5013 elseif (arg[1] == "-h" ) or (arg[1] == "--help") then
5014 printusage ()
5015
5016
5017 -- Unknown command:
5018
5019 else
5020 printhelp ()
5021 print ("\nlwarpmk: ****** Unknown command \""..arg[1].."\". ******\n")
5022 end
5023
5024 end -- not --version
5025 \end{filecontents*}
5026 % \end{Verbatim}% for syntax highlighting

5027 \end{LWR@createlwarpmk}
```

41 Stacks

for HTML output:

```
5028 \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@depthbook}{-2}
\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}
\let\LWR@depthdescitem\LWR@depthlistitem
```

41.2 Closing actions

A stack to record the action to take to close each nesting level: Add more levels of stack if necessary for a very deeply nested document, adding to `\pushclose` and `\popclose` as well.

```
\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

`\LWR@pushclose` {⟨*sectiontype*⟩}

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

```
\NewDocumentCommand{\LWR@pushclose}{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\csletcs{LWR@closeone}{LWR@printclose#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\csletcs{LWR@closedepthone}{LWR@depth#1}%
}
```

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

```
\newcommand*{\LWR@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@closedepthleven%
\global\let\LWR@closedepthleven\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}{%
    \let\LWR@temp@par\par%
    \let\par\relax%
    \edef\LWR@thisexparrayname{#1#2}%
    \ifstrempty{#3}%
        {\csdef{\LWR@thisexparrayname}{}}%
        {\csedef{\LWR@thisexparrayname}{#3}}%
    \let\par\LWR@temp@par%
}
```

\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 `latexmages` 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 7.6.1.

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

\HomeHTMLFilename

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

```
\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@closedepththree\ \LWR@closedepthtwo\
\LWR@closedepthfour\ \LWR@closedepthfive\ \LWR@closedepthsix\
\LWR@closedepthseven\ \LWR@closedepthnine\ \LWR@closedepheight\
\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

\LWR@popclose
}
```

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

```
\newcommand*{\LWR@closeprevious}[1]{
\LWR@traceinfo{%
    LWR@closeprevious to depth \csuse{LWR@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}{>}{\csuse{LWR@depth#1}}}}%
{%
    \LWR@traceinfo{LWR@closeprevious: closing out depth \LWR@closedepthone}%
    \LWR@closeoneprevious%
}%
\ifboolexpr{test{\ifnumcomp{\LWR@closedepthone}{=}{\csuse{LWR@depth#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*⟩}

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

\thispagestyle {⟨*style*⟩}

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

\markboth {⟨*left*⟩} {⟨*right*⟩}

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

\markright {⟨*right*⟩}

```
5322 \renewcommand*{\markright}[1]{}
```

\raggedbottom

```
5323 \renewcommand*{\raggedbottom}{}
```

\flushbottom

```
5324 \renewcommand*{\flushbottom}{}
```

\sloppy

```
5325 \renewcommand*{\sloppy}{}
```

\fussy

```
5326 \renewcommand*{\fussy}{}
```

\pagenumbering * {⟨*commands*⟩}

```
5327 \RenewDocumentCommand{\pagenumbering}{s m}{}
```

```
5328 \end{warpHTML}
```

51 HTML tags, spans, divs, elements

for HTML output:

```
5329 \begin{warpHTML}
```

51.1 Mapping LaTeX sections to HTML sections

```
5330 \newcommand*{\LWR@tagtitle}{h1}
5331 \newcommand*{\LWR@tagtitleend}{/h1}
5332 \newcommand*{\LWR@tagbook}{div class="book"}
5333 \newcommand*{\LWR@tagbookend}{/div}
5334 \newcommand*{\LWR@tagpart}{h2}
5335 \newcommand*{\LWR@tagpartend}{/h2}
5336 \newcommand*{\LWR@tagchapter}{h3}
5337 \newcommand*{\LWR@tagchapterend}{/h3}
5338 \newcommand*{\LWR@tagsection}{h4}
5339 \newcommand*{\LWR@tagsectionend}{/h4}
5340 \newcommand*{\LWR@tagsubsection}{h5}
5341 \newcommand*{\LWR@tagsubsectionend}{/h5}
5342 \newcommand*{\LWR@tagsubsubsection}{h6}
5343 \newcommand*{\LWR@tagsubsubsectionend}{/h6}
5344 \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{%
```

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

```
\@ifundefined{NoAutoSpacing}%
    {}%
    {%
        \LetLtxMacro\LWR@FBcancel\NoAutoSpacing%
    }%

\@ifundefined{frenchbsetup}%
{}%
{%
    \frenchbsetup{FrenchFootnotes=false}%

%
    \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}%
}
```

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

```
\newcommand*{\LWR@indentHTMLtwo}{%
    \LWR@orignewline\LWR@origrule{4em}{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 s inside a .

⚠ \begin{LWR@nestspan} must follow the opening tag to allow a paragraph to start if the span is at the beginning of a new paragraph.

⚠ \end{LWR@nestspan} must follow the or a <p> may appear inside the span.

```
\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}%
        {\LWR@subhtmlelementclass{span}{inlineminipage}}%
        {\LWR@htmltagc{/span}}%
    \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 /}}%
    \renewenvironment{quote}{\LWR@htmltagc{br /}}{\LWR@htmltagc{br /}}%
    \renewenvironment{quotation}{\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@forceemptyline%
\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@forceemptyline%
```

```
\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 90.

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@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@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@printclosebook}
    {\ifbool{HTMLDebugComments}{\LWR@htmlcomment{Closing book}}{}}
\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.

(The fill seems to be required to force the caption package to create flush left caption text in the HTML.)

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

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 lateximage, 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}
    }%
```

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 MathJax script

for HTML output:

```
\begin{warpHTML}
```

Default: `lwarp_mathjax.txt`

`\LWR@mathjaxfilename` The MathJax script filename to use. This file is copied into the head of each HTML page. This may be changed mid-document using `\MathJaxFilename`, allowing the use of a custom MathJax script, such as for a local repository, or different MathJax script files to be used for different sections of the document.

```
\newcommand*{\LWR@mathjaxfilename}{lwarp_mathjax.txt}
```

`\MathJaxFilename` {⟨*filename*⟩} Assigns the MathJax script file to be used by the following HTML pages.

```
\newcommand*{\MathJaxFilename}[1]{%
    \renewcommand*{\LWR@mathjaxfilename}{#1}%
    \@onelevel@sanitize\LWR@mathjaxfilename%
}

\end{warpHTML}
```

for PRINT output:

```
\begin{warpprint}
\newcommand*{\MathJaxFilename}[1]{}
\end{warpprint}
```

58 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}{\BaseJobname}

\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}
```

59 Footnotes

lwarp uses native LaTeX footnote code, although with its own \box to avoid the LaTeX output routine. The usual functions mostly work as-is.

footnote numbering

To have footnote numbers reset each time footnotes are printed:

```
\setcounter{footnoteReset}{1}
```

For bigfoot, manyfoot, or perpage:

```
\MakePerPage{footnoteX}
— or —
\MakeSortedPerPage{footnoteX}
```

The footnotes are reset when they are printed, according to section level as set by FootnoteDepth, which is not necessarily by HTML page. This is recommended for \alph, \Alph, or \fnsymbol footnotes, due to the limited number of symbols which are available.

footmisc

The footmisc stable option is emulated by lwarp.

⚠ sectioning commands

When using footnotes in sectioning commands, to generate consistent results between print and HTML, use the footmisc package with the stable option, provide a short TOC entry, and \protect the \footnote:

```
\usepackage[stable]{footmisc}
. . .
\subsection[Subsection Name]
    {Subsection Name\protect\footnote{A footnote.}}
```

memoir with footmisc

⚠ memoir

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

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

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

Several kinds of footnotes are used: in a regular page, in a minipage, or as thanks in the titlepage. Each of these is handle differently.

59.1 Regular page footnotes

In HTML documents, footnotes are placed at the bottom of the web page or the section, depending on `FootnoteDepth`, using the LaTeX box `\LWR@footnotebox`. Using this instead of the original `\footins` box avoids having footnotes be printed by the output routine, since footnotes should be printed per HTML page instead of per PDF page.

See section 59.4 for the implementation.

59.2 Minipage footnotes

See section 59.5 for how minipage footnotes are gathered. See section 89.4 for how minipage footnotes are placed into the document.

59.3 Titlepage thanks

See section 66.7 for titlepage footnotes.

59.4 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}
```

Ctr `footnoteReset` Default: `0`

If non-zero, the `footnote` counter is reset to this value each time the footnotes are printed, as controlled by `FootnoteDepth`. For the manyfoot and bigfoot packages, additional counters such as `footnote<suffix>Reset` will be defined as well. These counters may be set non-zero by the user, and are also set if the perpage's `\MakePerPage` or `\MakeSortedPerPage` macros are used for the `footnote` or `footnote<suffix>` counters.

(The name is not capitalized because it is made from the counter's name with "Reset" appended.)

```
\newcounter{footnoteReset}
\setcounter{footnoteReset}{0}
```

```
\end{warpall}
```

for HTML output:

```
\begin{warpHTML}
```

\LWR@footnotebox Patch LaTeX footnotes to use a new \box instead of an insert for lwarp footnotes. This avoids having the original \footins appear at the bottom of a lateximage, which is on its own new page.

```
\newbox\LWR@footnotebox
```

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*⟩} {⟨*footnote box name*⟩}

Factored to allow multiple footnote boxes for manyfoot.

```
\long\def\LWR@@footnotetext#1#2{%
\LWR@traceinfo{LWR@footnotetext}%
\global\setbox\csname #2\endcsname=\vbox{%
```

Add to any current footnotes:

```
    \unvbox\csname #2\endcsname%
```

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%
}%
```

`\LWR@footnotetext` {⟨*text*⟩}

```
\long\def\LWR@footnotetext#1{\LWR@@footnotetext{#1}{LWR@footnotebox}}
```

`\@footnotetext` {⟨*text*⟩}

```
\LetLtxMacro\@footnotetext\LWR@footnotetext
```

59.5 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}}%
        {}%
        {%
            \leavevmode\LWR@orignewline%
            \LWR@htmltagc{/\LWR@tagregularparagraph}%
            \LWR@origpar%
        }%
    \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}
}
```

59.6 Printing pending footnotes

\LWR@@printpendingfootnotes {⟨*footnote counter name*⟩}

```
\newcommand*{\LWR@@printpendingfootnotes}[1]{%
\expandafter\ifvoid\csname LWR@#1box\endcsname\else
    \LWR@forcenewpage
    \begin{BlockClass}{footnotes}
    \null
    \unvbox\csuse{LWR@#1box}
    \setbox\csuse{LWR@#1box}=\vbox{}
    \end{BlockClass}
    \ifltxcounter{#1Reset}{%
        \ifnumgreater{\value{#1Reset}}{0}{%
            \setcounter{#1}{\value{#1Reset}}%
            \addtocounter{#1}{-1}%
        }{}%
    }{}%
\fi
}
```

\LWR@printpendingfootnotes Enclose the footnotes in a class, print, then clear. For manynotes, new footnotes may be added via \appto.

```
\newcommand*{\LWR@printpendingfootnotes}{%
    \LWR@@printpendingfootnotes{footnote}%
}
```

\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}
    \null
    \unvbox\LWR@mpfootnotes
    \setbox\LWR@mpfootnotes=\vbox{}
    \end{BlockClass}
\fi
}
```

```
\end{warpHTML}
```

60 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][]{%
\LWR@stoppars%
\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}
}%
\LWR@startpars%
}
```

\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}
```

61 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*⟩}

\BaseJobname is added to the label in case xr or xr-hyper are used.

```
\newcommand*{\LWR@newautopagelabel}[1]{%
\ifnumequal{\value{LWR@previousautopagelabel}}{\value{page}}%
{}% no action if this autopage label has already been defined
{%
    \label{\BaseJobname-autopage-\arabic{#1}}%
    \setcounter{LWR@previousautopagelabel}{\value{page}}
}%
}
```

61.1 Sanitizing expressions for HTML

Math expressions are converted to lateximages, and some math environments may contain &, <, or >, which should not be allowed inside an HTML <alt> tag, so must convert them to HTML entities.

\LWR@subHTMLsanitize \LWR@strresult must first be set by \LWR@HTMLsanitize, \LWR@HTMLsanitizeexpand, or \CustomizeMathJax.

```
\newcommand{\LWR@subHTMLsanitize}{%
```

The &, <, and > may be interpreted by the browser:

```
    \protect\StrSubstitute{\detokenize\expandafter{\LWR@strresult}}%
        {\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]%
```

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@HTMLsanitize` {⟨*text*⟩}

```
\newrobustcmd{\LWR@HTMLsanitize}[1]{%
```

Cancel French babel character handling, and fully expand the strings:

```
    \begingroup%
    \LWR@FBcancel%
    \fullexpandarg%
    \protect\StrSubstitute{\detokenize{#1}}%
        {\detokenize{\&}}{\detokenize{&}}[\LWR@strresult]%
    \LWR@subHTMLsanitize%
    \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{&}}[\LWR@strresult]%
    \LWR@subHTMLsanitize%
    \LWR@strresult%
    \endgroup%
}
```

61.2 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:

```
\begin{warpMathJax}
\CustomizeMathJax{
    \newcommand{\expval}[1]{\langle#1\rangle}
    \newcommand{\abs}[1]{\lvert#1\rvert}
}
\CustomizeMathJax{\newcommand{\arcsinh}{\text{arcsinh}}}
\CustomizeMathJax{\newcommand{\arccosh}{\text{arccosh}}}
\CustomizeMathJax{\newcommand{\NN}{\mathbb{N}}}
\end{warpMathJax}
```

lwarp already provides MathJax customizations for some packages.

siunitx When using siunitx, a similar process may be used to add custom units:

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\myunit}{\mathrm{WXYZ}}}
\CustomizeMathJax{\newcommand{\umyunit}{\mathrm{\micro\myunit}}}
\end{warpMathJax}
```

```
\newcommand*{\LWR@subcustomizedmathjax}[1]{%
    \begingroup%
    \LWR@FBcancel%
    \fullexpandarg%
    \protect\StrSubstitute{\detokenize{#1}}%
        {\detokenize{\&}}{\detokenize{&}}[\LWR@strresult]%
    \LWR@subHTMLsanitize%
    \xdef\LWR@customizedMathJax{%
        \LWR@customizedMathJax%
            \LWR@strresult%
    }%
    \endgroup%
}

\newcommand*{\CustomizeMathJax}[1]{%
    \appto\LWR@customizedMathJax{\LWRbackslash(}%
    \LWR@subcustomizedmathjax{#1}%
    \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
}
```

Default customizations:

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand\ensuremath[1]{#1}}
\CustomizeMathJax{\newcommand\footnote[2][]{\text{( Footnote #1 )}}}
\CustomizeMathJax{\newcommand\footnotemark[1][]{\text{( Footnote #1 )}}}
\CustomizeMathJax{% absorb two optional arguments
    \newcommand{\LWRframebox}[2][]{\fbox{#2}}
    \newcommand{\framebox}[1][]{\LWRframebox}
}
\CustomizeMathJax{\newcommand{\setlength}[2]{}}
\CustomizeMathJax{\newcommand{\addtolength}[2]{}}
\CustomizeMathJax{\newcommand{\setcounter}[2]{}}
\CustomizeMathJax{\newcommand{\addtocounter}[2]{}}
```

```
\CustomizeMathJax{\newcommand{\cline}[1]{}}
\CustomizeMathJax{\newcommand{\directlua}[1]{\text{(directlua)}}}
\CustomizeMathJax{\newcommand{\luatexdirectlua}[1]{\text{(directlua)}}}
\end{warpMathJax}
```

`\LWR@customizeMathJax` Prints MathJax commands to the HTML output.

```
\newcommand{\LWR@customizeMathJax}{%
\ifbool{mathjax}{

\LWR@stoppars
\LWR@htmlcomment{MathJax customizations:}

\begin{BlockClass}{hidden}
\LWR@stoppars
\LWR@customizedMathJax
\LWR@startpars
\end{BlockClass}

\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}
{}
{


    \ifdefempty{\LWR@pagebottom}{}{%
        \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\MessageBreak
                ``#1'',\MessageBreak
                at the line number listed below,\MessageBreak
                generates the filename\MessageBreak
                ``\LWR@thisfilename'',\MessageBreak
                which appears to be a duplicate.\MessageBreak
                There is a previous section with an\MessageBreak
                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}}
    {}
    {

        \ifdefempty{\LWR@pagetop}{}{
            \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, print the customizations here.

```
\LWR@customizeMathJax

\LWR@traceinfo{LWR@newhtmlfile: done}
}

\end{warpHTML}
```

62 Sectioning

Sectioning and cross-references have been emulated from scratch, rather than try to patch several layers of existing LaTeX code and packages. Formatting is handled by CSS, so the emulated code has much less work to do than the print versions.

Unicode

⚠ 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}
```

62.1 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}
```

62.2 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}
}
```

62.3 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` {⟨*sectiontype*⟩} Stacks the new sectioning level's closing tag, to be used when this section is closed some time later.

⚠ `\LWR@stoppars` must be executed first.

```
\NewDocumentCommand{\LWR@pushoneclose}{m}{%
\LWR@traceinfo{LWR@pushoneclose #1}%
    \LWR@pushclose{#1}%
}
```

`\LWR@startnewdepth` {⟨*sectiontype*⟩}

Closes currently stacked tags of a lesser level, then opens the new nesting level by saving this new sectioning level's closing tag for later use.

⚠ `\LWR@stoppars` must be executed first.

```
\NewDocumentCommand{\LWR@startnewdepth}{m}{%
```

Close any stacked sections up to this new one.

```
\LWR@closeprevious{#1}%
```

Push a new section depth:

```
\LWR@pushoneclose{#1}%
}
```

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}{%
\IfValueTF{#2}%
```

```
    {\LWR@traceinfo{LWR@section: starting #4 #2}}%
    {\LWR@traceinfo{LWR@section: starting #4 #3}}%
\LWR@maybeprintpendingfootnotes{\csuse{LWR@depth#4}}%
\LWR@stoppars%
\LWR@startnewdepth{#4}%
```

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}%
}
```

62.4 Pre- and post- sectioning names

\prebookname
\postbookname

Usually null, but is used by uj* and ut* Japanese classes.

```
\providecommand*{\prebookname}{}
\providecommand*{\postbookname}{}
```

\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
```

62.5 \section and friends

For memoir, a second optional argument is allowed.

For hypbmsec, a second optional argument or either parenthesis argument is allowed.

Each of these additional arguments are for headers or PDF bookmarks, and are ignored for HTML output.

\part * (⟨*2:PDF name*⟩) [⟨*3:TOC name*⟩] [⟨*4:PDF name*⟩] (⟨*5:PDF name*⟩) {⟨*6:name*⟩}

```
\newcommand{\part@preamble}{}% for koma-script

\DeclareDocumentCommand{\part}{s d() o o d() m}{%
    \LWR@section{#1}{#3}{#6}{part}%

```

```
    \part@preamble% for koma-script
    \renewcommand{\part@preamble}{}%
}
```

\chapter * (⟨2:PDF name⟩) [⟨3:TOC name⟩] [⟨4:PDF name⟩] (⟨5:PDF name⟩) {⟨6:name⟩}

```
\let\@printcites\relax% for quotchap package

\newcommand{\chapter@preamble}{}% for koma-script

\@ifundefined{chapter}
{}
{%
    \DeclareDocumentCommand{\chapter}{s d() o o d() m}{%
        \LWR@section{#1}{#3}{#6}{chapter}%

        \@printcites% for quotchap package

        \chapter@preamble% for koma-script
        \renewcommand{\chapter@preamble}{}%
    }
}
```

\section * (⟨2:PDF name⟩) [⟨3:TOC name⟩] [⟨4:PDF name⟩] (⟨5:PDF name⟩) {⟨6:name⟩}

```
\DeclareDocumentCommand{\section}{s d() o o d() m}{%
    \LWR@section{#1}{#3}{#6}{section}%
}
```

\subsection * (⟨2:PDF name⟩) [⟨3:TOC name⟩] [⟨4:PDF name⟩] (⟨5:PDF name⟩) {⟨6:name⟩}

```
\DeclareDocumentCommand{\subsection}{s d() o o d() m}{%
    \LWR@section{#1}{#3}{#6}{subsection}%
}
```

\subsubsection * (⟨2:PDF name⟩) [⟨3:TOC name⟩] [⟨4:PDF name⟩] (⟨5:PDF name⟩) {⟨6:name⟩}

```
\DeclareDocumentCommand{\subsubsection}{s d() o o d() m}{%
    \LWR@section{#1}{#3}{#6}{subsubsection}%
}
```

\paragraph * (⟨2:PDF name⟩) [⟨3:TOC name⟩] [⟨4:PDF name⟩] (⟨5:PDF name⟩) {⟨6:name⟩}

```
\DeclareDocumentCommand{\paragraph}{s d() o o d() m}{%
    \LWR@section{#1}{#3}{#6}{paragraph}%
}
```

\subparagraph * (⟨2:PDF name⟩) [⟨3:TOC name⟩] [⟨4:PDF name⟩] (⟨5:PDF name⟩) {⟨6:name⟩}

```
\DeclareDocumentCommand{\subparagraph}{s d() o o d() m}{%
    \LWR@section{#1}{#3}{#6}{subparagraph}%
}
```

```
\end{warpHTML}
```

63 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{\LWR@mathjaxfilename}%

    \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}
```

64 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
```

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.}
```

```
\ifdefempty{\LWR@firstpagetop}{}{%
    \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}
```

65 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{finished}
```

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:

```
\ifdefempty{\LWR@pagebottom}{}{%
    \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
}
```

`\enddocument` If labels have not changed, mark successful completion of the `lateximages` file. Executed as everything is being shut down.

```
\xpatchcmd{\enddocument}
    {%
        \if@tempswa
        \@latex@warning@no@line{Label(s) may have changed.
        Rerun to get cross-references right}%
```

```
        \fi
    }
    {%
        \if@tempswa
            \@latex@warning@no@line{Label(s) may have changed.
            Rerun to get cross-references right}%
        \else
            \immediate\write\LWR@lateximagesfile{%
                |end|end|end|%
            }%
        \fi
    }
    {}
    {
        \PackageWarningNoLine{lwarp}
        {%
            Could not patch \protect\enddocument.\MessageBreak
            If labels have changed, be sure to recompile before\MessageBreak
            creating lateximages with\MessageBreak
            \space\space lwarpmk limages,\MessageBreak
            or the images may be corrupt%
        }
    }

\end{warpHTML}
```

66 Title page

package support
load order

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

\published and \subtitle

If using the titling package, additional titlepage fields for \published and \subtitle may be added by using \AddSubtitlePublished in the preamble. See section 66.8.

affiliation

lwarp provides for the \author macro an additional \affiliation macro to provide an affiliation and other additional information for each author in the title page. The affiliation information is removed when using titlingpage's \theauthor in the main text.

reusing titlepage information

The titling package maintains the definitions of \thetitle, \theauthor, etc., after the title has been typeset. These commands are to be used to refer to the document's title and author, etc., in the main text. These definitions have the \thanks and \affiliation removed, and for \author the \and is replaced to generate a simple inline list of authors separated by commas. Note: \theauthor does not work well with authblk unless the traditional LaTeX syntax is used.

load order
\theauthor, authblk

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.

66.1 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, and \@date store the values as originally assigned, including any \thanks, \and, or \affiliation. These are low-level macros intended to be used by other macros only inside a titlepage or titlingpage, and are used by \maketitle. The author is printed inside a single-column tabular, which becomes multiple single-column tabulars if multiples authors are included. For HTML, these tabulars become side-by-side 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 \HTMLPageBottom \thetitle, \theauthor, and \thedate are available if titling has been loaded, and are sanitized user-level versions from which have been removed the \thanks and \affiliation, and \and is changed for inline text usage. The author is printed inline without \affiliation or \thanks, with \and placing commas between multiple authors. Thus, these commands are to be used in the main text whenever the user wishes to refer to the document's title and such. One practical use for this is to place the authors at the bottom of each HTML page, such as:

```
\HTMLPageBottom{
  \begin{center}\textcopyright~20xx \theauthor\end{center}
}
```

⚠ \theauthor, authblk \theauthor does not work well if authblk is used. If \theauthor is important, it is recommended to use the standard LaTeX syntax for \author, optionally with lwarp's \affiliation macro as well.

⚠ affiliations After \maketitle has completed, \theauthor retains the definition of the author, but \and is changed to become a comma and a space, intending to print the authors names separated by spaces. This fails when affiliations are included on their own table rows.

\affiliation A solution, provide here, is to define a macro \affiliation which, during \maketitle, starts a new row and adds the affiliation, but after \maketitle is finished \affiliation

is re-defined to discard its argument, thus printing only the author names when \author is later used inline.

66.2 \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}
```

66.3 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}
```

66.4 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}{\LWR@stoppars\@thanks\LWR@startpars}

\end{warpall}
```

66.5 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 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 named oneauthor:

```
\begin{BlockClass}{oneauthor}
\@author
\end{BlockClass}
\end{BlockClass}
}
```

\printdate

```
\newcommand*{\printdate}{%
\begin{BlockClass}{titledate}
\@date
\end{BlockClass}
}
```

```
\end{warpHTML}
```

66.6 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}
```

66.7 \maketitle for HTML output

An HTML of class titlepage is used.

\thanks are a form of footnotes used in the title page. See section 59 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:

```
\LWR@stoppars\@thanks\LWR@startpars
```

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:

```
\LWR@stoppars\@thanks\LWR@startpars
```

Set up special patches:

```
\LWR@maketitlesetup
```

Typeset the title, etc:

```
\@maketitle
```

Immediately generate any \thanks footnotes:

```
\LWR@stoppars\@thanks\LWR@startpars
}
```

```
\end{warpHTML}
```

66.8 \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}
```

67 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}
```

68 Quote and verse

68.1 Attributions

`\attribution` `{`⟨*name*⟩`}`

For use with quote, quotation, verse:

Ex: `"A quotation." \attribution{\textsc{Author Name}\\\textsl{Book Title}}`

for HTML output:

```
\begin{warpHTML}
\newcommand{\attribution}[1]{%
    \LWR@stoppars%
    \begin{BlockClass}{attribution}
    #1
    \end{BlockClass}
    \LWR@startpars%
}
\end{warpHTML}
```

for PRINT output:

```
\begin{warpprint}
\newcommand{\attribution}[1]{
    \begin{flushright}
    \unskip
    #1
    \end{flushright}%
}
\end{warpprint}
```

68.2 Quotes, quotations

for HTML output:

```
\begin{warpHTML}
```

Env `quote`

```
\renewenvironment*{quote}
{
    \LWR@forcenewpage
    \LWR@htmlblocktag{blockquote}
}
{\LWR@htmlblocktag{/blockquote}}
```

Env `quotation`

```
\renewenvironment*{quotation}
{
    \LWR@forcenewpage
    \LWR@htmlblocktag{blockquote}
}
{\LWR@htmlblocktag{/blockquote}}
```

```
\end{warpHTML}
```

68.3 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.

68.3.1 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}
```

68.3.2 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}
```

69 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 `latexi`mages do not yet work inside the environment.

⚠ math in `tabbing` If math is used inside `tabbing`, place `tabbing` inside a `latexi`mage 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}
```

70 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}
```

71 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.

71.1 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{listitem}%
\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{list}%
\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{list}%
\LWR@startpars%
}
```

71.2 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{listitem}%
\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%
}
```

71.3 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%
}
```

71.4 Description

\LWR@descitem [⟨*label*⟩] Handles an \item inside a description.

```
\newcommand*{\LWR@descitem}[1][]%
{%
\LWR@stoppars%
\LWR@setlatestname{#1}%
\LWR@startnewdepth{descitem}%
```

While creating the label, encase it inside tags and disable \hspace, which is used by the standard classes to add space to the labels.

```
\begingroup%
\let\LWR@orig@desc@makelabel\makelabel
\renewcommand*{\makelabel}[1]{%
    \LWR@htmltag{dt}%
```

```
    \LWR@orig@desc@makelabel{#1}%
    \LWR@htmltag{/dt}%
}
\LWR@select@html@nohspace%
\LWR@origitem[#1]%
\endgroup%
\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%
}
```

71.5 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}
```

72 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.

72.1 Limitations

Tabular mostly works as expected, but pay special attention to the following, especially if working with environments, macros inside tabulars, multirows, siunitx `S` columns, or the packages multirow, longtable, supertabular, or xtab.

Defining macros and environments:

⚠ Misplaced alignment tab character &

- When defining environments or macros which include `tabular` and instances of the `&` character, it may be necessary to make `&` active before the environment or macro is defined, then restore `&` to its default catcode after, using the following commands. These are are ignored in print mode.

```
\StartDefiningTabulars
<define macros or environments using tabular and & here>
\StopDefiningTabulars
```

⚠ floatrow

This includes before and after defining any macro which used \ttabbox from floatrow.

⚠ tabular inside another environment

- When creating a new environment which contains a tabular environment, lwarp's emulation of the tabular does not automatically resume when the containing enviroment ends, resulting in corrupted HTML rows. To fix this, use \ResumeTabular as follows. This is ignored in print mode.

```
\StartDefiningTabulars % because & is used in a definition
\newenvironment{outerenvironment}
{
\tabular{cc}
left & right \\
}
{
\TabularMacro\ResumeTabular
left & right \\
\endtabular
}
\StopDefiningTabulars
```

Cell contents:

⚠ macro in a table

- Using a custom macro inside a tabular data cell may result in an extra HTML data cell tag, corrupting the HTML table. To avoid this, use \TabularMacro just before the macro. This is ignored in print mode.

```
\TabularMacro\somemacro & more row contents \\
```

Column specifiers:

@ 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. When given an optional width, each cell is styled to create the custom border. Without an optional width, the entire row is given a class to assign the standard border.

combined rules

- If you wish to use \cmidrule followed by \bottomrule, it may be necessary to use:

```
\cmidrule{2-3} \\[-2ex]
\bottomrule
```

The optional -2ex is ignored in HTML, but improves the visual formatting in the print output.

⚠ \warpprintonly
⚠ Misplaced \noalign

- For \toprule and \bottomrule, when combined with a warpprint or warpHTML environment, if a "Misplaced \noalign" error occurs, change

  ```
  This & That \endhead
  ```

  to

  ```
  \warpprintonly{This & That \endhead}
  ```

  and likewise with the other \end headings. Keep the \endfirsthead row unchanged, as it is still relevent to HTML output.

Other:

- tabularx ignores the width, but X columns do produce paragraph columns or multicolumns.

longtable headings

- For longtable, place headings and footings which do not apply to HTML inside \warpprintonly{}.

⚠ S columns

- For S columns (from the siunitx package), while producing print output, anything non-numeric must be placed inside { } braces, including commands such as \multirow. While producing HTML output, though, anything placed inside braces is not seen by lwarp's tabular handling algorithm. To resolve this problem, make a copy of the row, with one version for print output, containing the extra braces, and another version for HTML output, without the extra braces, such as:

  ```
  \warpprintonly{1 & 2 & {\multirow{2}{2cm}{Text}} & 3 \\}
  \warpHTMLonly{1 & 2 & \multirow{2}{2cm}{Text} & 3 \\}
  ```

⚠ tabular inside a

- In LaTeX, a tabular may be placed inside a minipage, but in HTML a <table> may not be inside a . If this situation is detected, a warning is printed instructing the user to isolate the using \warpprintonly or the warpprint environment.

for HTML output:

```
\begin{warpHTML}
```

72.2 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.

72.2.1 arydshln

Emualated by the original LaTeX non-dashed versions.

```
\LetLtxMacro\hdashline\hline
\LetLtxMacro\cdashline\cline
\LetLtxMacro\firsthdashline\hline
\LetLtxMacro\lasthdashline\hline
```

72.3 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 it.

\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
}
```

72.4 Tabular variables

In order to support nested tabulars, each of these is used locally. For local counters, etoolbox's \defcounter and lwarp's new \defaddtocounter are used.

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@usedmultirow Used to error if used \multirow or \multicolumnrow without using \mrowcell or \mcolrowcell.

```
\newbool{LWR@usedmultirow}
```

Bool LWR@foundmrowcell Used to error if used \multirow or \multicolumnrow without using \mrowcell or \mcolrowcell.

```
\newbool{LWR@foundmrowcell}
```

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}
```

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}
```

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}
```

Used to add a style to a table data cell:

```
\newbool{LWR@tdhavecellstyle}
```

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.

Ctr LWR@cellcolordepth — Counts how many cell color s were added to the current tabular data cell.

```
\newcounter{LWR@cellcolordepth}
```

72.4.1 Multicolumn variables

```
\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}
```

72.4.2 Longtable variables

Bool `LWR@starredlongtable` Per the caption pacakge, step the counter if `longtable*`.

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

72.4.3 Midrule variables

Ctr `LWR@midrulecounter` Indexes across the `LWR@midrules` and `LWR@trim<l/r>rules` data arrays.

```
\newcounter{LWR@midrulecounter}
```

72.5 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}{%
    \booltrue{LWR@intabularmetadata}%
    \ifbool{LWR@exitingtabular}%
    {%

        \LWR@stoppars%
    }%
    {% not exiting tabular
      \ifboolexpr{bool{LWR@skippingmrowcell} or bool{LWR@skippingmcolrowcell}}%
        {%

            \LWR@stoppars%
```

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:

```
            \LWR@stoppars%
            \boolfalse{LWR@tableparcell}%
```

Close the table data cell.

Close any color s.

```
         \whileboolexpr{test {\ifnumcomp{\value{LWR@cellcolordepth}}{>}{0}}}{%
                \LWR@htmltag{/div}\LWR@orignewline%
                \defaddtocounter{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
    \boolfalse{LWR@skippingmrowcell}%
    \boolfalse{LWR@skippingmcolrowcell}%
    \boolfalse{LWR@skipatbang}%
```

Color control. Column is set by >{} for each cell, so it must be cleared here.

```
    \def\LWR@cellHTMLcolor{}%
    \def\LWR@columnHTMLcolor{}%
    \defcounter{LWR@cellcolordepth}{0}%
}
```

When not used inside a tabular, & performs its original function as recorded here (with catcode 4).

```
\let\LWR@origampmacro&
```

```
\end{warpHTML}
```

72.5.1 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.

```
        \defaddtocounter{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.

72.5.2 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}{%
        \booltrue{LWR@tabularmutemods}%
    }{%
        \boolfalse{LWR@tabularmutemods}%
    }%
```

Locally reenable the table data tags until finished with the final row:

```
    \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%
        \defaddtocounter{LWR@tableLaTeXcolindex}{1}%
        \global\boolfalse{LWR@tabularcelladded}%
        \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}{%
        \booltrue{LWR@exitingtabular}%
    }{%
        \boolfalse{LWR@exitingtabular}%
    }%
    \boolfalse{LWR@tabularmutemods}%

    \boolfalse{LWR@emptyatbang}%
    }{}% ifboolexpr
}
```

72.6 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:

```
    \booltrue{LWR@intabularmetadata}%
```

Not yet started a table row:

```
    \boolfalse{LWR@startedrow}%
```

Additional setup:

```
    \defcounter{LWR@hlines}{0}%
    \defcounter{LWR@hdashedlines}{0}%
    \boolfalse{LWR@doingtbrule}%
    \boolfalse{LWR@doingcmidrule}%
    \LWR@clearmidrules%

    \def\LWR@rowHTMLcolor{}%
```

Start at first column:

```
    \defcounter{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%
}
```

72.7 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%
}
```

72.8 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}%
    \defaddtocounter{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}%
            {\expandafter\@firstofone\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}}%
            {\expandafter\@firstofone\LWR@colparameter}%
        \LWR@traceinfo{at \arabic{LWR@tabletotalLaTeXcols}: %
        \LWR@getexparray{LWR@colatspec}{\arabic{LWR@tabletotalLaTeXcols}}}%
    }%
    \let\LWR@colparameter\relax%
    \booltrue{LWR@validtablecol}%
}
```

\LWR@parsebangcolumn Handles !{text} columns.

```
\newcommand*{\LWR@parsebangcolumn}{%
```

Move to the next token after the '!':

```
    \LWR@traceinfo{bang column}%
    \defaddtocounter{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}%
            {\expandafter\@firstofone\LWR@colparameter}%
    }%
    {% not at the left edge:
        \LWR@traceinfo{not at the left edge}%
        \LWR@setexparray{LWR@colbangspec}%
```

```
            {\arabic{LWR@tabletotalLaTeXcols}}%
            {\expandafter\@firstofone\LWR@colparameter}%
      \LWR@traceinfo{bang \arabic{LWR@tabletotalLaTeXcols}: \LWR@colparameter!}%
    }%
    \let\LWR@colparameter\relax%
    \booltrue{LWR@validtablecol}%
}
```

`\LWR@parsebeforecolumn` Handles `>{text}` columns.

```
\newcommand*{\LWR@parsebeforecolumn}{%
```

Move to the next token after the '>':

```
    \defaddtocounter{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}}%
        {\expandafter\@firstofone\LWR@colparameter}%
    \let\LWR@colparameter\relax%
    \booltrue{LWR@validtablecol}%
}
```

`\LWR@parseaftercolumn` Handles `<{text}` columns.

```
\newcommand*{\LWR@parseaftercolumn}{%
```

Move to the next token after the '<':

```
    \defaddtocounter{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}}%
        {\expandafter\@firstofone\LWR@colparameter}%
```

```
    \let\LWR@colparameter\relax%
    \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}%
        }%
    }%
    \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}}%
    }%
    \booltrue{LWR@validtablecol}%
}
```

\LWR@parsesemicoloncolumn Handles vertical rules.

```
\newcommand*{\LWR@parsesemicoloncolumn}{%
```

Treat ; as a : column:

```
    \LWR@parsecoloncolumn%
```

Skip the following width token:

```
    \defaddtocounter{LWR@tablecolspecindex}{1}%
}
```

72.9 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]{%
    \defaddtocounter{LWR@tabletotalLaTeXcols}{1}%
    \defaddtocounter{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}}{}%
    \booltrue{LWR@validtablecol}%
}
```

72.10 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:

```
    \defaddtocounter{LWR@tablecolspecindex}{1}%
}
```

72.11 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:

```
    \defaddtocounter{LWR@tablecolspecindex}{2}%
}
```

72.12 Parsing '*' columns

\LWR@parsestarcolumn Star columns should already have been expanded, so this should never be used.

```
\newcommand*{\LWR@parsestarcolumn}{%
    \defaddtocounter{LWR@tablecolspecindex}{2}%
}
```

72.13 Parsing 'D' columns

From the dcolumn package.

\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.

```
    \defaddtocounter{LWR@tablecolspecindex}{3}%
}
```

72.14 Expanding the star column specifications

Ctr LWR@starcount — Internal count for duplicating star columns.

```
\newcount\LWR@starcount
```

\LWR@expcolspec — Temporary storage used to build the expanded column specifier.

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

Ctr LWR@splitstarindex — Indexes into the column specifiers.

```
\newcounter{LWR@splitstarindex}
```

Ctr LWR@splitstarcopies — Number of copies.

```
\newcounter{LWR@splitstarcopies}
```

\LWR@splitstarcontents — Contents to duplicate.

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

\expandcolspec — Expands \LWR@origcolspec for star columns.

```
\newcommand*{\expandcolspec}{%
```

Find the position of any star token.

```
    \StrPosition{\LWR@origcolspec}{*}[\LWR@tempone]%
```

Expand until no stars are found:

```
    \whileboolexpr{ test {\ifnumgreater{\LWR@tempone}{0}}}%
    {%
```

Begin with any characters to the left of the star.

```
        \setcounter{LWR@splitstarindex}{\LWR@tempone}%
        \addtocounter{LWR@splitstarindex}{-1}%
      \StrLeft{\LWR@origcolspec}{\value{LWR@splitstarindex}}[\LWR@expcolspec]%
```

Move past the star to remember its number of copies.

```
        \addtocounter{LWR@splitstarindex}{2}%
        \StrChar{\LWR@origcolspec}{\value{LWR@splitstarindex}}[\LWR@tempone]%
       \setcounter{LWR@splitstarcopies}{\expandafter\@firstofone\LWR@tempone}%
```

Move past the number of copies and remember the contents.

```
        \addtocounter{LWR@splitstarindex}{1}%
     \StrChar{\LWR@origcolspec}{\value{LWR@splitstarindex}}[\LWR@splitstarcontents]%
```

Table 10: Tabular baseline

l	p	m	b	r
			bot	
		mid	bot	
l	par	mid	bot	r
	par	mid		
	par			

For each copy, append the contents.

```
7814        \ifnumgreater{\value{LWR@splitstarcopies}}{0}%
7815        {%
7816            \LWR@starcount=\value{LWR@splitstarcopies}%
7817            \loop
7818                \appto\LWR@expcolspec{%
```

(Remove the enclosing braces.)

```
7819                    \expandafter\@firstofone\LWR@splitstarcontents%
7820                }%
7821                \advance \LWR@starcount -1
7822            \ifnum \LWR@starcount>0 \repeat
7823        }{}%
```

Remove any token to the left, and append the rightmost remainding tokens.

```
7824      \StrGobbleLeft{\LWR@origcolspec}{\value{LWR@splitstarindex}}[\colspecremainder]%
7825        \appto{\LWR@expcolspec}{\colspecremainder}%
```

Remember the final result.

```
7826        \edef\LWR@origcolspec{\LWR@expcolspec}%
```

See if more stars exist.

```
7827        \StrPosition{\LWR@origcolspec}{*}[\LWR@tempone]%
7828    }%
7829 }
```

72.15 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.

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.

`\LWR@parsetablecols` {⟨*colspecs*⟩}

Scans the column specification left to right.

Builds `\LWR@tablecolspec` with the final specification, one LaTeX column per entry. The final number of LaTeX columns in each row is stored in `LWR@tabletotalLaTeXcols`, which is the number of `&` and `\\` in each line, but which does not include `@`, `!`, `<`, `>` specifications in the count.

```
\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]%
```

Expand any star columns:

```
    \expandcolspec%
```

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:

```
    \defcounter{LWR@tabletotalLaTeXcols}{0}%
    \defcounter{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:

```
    \defcounter{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}%
    \defcounter{LWR@tablecolspecwidth}{\LWR@strresult}%
```

Haven't seen any optional arguments so far

```
    \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:

```
    \boolfalse{LWR@validtablecol}%
```

Skip over any optional arguments, such as siunitx `S` column:

```
    \IfStrEq{\LWR@strresult}{[}{\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}{}%
```

A star column:

```
    \IfStrEq{\LWR@strresult}{*}{\LWR@parsestarcolumn}{}%
```

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}{]}{\boolfalse{LWR@opttablecol}}{}%
```

Move to the next character:

```
    \defaddtocounter{LWR@tablecolspecindex}{1}%
    }% whiledo
}%
```

72.16 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}
```

72.17 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:

```
        \booltrue{LWR@startedrow}%
```

Create the row tag, with a class if necessary.

```
        \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
}
```

72.18 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}%
}
```

72.19 Printing @ or ! 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
}
```

72.20 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}%
                 \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}}%
             }%
             \boolfalse{LWR@intabularmetadata}%
         }% not exiting tabular
     }{}% in tabular metadata
     \LWR@traceinfo{LWR@tabledatasinglecolumntag: done}%
}%
```

72.21 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.

Len \LWR@heavyrulewidth The default width of the rule.

```
\newlength{\LWR@heavyrulewidth}
\setlength{\LWR@heavyrulewidth}{.08em}
```

Len \LWR@lightrulewidth The default width of the rule.

```
\newlength{\LWR@lightrulewidth}
\setlength{\LWR@lightrulewidth}{.05em}
```

Len \LWR@cmidrulewidth The default width of the rule.

```
\newlength{\LWR@cmidrulewidth}
\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.

```
\newlength{\LWR@thiscmidrulewidth}
\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.

```
\newcommand*{\LWR@clearmidrules}
{%
    \defcounter{LWR@midrulecounter}{1}%
    \whileboolexpr{%
        not test{%
            \ifnumcomp{\value{LWR@midrulecounter}}{>}%
                {\value{LWR@tabletotalLaTeXcols}}%
        }%
    }%
    {%
        \LWR@setexparray{LWR@midrules}{\arabic{LWR@midrulecounter}}{0pt}%
        \setlength{\LWR@thiscmidrulewidth}{\LWR@cmidrulewidth}%
        \LWR@setexparray{LWR@trimlrules}{\arabic{LWR@midrulecounter}}{}%
        \LWR@setexparray{LWR@trimrrules}{\arabic{LWR@midrulecounter}}{}%
        \LWR@setexparray{LWR@cdashlines}{\arabic{LWR@midrulecounter}}{N}%
        \defaddtocounter{LWR@midrulecounter}{1}%
    }%
}
```

\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]{%
    \defcounter{LWR@midrulecounter}{#3}%
    \whileboolexpr{%
        not test {%
            \ifnumcomp{\value{LWR@midrulecounter}}{>}{#4}%
        }%
    }%
    {%
        \LWR@setexparray{LWR@midrules}{\arabic{LWR@midrulecounter}}{#1}%
        \defaddtocounter{LWR@midrulecounter}{1}%
    }% whiledo
    \IfSubStr{#2}{l}{\LWR@setexparray{LWR@trimlrules}{#3}{l}}{}%
    \IfSubStr{#2}{r}{\LWR@setexparray{LWR@trimrrules}{#4}{r}}{}%
    \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]{%
    \defcounter{LWR@midrulecounter}{#1}%
    \whileboolexpr{%
        not test {%
            \ifnumcomp{\value{LWR@midrulecounter}}{>}{#2}%
        }%
    }%
    {%
        \LWR@setexparray{LWR@cdashlines}{\arabic{LWR@midrulecounter}}{Y}%
        \defaddtocounter{LWR@midrulecounter}{1}%
    }% whiledo
    \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}
```

\LWR@tdstartstyles Begins possibly adding a table data cell style.

```
\newcommand*{\LWR@tdstartstyles}{\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}%
        {%
            "%
            \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 % space
```

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}}{}%
    }{}%
}
```

72.22 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%
        }{}%
    }%
}
```

`\LWR@subaddtabularcellcolor` {⟨*HTML color*⟩}

```
\newcommand*{\LWR@subaddtabularcellcolor}[1]{%
    \LWR@htmltag{div class="cellcolor" style="%
        background:\LWR@origpound{}{}#1 %
    " }%
    \defaddtocounter{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}}%
}
```

72.23 Multicolumns

72.23.1 Parsing multicolumns

`\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}%
            {\defaddtocounter{LWR@mcolvertbarsl}{1}}% left edge
            {\defaddtocounter{LWR@mcolvertbarsr}{1}}% not left edge
    }%
    {%
        \IfStrEq{\LWR@strresult}{:}%
        {%
            \ifbool{LWR@mcolvertbaronleft}%
                {\defaddtocounter{LWR@mcolvertbarsldash}{1}}% left edge
                {\defaddtocounter{LWR@mcolvertbarsrdash}{1}}% not left edge
        }%
        {%
            \IfStrEq{\LWR@strresult}{;}%
            {%
                \ifbool{LWR@mcolvertbaronleft}%
                    {\defaddtocounter{LWR@mcolvertbarsldash}{1}}% left edge
                   {\defaddtocounter{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%
    \defaddtocounter{LWR@tablemulticolspos}{#1}%
```

```
8402     \LWR@stoppars%
8403 }
```

`\LWR@multicolother` {⟨*colspec*⟩} For `@`, `!`, `>`, `<`, print the next token without paragraph tags:

```
8404 \newcommand*{\LWR@multicolother}[1]{%
8405     \defaddtocounter{LWR@tablemulticolspos}{1}%
8406     \StrChar{#1}{\arabic{LWR@tablemulticolspos}}[\LWR@strresult]%
8407     \LWR@strresult%
```

A valid column data type was found:

```
8408     \booltrue{LWR@validtablecol}%
8409 }
```

`\LWR@multicolskip` Nothing to print for this column type.

```
8410 \newcommand*{\LWR@multicolskip}{%
```

A valid column data type was found:

```
8411     \booltrue{LWR@validtablecol}%
8412 }
```

`\LWR@printmccoldata` {⟨*colspec*⟩} Print the data for any valid column type found.

```
8413 \newcommand*{\LWR@printmccoldata}[1]{%
8414     \LWR@traceinfo{lwr@printmccoldata -#1}%
```

Not yet found a valid column type:

```
8415     \boolfalse{LWR@validtablecol}%
```

Get one token of the column spec, into a local copy in case nested.

```
8416     \StrChar{#1}{\arabic{LWR@tablemulticolspos}}[\LWR@strresult]%
8417     \edef\LWR@printmccoldatatoken{\LWR@strresult}%
```

Print the text depending on which column type is found. Also handles `@`, `>`, `<` as it comes to them.

```
8418     \IfStrEq{\LWR@printmccoldatatoken}{l}{\LWR@multicoltext}{}%
8419     \IfStrEq{\LWR@printmccoldatatoken}{c}{\LWR@multicoltext}{}%
8420     \IfStrEq{\LWR@printmccoldatatoken}{r}{\LWR@multicoltext}{}%
8421     \IfStrEq{\LWR@printmccoldatatoken}{D}{%
8422         \defaddtocounter{LWR@tablemulticolspos}{3}% skip parameters
8423         \LWR@multicoltext%
8424     }{}%

8425     \IfStrEq{\LWR@printmccoldatatoken}{p}{\LWR@multicolpartext{2}}{}%
8426     \IfStrEq{\LWR@printmccoldatatoken}{m}{\LWR@multicolpartext{2}}{}%
```

```
    \IfStrEq{\LWR@printmccoldatatoken}{b}{\LWR@multicolpartext{2}}{}%
    \IfStrEq{\LWR@printmccoldatatoken}{P}{\LWR@multicolpartext{2}}{}%
    \IfStrEq{\LWR@printmccoldatatoken}{M}{\LWR@multicolpartext{2}}{}%
    \IfStrEq{\LWR@printmccoldatatoken}{B}{\LWR@multicolpartext{2}}{}%

    \IfStrEq{\LWR@printmccoldatatoken}{w}{\LWR@multicolpartext{3}}{}%
    \IfStrEq{\LWR@printmccoldatatoken}{W}{\LWR@multicolpartext{3}}{}%

    \IfStrEq{\LWR@printmccoldatatoken}{S}{\LWR@multicoltext}{}%
    \IfStrEq{\LWR@printmccoldatatoken}{s}{\LWR@multicoltext}{}%

    \IfStrEq{\LWR@printmccoldatatoken}{X}{\LWR@multicolpartext{1}}{}%
    \IfStrEq{\LWR@printmccoldatatoken}{|}{\LWR@multicolskip}{}%
    \IfStrEq{\LWR@printmccoldatatoken}{:}{\LWR@multicolskip}{}%
    \IfStrEq{\LWR@printmccoldatatoken}{;}{%
        \LWR@multicolskip%
        \defaddtocounter{LWR@tablemulticolspos}{1}% skip parameter
    }{}%
   \IfStrEq{\LWR@printmccoldatatoken}{\detokenize{@}}{\LWR@multicolother{#1}}{}%
   \IfStrEq{\LWR@printmccoldatatoken}{\detokenize{!}}{\LWR@multicolother{#1}}{}%
   \IfStrEq{\LWR@printmccoldatatoken}{\detokenize{>}}{\LWR@multicolother{#1}}{}%
   \IfStrEq{\LWR@printmccoldatatoken}{\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]{%
    \defcounter{LWR@tablemulticolspos}{1}%
    \StrLen{#1}[\LWR@strresult]%
    \defcounter{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:

```
        \defaddtocounter{LWR@tablemulticolspos}{1}%
    }%
}
```

72.23.2 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%
        \booltrue{LWR@validtablecol}%
    }%
```

Compute the rightmost column to be included. This is used to create the right trim.

```
    \defcounter{LWR@lastmulticolumn}{\value{LWR@tableLaTeXcolindex}}%
    \defaddtocounter{LWR@lastmulticolumn}{#3}%
    \defaddtocounter{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:

```
        \defcounter{LWR@mcolvertbarsl}{0}%
        \defcounter{LWR@mcolvertbarsr}{0}%
        \defcounter{LWR@mcolvertbarsldash}{0}%
        \defcounter{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:

```
        "%


        \LWR@tdstartstyles%


        \LWR@addtabularrowcolor%


        \LWR@addcmidrulewidth%
        \LWR@addcdashline%
        \LWR@addtabularhrulecolor%
        \LWR@addmulticolvertrulecolor%
        \LWR@addformatwpalignment{#5}%
        \LWR@tdendstyles%
    }% end of the opening table data tag
```

```
    \boolfalse{LWR@intabularmetadata}%
    \LWR@parsemulticolumnalignment{#5}{\LWR@printmccoldata}%
}
```

72.23.3 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:

```
    \defaddtocounter{LWR@tableLaTeXcolindex}{#1}%
    \defaddtocounter{LWR@tableLaTeXcolindex}{-1}%
```

Skip any trailing `@` or `!` columns for this cell:

```
    \booltrue{LWR@skipatbang}%
}
```

72.23.4 Longtable captions

longtable captions use `\multicolumn`.

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:

```
    \booltrue{LWR@skipatbang}%
    }% end of \LWR@domulticolumn
    \defaddtocounter{LWR@tableLaTeXcolindex}{\value{LWR@tabletotalLaTeXcols}}%
    \defaddtocounter{LWR@tableLaTeXcolindex}{-1}

}
```

72.23.5 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}%
        {}%
        {\defaddtocounter{LWR@tabhtmlcoltotal}{1}}%
```

Likewise for the `!` columns:

```
    \edef\LWR@atbangspec{\LWR@getexparray{LWR@colbangspec}{#1}}%
    \ifdefempty{\LWR@atbangspec}%
        {}%
        {\defaddtocounter{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:

```
    \defcounter{LWR@tabhtmlcolindex}{#1}%
    \defcounter{LWR@tabhtmlcoltotal}{#2}%
    \defcounter{LWR@tabhtmlcolend}{#1}%
    \defaddtocounter{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}}%
        \defaddtocounter{LWR@tabhtmlcolindex}{1}%
    }% whiledo
}
```

```
\end{warpHTML}
```

72.24 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}
```

72.25 Multicolumnrow

A print-mode version is defined here, and is also used during HTML output while inside a `latexmage`.

See section 346 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}
```

72.26 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}
```

72.27 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 8.10.1.

```
\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}{%
    \boolfalse{LWR@exitingtabular}%
```

```
    \boolfalse{LWR@tabularmutemods}%
    \LWR@getmynexttoken%
}
\end{warpHTML}
```

for PRINT output:

```
\begin{warpprint}
\newcommand*{\ResumeTabular}{}
\end{warpprint}
```

72.28 Checking for a new table cell

for HTML output:

```
\begin{warpHTML}
```

`\LWR@tabledatacolumntag` Open a new HTML table cell unless the next token is for a macro which does not create data, such as `\hline`, `\toprule`, etc:

```
\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}%
        {\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}
```

72.29 \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}
```

72.30 \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}
```

72.31 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. memoir may have already loaded booktabs.

```
\providecommand*{\toprule}[1][]{\hline}
\providecommand*{\midrule}[1][]{\hline}
\providecommand*{\cmidrule}{\cline}
\providecommand*{\bottomrule}[1][]{\hline}
\providecommand*{\addlinespace}[1][]{}
\providecommand*{\morecmidrules}{}
\providecommand*{\specialrule}[3]{\hline}
```

\noalign {⟨*text*⟩} Redefined for use inside tabular.

```
\LetLtxMacro\LWR@orignoalign\noalign

\newcommand{\LWR@tabularnoalign}[1]{%
    \advance\rownum\m@ne%
    \LetLtxMacro\LWR@save@xcolorrowHTMLcolor\LWR@xcolorrowHTMLcolor%
    \renewcommand*{\LWR@xcolorrowHTMLcolor}{}%
    \multicolumn{\value{LWR@tabletotalLaTeXcols}}{l}{#1} \\
    \LetLtxMacro\LWR@xcolorrowHTMLcolor\LWR@save@xcolorrowHTMLcolor%
    % \@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{
```

```
8828
8829 \@ifpackageloaded{lwarp-tabls}
8830 {
8831     \newcommand*{\LWR@HTMLhline}[1][]{%
8832         \ifbool{FormatWP}%
8833             {\LWR@docmidrule{1-\arabic{LWR@tabletotalLaTeXcols}}}%
8834             {\defaddtocounter{LWR@hlines}{1}}%
8835         \LWR@getmynexttoken}%
8836 }
8837 {
8838     \newcommand*{\LWR@HTMLhline}{%
8839         \ifbool{FormatWP}%
8840             {\LWR@docmidrule{1-\arabic{LWR@tabletotalLaTeXcols}}}%
8841             {\defaddtocounter{LWR@hlines}{1}}%
8842         \LWR@getmynexttoken}%
8843 }
8844
8845 }% AtBeginDocument
```

`\LWR@HTMLcline` {⟨*columns*⟩}

```
8846 \NewDocumentCommand{\LWR@HTMLcline}{m}%
8847     {\LWR@docmidrule{#1}\LWR@getmynexttoken}%
```

`\LWR@tabular@warpprintonly` {⟨*contents*⟩}

Only process the contents if producing printed output. Modified inside a `tabular` to grab the next token.

```
8848 \newcommand{\LWR@tabular@warpprintonly}[1]{%
8849     \ifbool{warpingprint}{#1}{}%
8850     \LWR@getmynexttoken%
8851 }
```

`\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.

```
8852 \AtBeginDocument{
8853 \@ifundefined{NoAutoSpacing}%
8854 {% no babel-french
8855     \newcommand*{\LWR@nullifyNoAutoSpacing}{}
8856 }% no babel-french
8857 {% yes babel-french
8858     \newcommand*{\LWR@nullifyNoAutoSpacing}{%
8859         \NoAutoSpacing%
8860         \renewcommand*{\NoAutoSpacing}{}%
8861         \renewcommand*{\LWR@FBcancel}{}%
8862     }
8863 }% yes babel-french
8864 }% 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}%
```

⚠ <table> inside

In LaTeX, a tabular may be placed inside a minipage, but in HTML a <table> may not be inside a . Since there may be several nested s, with an unknown number of other objects between, it is hard to undo all these s before the <table> then redo them after. The broswer probably compensates for this situation, but formatting may be lost inside the <table> because several things are neutralized inside a . Furthermore, in the HTML output, the entire <table> is placed on a single line of HTML code, since the line breaking commands are neutralized inside a . Since this is such a sloppy situation, a warning is issued here instructing the user to please isolate the to print-only.

```
    \ifnumcomp{\value{LWR@spandepth}}{>}{0}{%
        \PackageWarning{lwarp}{%
            A tabular is being used inside a span such as\MessageBreak
            a minipage. Some formatting may not be correct\MessageBreak
            in the tabular.\MessageBreak
        It is recommended to use \protect\warpprintonly\space or the\MessageBreak
            warpprint environment to isolate the span to\MessageBreak
            print-only,%
        }
    }{}%
    \addtocounter{LWR@tabulardepth}{1}%
```

Not yet started a table row:

```
    \boolfalse{LWR@startedrow}%
```

Not yet doing any rules:

```
    \defcounter{LWR@hlines}{0}%
    \defcounter{LWR@hdashedlines}{0}%
    \boolfalse{LWR@doingtbrule}%
    \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.

```
    \boolfalse{LWR@exitingtabular}%
    \boolfalse{LWR@tabularmutemods}%
```

Error if failed to use \mrowcell or \mcolrowcell when needed.

```
    \boolfalse{LWR@usedmultirow}%
    \boolfalse{LWR@foundmrowcell}%

    \renewcommand*{\LWR@multicoltext}{}%
```

Create the `table` tag:

```
    \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 #:

```
    \defcounter{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}%
```

`\warpprintonly` inside a `tabular` must grab the next token.

```
    \LetLtxMacro\warpprintonly\LWR@tabular@warpprintonly%
```

The following adjust for colortbl.

```
    \LetLtxMacro\arrayrulecolor\arrayrulecolornexttoken%
    \LetLtxMacro\doublerulesepcolor\doublerulesepcolornexttoken%
    \def\LWR@columnHTMLcolor{}%
    \def\LWR@rowHTMLcolor{}%
    \def\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 s:

```
    \defcounter{LWR@cellcolordepth}{0}%
```

The following may appear before a data cell is created, so after doing their actions, we look ahead with \LWR@getmynextoken to see if the next token might create a new data cell:

The optional parameter for \hline supports the tabls package.

```
    \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}}}%
            {\defaddtocounter{LWR@hdashedlines}{1}}%
        \LWR@getmynexttoken%
    }%

    \DeclareDocumentCommand{\cdashline}{m}{%
        \LWR@docdashline{##1}\LWR@getmynexttoken%
    }%

    \DeclareDocumentCommand{\firsthdashline}{o}{%
        \ifbool{FormatWP}%
            {\LWR@docdashline{1-\arabic{LWR@tabletotalLaTeXcols}}}%
            {\defaddtocounter{LWR@hdashedlines}{1}}%
        \LWR@getmynexttoken%
    }%

    \DeclareDocumentCommand{\lasthdashline}{o}{%
        \ifbool{FormatWP}%
            {\LWR@docdashline{1-\arabic{LWR@tabletotalLaTeXcols}}}%
            {\defaddtocounter{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%
        \booltrue{LWR@skippingmrowcell}%
        \booltrue{LWR@foundmrowcell}%
```

```
    }%
    \renewcommand*{\mcolrowcell}{%
        \LWR@maybenewtablerow%
        \booltrue{LWR@skippingmcolrowcell}%
        \booltrue{LWR@foundmrowcell}%
    }%
    \LetLtxMacro\caption\LWR@longtabledatacaptiontag%
```

Reset for new processing:

```
    \boolfalse{LWR@tableparcell}%
    \boolfalse{LWR@skippingmrowcell}%
    \boolfalse{LWR@skippingmcolrowcell}%
    \boolfalse{LWR@skipatbang}%
    \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}%
    \boolfalse{LWR@intabularmetadata}%
```

Unnest one level of tabular:

```
    \addtocounter{LWR@tabulardepth}{-1}%
```

Restore & to its usual meaning:

```
    \ifnumequal{\value{LWR@tabulardepth}}{0}{%
        \protected\gdef&{\LWR@origampmacro}%
        \StopDefiningTabulars%
    }{}%
```

Error if used `\multirow` or `\multicolumnrow` without using `\mrowcell` or `\mcolrowcell`.

```
    \ifbool{LWR@usedmultirow}{%
        \ifbool{LWR@foundmrowcell}%
            {}%
            {%
                \PackageError{lwarp}%
                {%
             When using \protect\multirow, \protect\multicolumnrow,\MessageBreak
                    or the bigdelim package,\MessageBreak
             place \protect\mrowcell\space or \protect\mcolrowcell\MessageBreak
                    in empty cells which are to be skipped.\MessageBreak
                    See the Lwarp package documentation:\MessageBreak
                    "Special cases and limitations" -> "Tabular"
                }%
                {%
                    See the Lwarp package documentation:\MessageBreak
                    "Special cases and limitations" -> "Tabular".
                }%
            }%
    }{}%

    \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}
```

73 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 lateximage 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}
```

73.1 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}%
```

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 73.3), and another .aux entry (section 73.2).

\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 73.4.

\ref and \nameref: Adds HTML tags. See section 73.4.

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 59.4 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 87.

\label inside math: See section 79.7.1.

Footnotes: See \noteentry in section 59.4.

\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%
    }%
}
```

73.2 New lwarp labels.

File *_html.aux A new entry in *_html.aux is used to remember section name, file, and lateximage depth and number for each label:

```
\newlabel{<labelname>@lwarp}{{<section name>}{<filename>}
                                    {<limagedepth>}{<limagenumber>}}
```

See:
http://tex.stackexchange.com/questions/57194/extract-section-number-from-equation-reference

\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%
}
```

73.3 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}%
}
```

73.4 References

\LWR@addlinktitle

```
\newcommand*{\LWR@addlinktitle}{%
    \ifdefvoid{\LWR@ThisAltText}{}{ % space
        title="\LWR@ThisAltText" % space
        \gdef\LWR@ThisAltText{}%
    }%
}
```

\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:

```
    "%
```

Maybe add a title:

```
    \LWR@addlinktitle%
    }%
    \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
```

73.5 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@linkcatcodes Sets catcodes before processing macros which have hyperlinks as arguments.

```
\newcommand*{\LWR@linkcatcodes}{%
    \catcode`\#=12%
    \catcode`\%=12%
    \catcode`\&=12%
    \catcode`\~=12%
    \catcode`\_=12%
```

For babel-french:

```
    \LWR@FBcancel%
}
```

\LWR@linkmediacatcodes Sets catcodes before processing macros which have hyperlinks as arguments. Modified for multimedia links.

```
\newcommand*{\LWR@linkmediacatcodes}{%
    \catcode`\#=12%
    \catcode`\%=12%
%   \catcode`\&=12% left alone for splitting flash variables
    \catcode`\~=12%
    \catcode`\_=12%
```

For babel-french:

```
    \LWR@FBcancel%
}
```

\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
        \LWR@addlinktitle % 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@addlinktitle % space
    }\LWR@orignewline%
    #2%
    \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%
    \LWR@linkcatcodes%
```

```
    \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%
    \LWR@linkcatcodes%
    \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@urlahere` 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%
    \LWR@linkcatcodes%
    \LWR@urlb%
}
```

\LWR@subinlineimage {⟨<alt> *tag*⟩} {⟨*class*⟩} {⟨*filename*⟩} {⟨*extension*⟩} {⟨*CSS style*⟩}

Factored from lateximage.

```
\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.

74 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".

74.1 Float environment

for HTML output: 9301 \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}{
```

```
9330
9331 \appto\figure@atbegin{%
9332     \LWR@futurenonspacelet\LWR@mynexttoken\LWR@floatalignment%
9333 }
9334
9335 }{}% tocbasic
9336
9337 }% AtBeginDocument
```

\@float
\@dlbfloat

Support packages which create floats directly.

```
9338 \let\@float\LWR@floatbegin
9339 \let\@dblfloat\LWR@floatbegin
```

\LWR@floatend

Ends a \newfloat environment.

```
9340 \newcommand*{\LWR@floatend}{%
```

If saw a \centering, finish the center environment:

```
9341     \LWR@endfloatalignment%
```

Mark the float end for a word processor conversion:

```
9342     \ifboolexpr{bool{FormatWP} and bool{WPMarkFloats}}{%
9343
9344     === end ===
9345
9346     }{}%
9347     \LWR@stoppars%
```

Close an HTML figure tag:

```
9348     \ifbool{FormatWP}{\endLWR@BlockClassWP}{}%
9349     \LWR@htmlelementend{figure}%
9350     \endgroup%
9351     \boolfalse{LWR@freezethisautoid}%
9352     \LWR@startpars%
9353     \ifbool{FormatWP}{\newline}{}%
9354 }
```

\end@float
\end@dlbfloat

Support packages which create floats directly.

```
9355 \let\end@float\LWR@floatend
9356 \let\end@dblfloat\LWR@floatend
```

74.2 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.

```
9357 \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@forcenewautoidanchor` Adds a new <autoid> anchor.

```
\newcommand*{\LWR@forcenewautoidanchor}{%
    \addtocounter{LWR@thisautoid}{1}%
    \LWR@htmltag{a id="\LWR@print@mbox{autoid-\arabic{LWR@thisautoid}}"}%
        \LWR@htmltag{/a}%
}
```

`\LWR@newautoidanchor` Sometimes adds a new <autoid> anchor.

```
\newcommand*{\LWR@newautoidanchor}{%
    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
        {}%
        {\ifbool{LWR@freezethisautoid}{}{\LWR@forcenewautoidanchor}}%
}
```

`\@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}{}%
}
```

74.3 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}%
}%
}
```

74.4 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 entry. The autopage and the float's autoid are remembered for \l@figure to use when creating the HTML links.

File *_html.lot

```
\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 `<div>` 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%
```

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@forcenewautoidanchor%
    \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%
}

}% AtBeginDocument

\end{warpHTML}
```

75 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}
```

75.1 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{section}}%
        {\LWR@closeprevious{chapter}}%
```

Prints any pending footnotes so that they appear above the potentially large TOC:

```
9504     \LWR@printpendingfootnotes%
```

Place the list into its own chapter (if defined) or section:

```
9505     \@ifundefined{chapter}{\section*{#2}}{\chapter*{#2}}%
```

Create a new HTML nav containing the TOC/LOF/LOT:

```
9506     \LWR@htmlelementclass{nav}{#1}%
```

Create the actual list:

```
9507     \LWR@myshorttoc{#1}%
```

Close the nav:

```
9508     \LWR@htmlelementclassend{nav}{#1}%
9509 }
```

`\@starttoc` {⟨*ext*⟩}

Patch `\@starttoc` to encapsulate the TOC inside HTML tags:

```
9510 \let\LWR@orig@starttoc\@starttoc
9511
9512 \renewcommand{\@starttoc}[1]{
9513     \LWR@htmlelementclass{nav}{#1}%
9514     \LWR@orig@starttoc{#1}%
9515     \LWR@htmlelementclassend{nav}{#1}%
9516 }
```

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.)

```
9517 \newbool{LWR@copiedsidetoc}
9518 \boolfalse{LWR@copiedsidetoc}
```

`\tableofcontents` Patch `\tableofcontents`, etc. to print footnotes first. newfloat uses `\listoffigures` for all future float types.

```
9519 \AtBeginDocument{
9520
9521 \let\LWR@origtableofcontents\tableofcontents
9522
9523 \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
    }
}
```

75.2 High-level TOC commands

`\listof` {⟨*type*⟩} {⟨*title*⟩}

Emulate the `\listof` command from the float package (section 237). 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%
}
```

75.3 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`. Warn if parts of the website may be inaccessible.

```
\newcounter{SideTOCDepth}
\setcounter{SideTOCDepth}{1}

\AtBeginDocument{%
    \ifnumcomp{\value{SideTOCDepth}}{<}{\value{FileDepth}}{
        \PackageWarningNoLine{lwarp}
        {%
            SideTOCDepth is less than FileDepth,\MessageBreak
            so some website pages may be inaccessible%
        }
    }{}
}
```

\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}
}
```

75.4 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
```

`\LWR@maybetocdata` Replaced by tocdata. Adds author name.

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

`\hypertoc` {⟨*1: depth*⟩} {⟨*2: type*⟩} {⟨*3: name*⟩} {⟨*4: page*⟩}

Called by `\l@section`, etc. to create a hyperlink to a section.

The autopage label is always created just after the section opens.

#1 is depth

#2 is section, subsection, etc.

#3 the text of the caption

#4 page number

```
\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.

\BaseJobname is added to the label in case xr or xr-hyper are used.

```
            \LWR@subhyperrefclass{%
                \LWR@htmlrefsectionfilename{\BaseJobname-autopage-#4}%
                    \LWR@origpound\LWR@print@mbox{autosec-#4}%
            }{#3}{toc#2}%

            \LWR@maybetocdata%

            \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.

`\BaseJobname` is added to the label in case xr or xr-hyper are used.

```
            \LWR@subhyperrefclass{%
            \LWR@htmlrefsectionfilename{%
                \BaseJobname-autopage-\arabic{LWR@nextautopage}%
            }%
            \LWR@origpound\LWR@print@mbox{autoid-\arabic{LWR@nextautoid}}}%
            {#4}{toc#2}%
```

```
            \LWR@maybetocdata%

            \LWR@stoppars%
        }%
        {}%
}
```

Automatically called by \contentsline:

\l@book {⟨*name*⟩} {⟨*page*⟩}

Uses \DeclareDocumentCommand in case the class does not happen to have a \book.

```
\DeclareDocumentCommand{\l@book}{m m}{\hypertoc{-2}{book}{#1}{#2}}
```

\l@part {⟨*name*⟩} {⟨*page*⟩}

Uses \DeclareDocumentCommand in case the class does not happen to have a \part.

```
\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}
```

76 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}{\LWR@htmlcomment{}}#1%
}
```

\LWR@indexsubitem

```
\newcommand{\LWR@indexsubitem}{

    \InlineClass{indexsubitem}{\LWR@htmlcomment{}}%
}
```

\LWR@indexsubsubitem

```
\newcommand{\LWR@indexsubsubitem}{

    \InlineClass{indexsubsubitem}{\LWR@htmlcomment{}}%
}
```

\@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{\texteb}[1]{\LWR@HTML@texteb{\LWR@doindexentry{##1}}}%
    \renewrobustcmd{\textlg}[1]{\LWR@HTML@textlg{\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{\textulc}[1]{\LWR@HTML@textulc{\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}
```

77 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}
```

78 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%

    \let\&\LWR@origampersand%

    \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%

    \LetLtxMacro\@ensuredmath\LWR@origensuredmath%
%
    \LWR@restoreorigaccents%
    \LWR@restoreoriglists%
%
    \LWR@FBcancel%
}

\end{warpHTML}
```

79 Math

79.1 Limitations

See Math, section 8.7.

79.2 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}
```

\AltTextOpen
Default: (

The opening part of HTML alt tag for an image. The default is a left parenthesis.

```
\newcommand*{\AltTextOpen}{(}
```

\AltTextClose
Default: (

The closing part of HTML alt tag for an image. The default is a right parenthesis.

```
\newcommand*{\AltTextClose}{)}
```

\ImageAltText
Default: image

The HTML alt tag for an image.

```
\newcommand*{\ImageAltText}{image}
```

\MathImageAltText
Default: "math image"

The HTML alt tag for an SVG math image.

```
\newcommand*{\MathImageAltText}{math image}
```

\LWR@ThisAltText

The HTML alt tag for the next image. Cleared after use, and also after each lateximage, \LWR@subsingledollar, and each use of MathJax.

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

\ThisAltText {⟨*text*⟩}

Assigns the HTML alt tag for the next image generated by lwarp, such as a lateximage, picture, or SVG math.

```
\newcommand*{\ThisAltText}[1]{%
    \renewcommand{\LWR@ThisAltText}{#1}%
}
```

\PackageDiagramAltText
Default: "diagram"

Appended to the lateximage HTML alt tag for the images generated by many packages.

```
\newcommand*{\PackageDiagramAltText}{diagram}
```

```
\end{warpall}
```

79.3 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}{}
```

\LWR@setcurrentfont

Sets the actual LaTeX font to that which was selected for HTML output. Ex: In HTML mode, \bfseries sets \LWR@f@series to "bf". This sets the PDF output here for use inside a lateximage.

```
\newcommand*{\LWR@setcurrentfont}{%
    \LWR@traceinfo{Using font family \LWR@f@family}%
    \@nameuse{LWR@print@\LWR@f@family family}%
    \LWR@traceinfo{Using font series \LWR@f@series}%
    \@nameuse{LWR@print@\LWR@f@series series}%
    \LWR@traceinfo{Using font shape \LWR@f@shape}%
    \@nameuse{LWR@print@\LWR@f@shape shape}%
```

```
    \LWR@traceinfo{Using font caps shape \LWR@f@shapecaps}%
    \@nameuse{LWR@print@\LWR@f@shapecaps shape}%
}
```

\$ 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}%
    {\HTMLunicode{00024}}%
}
```

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@originincludegraphics{\LWR@baselinename}%
```

```
        }
    }{
        \@ifpackageloaded{graphics}{
            \newcommand*{\LWR@addbaselinemarker}{%
                \LWR@origincludegraphics{\LWR@baselinename}%
            }
        }{
            \PackageWarningNoLine{lwarp}{Load graphicx or graphics
                for improved\MessageBreak
                SVG math baselines%
            }
            \newcommand*{\LWR@addbaselinemarker}{}
        }
    }
}{% lwarp_baseline_marker.png or .eps is not present
    \PackageWarningNoLine{lwarp}{%
        File \LWR@baselinename\space is not installed\MessageBreak
        alongside the lwarp-*.sty files,\MessageBreak
        so SVG math baselines may not be accurate}
    \newcommand*{\LWR@addbaselinemarker}{}
}

}% AtBeginDocument
```

Bool `LWR@unknownmathsize` If Ti*k*z or other objects are used inside math mode, the resulting image may exceed the TEX box, resulting in an incorrect measurement of the size of the resulting image. If this is so, the HTML styles for image size and depth will be neutralized.

```
\newbool{LWR@unknownmathsize}
```

`\LWR@singledollarmeasure` {⟨*math expression*⟩}

Measures the size of the image of the math expression.

(In some circumstances SVG math is used even if MATHJAX is preferred.)

SVG math: `\LWR@origensuredmath` is part of argument #4.

SVG math `\ensuremath`: `\LWR@origensuredmath` is part of argument #4.

SVG dynamic math: `\LWR@origensuredmath` is part of argument #4.

Mathjax: Argument #4 is the contents of the math expression without `\LWR@origensuredmath`. This case is handled above.

Mathjax `\ensuremath`: `\LWR@origensuredmath` is part of argument #4.

Mathjax dynamic math: Argument #4 is the contents of the math expression without `\LWR@origensuredmath`, so `\LWR@origensuredmath` is added below.

`\ifmmode`: Included "just in case".

Factored from `\LWR@subsingledollarsvg`.

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

Temporarily disable formatting while measuring the image parameters:

```
    \LWR@restoreorigformatting%
    \RenewDocumentEnvironment{lateximage}{s o s o o}{}{}% inside group
    \LWR@print@normalsize%
```

Temporarily set font for the HTML PDF output:

```
    \LWR@setcurrentfont%
```

`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 the math in a box. While doing so, some macros or environments may set `LWR@unknownmathsize`, in which case this will be used to cancel the HTML styles being generated here.

```
    \boolfalse{LWR@unknownmathsize}%
    \ifmmode%
        \global\sbox{\LWR@singledollarbox}{#1}%
    \else%
        \ifbool{LWR@dynamicmath}{%
            \ifbool{mathjax}{%
                \global\sbox{\LWR@singledollarbox}%
                    {\LWR@origensuredmath{#1}}%
            }{%
                \global\sbox{\LWR@singledollarbox}{#1}%
            }%
        }{%
            \global\sbox{\LWR@singledollarbox}{#1}%
        }%
    \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%
}
```

`\LWR@subsingledollarsvg` * {⟨*2: alt text*⟩} {⟨*3: add'l hashing*⟩} {⟨*4: math expression*⟩}

For inline math. Uses SVG math. The image is measured and ajusted to the baseline of the HTML output, and placed inside a `lateximage`.

(In some circumstances SVG math is used even if MATHJAX is preferred.)

Factored from `\LWR@subsingledollar`.

```
\newcommand*{\LWR@subsingledollarsvg}[4]{%
```

Measure the depth, width, and height of the math image:

```
    \LWR@singledollarmeasure{#4}%
```

Set a style for the the height or width. The `em` unit is used so that the math scales according to the user's selected font size.

Start with the greater of the width or the height, biased towards the width:

```
    \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{}%
    }%
```

If using certain Ti*k*z actions inside math, the resulting image may exceed the TEX boundaries, so the HTML size styles may be incorrect, and must be neutralized.

```
    \ifbool{LWR@unknownmathsize}{%
        \def\LWR@singledollarstyle{}%
        \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
            [\MathImageAltText]% alt tag
            []% no add'l hashing
            [\LWR@singledollarstyle \LWR@singledollardepthstyle]% CSS
    }{% not dynamic math
        \LWR@traceinfo{subsingledollar: static}%
        \IfValueTF{#1}{% #1 True
            \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
                *% do not add open/closing braces
                [% addl' hashing
                    #3%
                    FM\LWR@f@family%
                    SR\LWR@f@series%
                    SH\LWR@f@shape%
                    SHC\LWR@f@shapecaps%
                    CL\LWR@tempcolor%
                    FB\LWR@tempone% xfakebold
                ]%
                [\LWR@singledollarstyle \LWR@singledollardepthstyle]% CSS
        }{% #1 False
            \begin{lateximage}% no hashing
                [#2]% alt
                []% no add'l hashing
                [\LWR@singledollarstyle \LWR@singledollardepthstyle]% CSS
        }%
    }% not dynamic math
```

Place small and almost transparent markers on the baseline at the left and right edges of the image. These markers are seen by *pdfcrop*, and force vertically-centered objects such as a dash to be raised off the baseline in the cropped image, and also force the total width and left/right margins to be correct. (Except that in some fonts a character may exceed the bounding box, and thus may appear wider than expected when converted to an image.)

```
    \LWR@addbaselinemarker%
```

Support for xfakebold:

```
    \LWR@applyxfakebold%
```

Typeset the contents:

```
    \usebox{\LWR@singledollarbox}%
```

The closing baseline marker:

```
10052     \LWR@addbaselinemarker%

10053     \end{lateximage}%
10054 %
10055 }
```

`\LWR@subsingledollar` * {⟨*2: alt text*⟩} {⟨*3: add'l hashing*⟩} {⟨*4: math expression*⟩}

For inline math. Uses MathJax, or for SVG math the image is measured and ajusted to the baseline of the HTML output, and placed inside a `lateximage`.

SVG math: `\LWR@origensuredmath` is part of argument #4.

SVG math `\ensuremath`: `\LWR@origensuredmath` is part of argument #4.

SVG dynamic math: `\LWR@origensuredmath` is part of argument #4.

Mathjax: Argument #4 is the contents of the math expression without `\LWR@origensuredmath`. This case is handled above.

Mathjax `\ensuremath`: `\LWR@origensuredmath` is part of argument #4.

Mathjax dynamic math: Argument #4 is the contents of the math expression without `\LWR@origensuredmath`, so `\LWR@origensuredmath` is added below.

image filename hashing

If starred, a hashed filename is used.If so, the hash is based on the `alt` tag and also the additional hashing argument.

This may be used to provide an expression with a simple `alt` tag but also enough additional information to provide a unique hash.

An example is when the expression is a complicated TEX expression, which would not copy/paste well. A simplified tag may be used, while the complicated expression is 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.

```
10056 \newlength{\LWR@singledollarwidth}
10057 \newlength{\LWR@singledollarheight}
10058 \newlength{\LWR@singledollardepth}
10059
10060 \newsavebox{\LWR@singledollarbox}
10061
10062 \NewDocumentCommand{\LWR@subsingledollar}{s m m m}{%
10063     \LWR@traceinfo{LWR@subsingledollar}%

10064     \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
10065     {%
```

```
        \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 `\MathImageAltText`. If this is the case, force the use of a `lateximage` even if MathJax. Likewise for siunitx if `parse-numbers=false`.

If MathJax, or if formatting math for a word processor, and not `\ensuredmath`, and not a dynamic math expression, print the math expression:

```
        \ifboolexpr{%
            (
                bool{mathjax} or
                ( bool{FormatWP} and bool{WPMarkMath} )
            ) and
            ( not test {
                    \ifstrequal {#2}% from \ensuredmath
                        {\AltTextOpen\MathImageAltText\AltTextClose}
                }
            ) and
            ( 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%
            }%
            \LWR@subsingledollarsvg{#1}{#2}{#3}{#4}%
        }% not mathjax
        \endgroup%
    }% not in a lateximage
```

Clear the single-use `alt` text:

```
    \gdef\LWR@ThisAltText{}%
    \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{]}%
    ]%
    *% do not add open/closing braces
```

Support for xfakebold:

```
    \LWR@applyxfakebold%

    \LWR@origdollar\LWR@origdollar#1\LWR@origdollar\LWR@origdollar%
    \end{lateximage}%
    \end{BlockClass}%
}% not mathjax
```

Clear the single-use `alt` text:

```
\gdef\LWR@ThisAltText{}%
}%
```

`\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
```

Clear the single-use `alt` text:

```
\gdef\LWR@ThisAltText{}%
}
```

`\(` `\[` 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*{\AltTextOpen\MathImageAltText\AltTextClose}{%
            \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*%
                        {\AltTextOpen\MathImageAltText\AltTextClose}%
                     {\protect\LWR@HTMLsanitize{\detokenize\expandafter{#1}}}%
                        {\LWR@origensuredmath{#1}}%
                }%
        \fi%
    }%
```

Clear the single-use `alt` text:

```
    \gdef\LWR@ThisAltText{}%
}
```

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}%
    [\MathImageAltText]%
    \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}[\MathImageAltText]%
    \LWR@origequation%
}
{%
    \LWR@origendequation%
    \end{lateximage}%
    \end{BlockClass}%
}
```

79.4 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 79.7.1.

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
}
```

79.5 Equation environment

Remember existing equation environment, after redefined by amsmath, if loaded.

```
\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 lateximage, along with an <alt> tag containing a detokenized copy of the LaTeX source for the math.

For MathJax output, the contents are typeset in an original equation environment placed inside a HTML comment, with special processing for \labels. The contents are also printed to the HTML output for processing by the MathJax script.

```
\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
```

Clear the single-use `alt` text:

```
    \gdef\LWR@ThisAltText{}%
}
```

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}%
    }{}%

```

Clear the single-use `alt` text:

```
    \gdef\LWR@ThisAltText{}%
}
```

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

}% AtBeginDocument
```

Env `equation*`

```
\AtBeginDocument{

\NewEnviron{equation*}%
    {\LWR@doequation{\BODY}{equation*}}%
    [\LWR@doendequation{equation*}]

\csletcs{LWR@equationnormalstar}{equation*}
\csletcs{LWR@endequationnormalstar}{endequation*}

}% AtBeginDocument
```

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*}
}
```

79.6 `\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}
```

79.7 AMS Math environments

79.7.1 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
}
```

79.7.2 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}%
```

Clear the single-use alt text:

```
    \gdef\LWR@ThisAltText{}%
}
```

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}
```

80 Lateximages

80.1 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 78.)

A `lateximage` is typeset on its own PDF page inside an HTML comment which starts on the preceeding page and ends on following page, and instructions are written to `lateximage.txt` for *lwarpmk* to extract the `lateximage` from the page of the PDF file then generate an accompanying `.svg` file image file. Meanwhile, instructions to show this image are placed into the HTML file after the comment.

An HTML `<span>` is created to hold both the HTML comment, which will have the *pdftotext* conversion, and also the link to the final `.svg` image.

A LaTeX label is used to remember which PDF page has the image. A label is used because footnotes, endnotes, and pagenotes may cause the image to appear at a later time. The label is declared along with the image, and so it correctly remembers where the image finally ended up.

HTML `alt` tag The HTML `alt` tag is set to the LaTeX source for SVG math, some chemistry expressions, and perhaps some other expressions which make sense for text copy/paste. In some other cases, the `alt` tag is set according to the package name.

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

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

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

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

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

SVG image font size For the `lateximage` environment, the size of the math and text used in the SVG image

may be adjusted by setting \LateximageFontSizeName to a font size name — *without the backslash,* which defaults to:

```
\renewcommand{\LateximageFontSizeName}{normalsize}
```

For inline SVG math, font size is instead controlled by \LateximageFontScale, which defaults to:

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

80.2 Support counters and macros

for HTML output:

```
\begin{warpHTML}
```

Ctr LWR@lateximagenumber — Sequence the images.

```
\newcounter{LWR@lateximagenumber}
\setcounter{LWR@lateximagenumber}{0}
```

Ctr LWR@lateximagedepth — Do not create \lateximage inside of \lateximage.

```
\newcounter{LWR@lateximagedepth}
\setcounter{LWR@lateximagedepth}{0}
```

A few utility macros to write special characters:

```
\edef\LWR@hashmark{\string#} % for use in \write
\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.

```
\newcounter{LWR@LIpage}
```

```
\end{warpHTML}
```

80.3 Font size

for HTML & PRINT:

```
\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.*

```
\newcommand*{\LateximageFontSizeName}{normalsize}
```

\LateximageFontScale — Declares how large to scale inline SVG math images. The .svg file text size should blend well with the surrounding HTML text size. The default is 1, but it may be redefined as needed depending on the HTML font.

```
\newcommand*{\LateximageFontScale}{1}

\end{warpall}
```

80.4 Equation numbers

for HTML output:

```
\begin{warpHTML}
```

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
```

80.5 HTML alt tags

\LWR@amsmathbody {⟨*envname*⟩} For use inside the optional argument to a lateximage to add the contents of a AMS math environment to the <alt> tag.

```
\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
}
```

80.6 lateximage environment

\LWR@lateximage@oneimageb {⟨*1: alt text*⟩} {⟨*2: filename*⟩} {⟨*3: CSS style*⟩} Creates the image for the lateximage.

```
\newcommand{\LWR@lateximage@oneimageb}[3]{%
    \LWR@subinlineimage{#1}{lateximage}%
    {%
        \LWR@print@mbox{%
            \LWR@ImagesDirectory\OSPathSymbol%
            #2%
        }%
    }{svg}{#3}%
}
```

\LWR@lateximage@oneimage {⟨*1: alt text*⟩} {⟨*2: filename*⟩} {⟨*3: css style*⟩} {⟨*4: delimit?*⟩}

Creates an image for the lateximage, whose alt text depends on the circumstances.

```
\newcommand{\LWR@lateximage@oneimage}[4]{%
    \ifdefvoid{\LWR@ThisAltText}{%
        \IfBooleanTF{#4}{%
            \LWR@lateximage@oneimageb{#1}{#2}{#3}%
        }{%
            \LWR@lateximage@oneimageb%
                {\AltTextOpen#1\AltTextClose}%
                {#2}{#3}%
        }%
    }{%
        \LWR@lateximage@oneimageb%
            {\AltTextOpen\LWR@ThisAltText\AltTextClose}%
            {#2}{#3}%
    }%
}
```

Env lateximage * [⟨*2:* <alt> *tag*⟩] * [⟨*4: add'l hashing*⟩] [⟨*5: css style*⟩]

Typesets the contents and then renders the result as an SVG file. Star #1 causes the image to be hashed for reuse. Star #3 causes the alt tag to not include \AltTextOpen and \AltTextClose, for use with math expressions.

The optional <alt> tag is included in the HTML code for use with copy/paste.

image filename hashing

If starred, a hashed filename is used.If so, the hash is based on the alt tag and also the additional hashing argument.

This may be used to provide an expression with a simple alt tag but also enough additional information to provide a unique hash.

An example is when the expression is a complicated TEX expression, which would not copy/paste well. A simplified tag may be used, while the complicated expression is 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. The last line has “end” in each field, and is used to detect an incomplete compile.

```
\catcode`\$=\active%

\NewDocumentEnvironment{lateximage}{s O{\ImageAltText} s 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
```

Compute and save the hashed file name for later use:

```
        \ifdefvoid{\LWR@ThisAltText}{%
            \IfBooleanTF{#3}{%
                \edef\LWR@hashedname{%
                    \LWR@mdfive{\detokenize\expandafter{#2}-!-#4}%
                }%
            }{%
                \edef\LWR@hashedname{%
              \LWR@mdfive{\detokenize\expandafter{\AltTextOpen#2\AltTextClose}-!-#4}%
                }%
            }%
        }{%
            \edef\LWR@hashedname{%
           \LWR@mdfive{\detokenize\expandafter{\AltTextOpen\LWR@ThisAltText\AltTextClose}-!-#4}%
            }%
        }%
        \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%
```

If the current page is larger, typeset the image in a "standard" width page and font size:

```
    \LWR@traceinfo{lateximage: about to create minipage}%
    \ifdimless{\linewidth}{6in}{%
        \LWR@print@minipage{\linewidth}%
    }{%
        \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%
```

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@lateximage@oneimage{#2}{\LWR@hashedname}{#5}{#3}%
    }% hash
    {% no hash
     \LWR@lateximage@oneimage{#2}{\LWR@ImagesName\theLWR@externalfilecnt}{#5}{#3}%
    }% 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}%
```

Clear the single-use `alt` text:

```
    \gdef\LWR@ThisAltText{}%
}% 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*⟩] * [⟨*add'l hashing*⟩] [⟨*css style*⟩]

`varwidth` is used to create a box of the natural width of its contents.

```
\NewDocumentEnvironment{lateximage}{s o s o o}
    {\begin{varwidth}[b]{\linewidth}}
    {\end{varwidth}}
```

```
\end{warpprint}
```

81 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}
```

82 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 inputtrc was loaded before lwarp, as is usually done, explicitly load the lwarp patches now:

```
\LWR@PreloadedPackage{inputtrc}
```

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}
```

fontaxes must be preloaded so that lwarp may patch it for HTML.

```
\LWR@PreloadedPackage{fontaxes}
```

nfssext-cfr may be preloaded by cfm-lm or related font packages.

```
\LWR@PreloadedPackage{nfssext-cfr}
```

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}
```

83 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.

⚠ 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`.

⚠ MathJax

For math mode with SVG display, the original siunitx code is used while generating the SVG image. For text mode, lwarp uses an emulation which provides a very effective HTML interpretation of siunitx. For math expressions while using MathJax, a limited emulation is used. Most functions work reasonably well, but many options cannot be emulated. siunitx macros with more than one optional value cannot absorb the second optional value, and complicated parsing such as for `\ang` is not supported. The result usually looks fine, and otherwise is enough to get the meaning across.

lwarp's MathJax emulation for siunitx is meant to be a stop-gap measure until an extension is included in MathJax. As of this writing, the third-party siunitx extension for MathJax is not currently hosted at any public CDN, thus siunitx is not usable with this extension unless a local copy of this extension is created first. See `\MathJaxFilename` to select a custom MathJax script, but lwarp's emulation would have to be diabled as well.

Document modifications required for MathJax:

custom units

- Custom units may be added with `\CustomizeMathJax`. See the lwarp-siunitx code for examples.

⚠ unit spacing

- Units work better using ~ between units instead of using periods.

⚠ `\square`, `\cubic`

- To square or cube compound units, enclose the following compound units in braces:

  ```
  \cubic{\centi\meter}
  ```

  Single units do not require braces.

Also see MathJax option, section 8.7.4.

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}
```

84 Graphics print-mode modifications

84.1 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

⚠ 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 according to `\ImageAltText`.

\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.

84.2 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 367 for the HTML version.

```
\AtBeginDocument{
\@ifpackageloaded{overpic}{
\newcommand*{\overpicfontsize}{12}
\newcommand*{\overpicfontskip}{14}
}{}
}
\end{warpprint}
```

85 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}
```

86 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 175.4.

```
\DeclareDocumentEnvironment{redoxreaction}{m m}
    {\rule{0pt}{#1}}{\rule[-#2]{0pt}{#2}}
}{}% chemmacros
}% AtBeginDocument

\end{warpprint}
```

87 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 550 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 472 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}
```

88 `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}
```

89 Minipages and Boxes

A css flexbox is used for minipages and parboxes, allowing external and internal vertical positioning.

⚠ inline A line of text with an inline `minipage` or `\parbox` will have the `minipage` or `\parbox` placed onto its own line, because a paragraph is a block element and cannot be made `inline-block`.

placement `minipages` and `\parboxes` will be placed side-by-side in HTML unless you place a `\newline` between them.

side-by-side Side-by-side `minipages` may be separated by `\quad`, `\qquad`, `\enskip`, `\hspace`, `\hfill`, or a `\rule`. When inside a `center` environment, the result is similar in print and HTML. Paragraph tags are suppressed between side-by-side `minipages` and these spacing commands, but not at the start or end of the paragraph.

⚠ minipage in a span There is limited support for `minipages` inside an HTML . 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 `minipage`, `\parbox`, and `fminipage`, a virtual 6×9 inch text area is used for `\linewidth`, `\textwidth`, and `\textheight`, both for sizing the minipage, and also for its contents.

if width is `\linewidth` — If a `minipage` or `\parbox` is assigned a width of exactly `\linewidth`, in HTML it is automatically given no HTML width, thus allowed to fill the line as needed, similar to how it appears in print output.

full-width if HTML — A new macro `\minipagefullwidth` requests that, during HTML output, the next single `minipage` or `\parbox` be generated without an HTML `width` attribute, allowing it to be the full width of the display rather than the declared print-output width. This may be useful where the printed version's width makes no sense in HTML.

⚠ `tabular`, `multicols` — Inside a `tabular` or `multicols` environment, where the width depends on the browser window, `\minipagefullwidth` is effectively used by default for every `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.

`\UseMinipageWidths`
`\IgnoreMinipageWidths`

⚠ multicol — Inside a `multicols`, `\linewidth` is divided by the specified number of columns.

⚠ text alignment — Nested `minipages` adopt their parent's text alignment in HTML, whereas in regular LaTeX PDF output they do not. Use a `flushleft` or similar environment in the child `minipage` to force a text alignment.

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

89.1 Computed lengths

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}
```

89.2 Virtual page size

Ctr `LWR@virtualpagedepth` — Used to only reset the line width at the outermost minipage.

```
\newcounter{LWR@virtualpagedepth}
\setcounter{LWR@virtualpagedepth}{0}
```

Env `LWR@setvirtualpage` — `*` `[`⟨*columns*⟩`]`

If not nesting a minipage, adjust `\linewidth`, `\textwidth`, and `\textheight` for a virtual 6×9 page, and start on a new PDF page to help prevent page overflows.

If starred, force a new page in the PDF before generating more HTML. This may be done to reduce the chance of page overflow when starting a new minipage.

The optional number of columns defaults to 1.

```
\NewDocumentEnvironment{LWR@setvirtualpage}{s O{1}}{%
    \ifnumequal{\value{LWR@virtualpagedepth}}{0}{%
        \IfBooleanT{#1}{\LWR@orignewpage}%
        \setlength{\linewidth}{6in/#2}%
        \setlength{\textwidth}{6in}%
        \setlength{\textheight}{9in}%
    }{}%
    \addtocounter{LWR@virtualpagedepth}{1}%
}
{\addtocounter{LWR@virtualpagedepth}{-1}}
```

89.3 Footnote handling

Also see section 59 for other forms of footnotes. Minipage footnotes are gathered in section 59.5, and then placed into the document in section 89.4.

89.4 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 ignored.

```
\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}%
```

Start an environment, in which width and height is computed based on a virtual page size instead of the extra-large PDF page used during HTML tag generation.

```
\begin{LWR@setvirtualpage}*%
```

Save the requested width now that `\linewidth`, etc. are adjusted to virtual size.

```
\setlength{\LWR@minipagewidth}{#4}%
\ifnumequal{\value{LWR@virtualpagedepth}}{1}{%
    \addtolength{\LWR@minipagewidth}{3em}% room for frames
}{}%
\LWR@traceinfo{computed width is \LWR@printlength{\LWR@minipagewidth}}%
```

Compute height:

```
\setlength{\LWR@minipageheight}{\textheight}% default unless specified
\ifblank{#2}{}{\setlength{\LWR@minipageheight}{#2}}%
```

LaTeX wants to start a paragraph for the virtual minipage, then start a paragraph again for the contents of the minipage, so cancel the paragraph tag handling until the minipage has begun.

```
\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}%
        {}%
        {%
            \ifdimequal{#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. \LWR@minipagewidth was the original then padded by 3em, which is restored here. This is done instead of settings back to #4, in case #4 was \linewidth, which was changed to 6in above.

```
\ifnumequal{\value{LWR@virtualpagedepth}}{1}{%
    \addtolength{\LWR@minipagewidth}{-3em}% undo frame padding
}{}%
```

```
\setlength{\linewidth}{\LWR@minipagewidth}%
```

\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%

\ifbool{FormatWP}{%

\LWR@htmlelementend{div}%

}{}%
\LWR@htmldivclassend{minipage}%

\end{LWR@setvirtualpage}%
\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}
```

89.5 \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}
```

\LWR@@makebox@paren {⟨*width*⟩} , {⟨*height*⟩}

Adds to the style in \LWR@temptwo.

```
\NewDocumentCommand{\LWR@@makebox@paren}{m m}{%
\IfValueTF{#2}{%
    \setlength{\LWR@tempwidth}{#1\unitlength}%
    \setlength{\LWR@tempheight}{#2\unitlength}%
    \appto{\LWR@temptwo}{%
        \LWR@print@mbox{width:\LWR@printlength{\LWR@tempwidth}} ; % space
        \LWR@print@mbox{height:\LWR@printlength{\LWR@tempheight}} ; % space
    }%
}{%
    \PackageError{lwarp}%
        {(width,height) is missing a comma ',' character}%
        {\protect\makebox\space and \protect\framebox\space accept
            a size in the format (width,height).}%
}%
}
```

\LWR@@makebox@align {⟨*alignment character*⟩}

Adds to the style in \LWR@temptwo.

```
\newcommand*{\LWR@@makebox@align}[1]{%
    \def\LWR@align{center}%
    \ifstrequal{#1}{l}{\def\LWR@align{left}}{}%
    \ifstrequal{#1}{r}{\def\LWR@align{right}}{}%
    \ifstrequal{#1}{s}{\def\LWR@align{justify}}{}%
    \appto{\LWR@temptwo}{%
        \LWR@print@mbox{text-align:\LWR@align} ; %
    }%
}
```

\makebox (⟨*width,height*⟩) [⟨*width*⟩] [⟨*pos*⟩] {⟨*text*⟩}

```
\NewDocumentCommand{\LWR@HTML@makebox}{>{\SplitArgument{1}{,}}d() o o +m}{%
```

Build the style depending on arguments:

```
    \begin{LWR@setvirtualpage}%
        \def\LWR@temptwo{}%
        \IfValueTF{#1}%
        {% (width,height) ..
            \LWR@@makebox@paren #1%
            \IfValueT{#2}%
            {% (width,height) [posn]
                \LWR@@makebox@align{#2}%
            }%
        }%
        {% [width]
            \IfValueT{#2}% [width]
            {%
                \setlength{\LWR@tempwidth}{#2}%
                \ifdimgreater{\LWR@tempwidth}{0pt}{%
                    \appto{\LWR@temptwo}{%
                        width:\LWR@printlength{\LWR@tempwidth} ; % space
                    }%
                }{}%
            }%
        }%
        \IfValueT{#3}%
        {% [width] [posn]
            \LWR@@makebox@align{#3}%
        }%
        \InlineClass[%
            \LWR@print@mbox{display:inline-block} ; %
            \LWR@temptwo%
        ]%
        {makebox}%
        {#4}%
    \end{LWR@setvirtualpage}%
}
\LWR@formatted{makebox}
```

`\framebox` `(`⟨*width,height*⟩`)` `[`⟨*width*⟩`]` `[`⟨*pos*⟩`]` `{`⟨*text*⟩`}`

```
\NewDocumentCommand{\LWR@HTML@framebox}{d() o o +m}{%
    \fbox{\makebox(#1)[#2][#3]{#4}}%
}

\LWR@formatted{framebox}
```

`\LWR@forceminwidth` `{`⟨*legth*⟩`}`

Sets `\LWR@atleastonept` to be at least 1pt.

```
\newlength{\LWR@atleastonept}

\newcommand*{\LWR@forceminwidth}[1]{%
\setlength{\LWR@atleastonept}{#1}%
\ifthenelse{%
    \lengthtest{\LWR@atleastonept>0pt}\AND%
    \lengthtest{\LWR@atleastonept<1pt}%
}%
    {\setlength{\LWR@atleastonept}{1pt}}%
    {}%
}
```

`\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.

```
\newcommand*{\LWR@fboxstyle}{%
\LWR@findcurrenttextcolor%
border:\LWR@printlength{\LWR@atleastonept} solid \LWR@origpound\LWR@tempcolor ; %
padding:\LWR@printlength{\fboxsep} ; %
color:\LWR@origpound\LWR@tempcolor%
}
```

`\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:

```
\newcommand{\LWR@HTML@fbox}[1]{%
\LWR@traceinfo{HTML fbox}%
\LWR@forceminwidth{\fboxrule}%
\InlineClass[%
    \LWR@print@mbox{display:inline-block} ; %
    \LWR@fboxstyle%
]{fbox}{#1}%
}
```

xcolor `\let`s things to `\fbox` when it is loaded, and this must remain even for HTML output while in a `latexmage`, so `\fbox` is not modified until `\AtBeginDocument`:

```
\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}%
\LWR@stoppars%
\begin{BlockClass}[%
\LWR@fboxstyle%
]{fboxBlock}
#1
\end{BlockClass}
\LWR@startpars%
}

\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}
```

HTML version:

for HTML output:

```
\begin{warpHTML}

\NewDocumentEnvironment{LWR@HTML@fminipage}{O{t} o O{t} m}
{%
\LWR@traceinfo{fminipage #1 #2 #3 #4}%
```

Locally change to the virtual page size before processing the requested sizes:

```
\begin{LWR@setvirtualpage}*%
\setlength{\LWR@tempwidth}{#4}%
\IfValueT{#2}{\setlength{\LWR@tempheight}{#2}}%
```

Use a rule of at least one pixel in width:

```
\LWR@forceminwidth{\fboxrule}%


\LWR@stoppars%
```

```
\begin{BlockClass}[%
\LWR@fboxstyle ; %
\IfValueT{#2}{height:\LWR@printlength{\LWR@tempheight} ; }%
\ifbool{LWR@minipagefullwidth}%
{\global\boolfalse{LWR@minipagefullwidth}}%
{%
    \ifbool{LWR@forceminipagefullwidth}%
        {}%
        {%
            \ifdimequal{\LWR@tempwidth}{\linewidth}%
                {}%
                {width:\LWR@printlength{\LWR@tempwidth} ; }%
        }%
}%
]{fminipage}%
}
{%
\end{BlockClass}%
\end{LWR@setvirtualpage}%
```

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}
```

`\raisebox` `{`⟨*raiselen*⟩`}` `[`⟨*height*⟩`]` `[`⟨*depth*⟩`]` `{`⟨*text*⟩`}`

```
\NewDocumentCommand{\LWR@HTML@raisebox}{m o o m}{%
#4%
}

\LWR@formatted{raisebox}

\end{warpHTML}
```

90 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}

\newcommand{\LWR@null@emph}[1]{#1}
```

\textmd {⟨*text*⟩}

```
\DeclareRobustCommand{\LWR@HTML@textmd}[1]{%
    {%
        \mdseries%
        \InlineClass(font-weight:normal){textmd}{#1}%
    }%
}

\LWR@formatted{textmd}

\newcommand{\LWR@null@textmd}[1]{#1}
```

\textbf {⟨*text*⟩}

```
\DeclareRobustCommand{\LWR@HTML@textbf}[1]{%
    {%
        \bfseries%
        \LWR@htmlspan{b}{#1}%
    }%
}

\LWR@formatted{textbf}

\newcommand{\LWR@null@textbf}[1]{#1}
```

\texteb {⟨*text*⟩} From nfssext-cfr.

```
\DeclareRobustCommand{\LWR@HTML@texteb}[1]{%
    {%
        \ebweight%
        \InlineClass{texteb}{#1}%
```

```
    }%
}

\LWR@formatted{texteb}

\newcommand{\LWR@null@texteb}[1]{#1}
```

`\textlg {⟨text⟩}` From nfssext-cfr.

```
\DeclareRobustCommand{\LWR@HTML@textlg}[1]{%
    {%
        \lgweight%
        \InlineClass{textlg}{#1}%
    }%
}

\LWR@formatted{textlg}

\newcommand{\LWR@null@textlg}[1]{#1}
```

`\textrm {⟨text⟩}`

```
\DeclareRobustCommand{\LWR@HTML@textrm}[1]{%
    {%
        \rmfamily%
        \InlineClass(font-family:serif){textrm}{#1}%
    }%
}

\LWR@formatted{textrm}

\newcommand{\LWR@null@textrm}[1]{#1}
```

`\textsf {⟨text⟩}`

```
\DeclareRobustCommand{\LWR@HTML@textsf}[1]{%
    {%
        \sffamily%
        \InlineClass(font-family:sans){textsf}{#1}%
    }%
}

\LWR@formatted{textsf}

\newcommand{\LWR@null@textsf}[1]{#1}
```

`\texttt {⟨text⟩}`

```
\DeclareRobustCommand{\LWR@HTML@texttt}[1]{%
    {%
        \ttfamily%
        \LWR@htmlspan{kbd}{#1}%
```

```
    }%
}

\LWR@formatted{texttt}

\newcommand{\LWR@null@texttt}[1]{#1}
```

\textup {⟨*text*⟩}

```
\DeclareRobustCommand{\LWR@HTML@textup}[1]{%
    {%
        \upshape%
        \InlineClass(font-style:normal){textup}{#1}%
    }%
}

\LWR@formatted{textup}

\newcommand{\LWR@null@textup}[1]{#1}
```

\textit {⟨*text*⟩}

```
\DeclareRobustCommand{\LWR@HTML@textit}[1]{%
    {%
        \itshape%
        \LWR@htmlspan{i}{#1}%
    }%
}

\LWR@formatted{textit}

\newcommand{\LWR@null@textit}[1]{#1}
```

\textsc {⟨*text*⟩}

```
\DeclareRobustCommand{\LWR@HTML@textsc}[1]{%
    {%
        \scshape%
        \InlineClass{textsc}{#1}%
    }%
}

\LWR@formatted{textsc}

\newcommand{\LWR@null@textsc}[1]{#1}
```

\textulc {⟨*text*⟩} From fontaxes.

```
\DeclareRobustCommand{\LWR@HTML@textulc}[1]{%
    {%
        \ulcshape%
        \InlineClass{textulc}{#1}%
```

```
11438     }%
11439 }
11440
11441 \LWR@formatted{textulc}
11442
11443 \newcommand{\LWR@null@textulc}[1]{#1}
```

\textsi {⟨*text*⟩}

```
11444 \@ifundefined{textsi}{
11445     \LetLtxMacro\LWR@print@textsi\LWR@print@textsc
11446 }{}
11447
11448 \DeclareRobustCommand{\LWR@HTML@textsi}[1]{%
11449     {%
11450         \sishape%
11451         \textsc{\textit{#1}}%
11452 %         \InlineClass(
11453 %             font-style: italic;
11454 %             font-variant: small-caps ;
11455 %             font-variant-numeric: oldstyle-nums ;
11456 %         ){textsi}{#1}%
11457     }%
11458 }
11459
11460 \LWR@formatted{textsi}
11461
11462 \newcommand{\LWR@null@textsi}[1]{#1}
```

\textsl {⟨*text*⟩}

```
11463 \DeclareRobustCommand{\LWR@HTML@textsl}[1]{%
11464     {%
11465         \slshape%
11466         \InlineClass(font-style:oblique){textsl}{#1}%
11467     }%
11468 }
11469
11470 \LWR@formatted{textsl}
11471
11472 \newcommand{\LWR@null@textsl}[1]{#1}
```

\textnormal {⟨*text*⟩}

```
11473 \DeclareRobustCommand{\LWR@HTML@textnormal}[1]{\textmd{\textrm{\textup{#1}}}}
11474
11475 \LWR@formatted{textnormal}
11476
11477 \newcommand{\LWR@null@textnormal}[1]{#1}
```

```
11478 \newcommand{\LWR@null@rmfamily}{}
11479 \newcommand{\LWR@null@sffamily}{}
11480 \newcommand{\LWR@null@ttfamily}{}
```

```
\newcommand{\LWR@null@bfseries}{}
\newcommand{\LWR@null@ebweight}{}
\newcommand{\LWR@null@lgweight}{}
\newcommand{\LWR@null@mdseries}{}
\newcommand{\LWR@null@upshape}{}
\newcommand{\LWR@null@slshape}{}
\newcommand{\LWR@null@scshape}{}
\newcommand{\LWR@null@itshape}{}
\newcommand{\LWR@null@normalfont}{}

\newcommand{\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\textulc\LWR@null@textulc%
\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\ulcshape\LWR@null@ulcshape%

\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{\hspace}{s m}{}%

\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. fontaxes support is integrated here.

```
\newcommand*{\LWR@f@family}{rm}
\newcommand*{\LWR@f@series}{md}
\newcommand*{\LWR@f@shape}{up}
\newcommand*{\LWR@f@shapecaps}{ulc}
```

`\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\LWR@origtilde{}%
                    text\LWR@f@shapecaps%
                }%
                {#1}%
            \addtocounter{LWR@textcurrentfontdepth}{-1}%
        }%
}
```

Env LWR@blocktextcurrentfont Prints the contents with the current font choices.

```
\newenvironment*{LWR@blocktextcurrentfont}{%
\LWR@stoppars%
\BlockClass{%
        text\LWR@f@family\LWR@origtilde{}%
        text\LWR@f@series\LWR@origtilde{}%
        text\LWR@f@shape\LWR@origtilde{}%
        text\LWR@f@shapecaps%
    }%
}{\endBlockClass\LWR@startpars}
```

\mdseries

```
\newrobustcmd*{\LWR@HTML@mdseries}{\renewcommand*{\LWR@f@series}{md}}
\LWR@formatted{mdseries}
```

\bfseries

```
\newrobustcmd*{\LWR@HTML@bfseries}{\renewcommand*{\LWR@f@series}{bf}}
\LWR@formatted{bfseries}
```

\ebweight From nfssext-cfr.

```
\newrobustcmd*{\LWR@HTML@ebweight}{\renewcommand*{\LWR@f@series}{eb}}
\LWR@formatted{ebweight}
```

\lgweight From nfssext-cfr.

```
\newrobustcmd*{\LWR@HTML@lgweight}{\renewcommand*{\LWR@f@series}{lg}}
\LWR@formatted{lgweight}
```

\rmfamily

```
\newrobustcmd*{\LWR@HTML@rmfamily}{\renewcommand*{\LWR@f@family}{rm}}
\LWR@formatted{rmfamily}
```

\sffamily

```
\newrobustcmd*{\LWR@HTML@sffamily}{\renewcommand*{\LWR@f@family}{sf}}
\LWR@formatted{sffamily}
```

\ttfamily

```
\newrobustcmd*{\LWR@HTML@ttfamily}{\renewcommand*{\LWR@f@family}{tt}}
\LWR@formatted{ttfamily}
```

\upshape

```
\newrobustcmd*{\LWR@HTML@upshape}{\renewcommand*{\LWR@f@shape}{up}}
\LWR@formatted{upshape}
```

\itshape

```
\newrobustcmd*{\LWR@HTML@itshape}{\renewcommand*{\LWR@f@shape}{it}}
\LWR@formatted{itshape}
```

\scshape

```
\newrobustcmd*{\LWR@HTML@scshape}{\renewcommand*{\LWR@f@shapecaps}{sc}}
\LWR@formatted{scshape}
```

\ulcshape From fontaxes.

```
\@ifundefined{ulcshape}{
    \LetLtxMacro\ulcshape\upshape
}{}
\newrobustcmd*{\LWR@HTML@ulcshape}{\renewcommand*{\LWR@f@shapecaps}{ulc}}
\LWR@formatted{ulcshape}
```

\sishape

```
\@ifundefined{sishape}{
    \LetLtxMacro\sishape\scshape
}{}
\newrobustcmd*{\LWR@HTML@sishape}{%
    \renewcommand*{\LWR@f@shape}{it}
    \renewcommand*{\LWR@f@shapecaps}{sc}%
}
\LWR@formatted{sishape}
```

\slshape

```
\newrobustcmd*{\LWR@HTML@slshape}{\renewcommand*{\LWR@f@shape}{sl}}
\LWR@formatted{slshape}
```

\normalfont

```
\newrobustcmd*{\LWR@HTML@normalfont}{\rmfamily\mdseries\upshape\ulcshape}
\LWR@formatted{normalfont}
```

\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}
```

91 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 remembered for future use:

```
\newcommand*{\LWR@font@size}{normalsize}

\newrobustcmd*{\LWR@HTML@normalsize}{\renewcommand*{\LWR@font@size}{normalsize}}
\LWR@formatted{normalsize}

\newrobustcmd*{\LWR@HTML@small}{\renewcommand*{\LWR@font@size}{small}}
\LWR@formatted{small}

\newrobustcmd*{\LWR@HTML@footnotesize}{\renewcommand*{\LWR@font@size}{footnotesize}}
\LWR@formatted{footnotesize}

\newrobustcmd*{\LWR@HTML@scriptsize}{\renewcommand*{\LWR@font@size}{scriptsize}}
\LWR@formatted{scriptsize}

\newrobustcmd*{\LWR@HTML@tiny}{\renewcommand*{\LWR@font@size}{tiny}}
\LWR@formatted{tiny}

\newrobustcmd*{\LWR@HTML@large}{\renewcommand*{\LWR@font@size}{large}}
\LWR@formatted{large}

\newrobustcmd*{\LWR@HTML@Large}{\renewcommand*{\LWR@font@size}{Large}}
\LWR@formatted{Large}
```

```
11778
11779 \newrobustcmd*{\LWR@HTML@LARGE}{\renewcommand*{\LWR@font@size}{LARGE}}
11780 \LWR@formatted{LARGE}
11781
11782 \newrobustcmd*{\LWR@HTML@huge}{\renewcommand*{\LWR@font@size}{huge}}
11783 \LWR@formatted{huge}
11784
11785 \newrobustcmd*{\LWR@HTML@Huge}{\renewcommand*{\LWR@font@size}{Huge}}
11786 \LWR@formatted{Huge}


11787 \DeclareDocumentCommand{\onecolumn}{}{}
11788
11789 \DeclareDocumentCommand{\twocolumn}{O{}}{
11790
11791 #1
11792
11793 }
```

\hfill

```
11794 \newcommand*{\LWR@HTML@hfill}{\qquad}
11795 \LWR@formatted{hfill}
```

\hrulefill

```
11796 \newcommand*{\LWR@HTML@hrulefill}{\rule{1in}{1pt}}
11797 \LWR@formatted{hrulefill}
```

\dotfill

```
11798 \newcommand*{\LWR@HTML@dotfill}{\dots}
11799 \LWR@formatted{dotfill}
```

\newpage

```
11800 \renewcommand*{\newpage}{
11801
11802 }
```

\newline Uses the HTML <br /> element.

```
11803 \newrobustcmd*{\LWR@newlinebr}{\unskip\LWR@htmltag{br /}\LWR@orignewline}%
11804 \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 greater than 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 \quads, 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}%
            }%
        }%
```

If NOT formatting for a word processor, include an empty comment to avoid an empty span:

```
        {\LWR@htmlcomment{}}%
```

Close the span:

```
        \LWR@htmltagc{/span}%
    }%
```

If had a minipage this paragraph, try to inline the white space without generating paragraph tags:

```
    \LWR@minipagestartpars%
}{}% width greater than 0
}%
}
```

\LWR@select@html@nohspace * {⟨*length*⟩}
\hspace

Used to disable \hspace while creating description \items.

```
\newcommand{\LWR@select@html@nohspace}{%
    \RenewDocumentCommand{\hspace}{s m}{}%
}
```

\LWR@select@print@hspace

```
\newcommand*{\LWR@select@print@hspace}{%
    \renewrobustcmd\hspace{\@ifstar\@hspacer\@hspace}%
}
```

\hspace * {⟨*length*⟩}

Handles special minipage & horizontal space interactions.

```
\LWR@select@html@hspace
```

`\LWR@vspace` `*` `{`⟨*length*⟩`}` Nullified vspace.

```
\NewDocumentCommand{\LWR@HTML@vspace}{s m}{}

\LWR@formatted{vspace}
```

`\linebreak` `[`⟨*num*⟩`]` Inserts an HTML `br` tag.

```
\renewcommand*{\linebreak}[1][]{\newline}
```

`\nolinebreak` `[`⟨*num*⟩`]`

```
\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 HTML background color is used to draw the filled rule according to the LaTeX foreground color set 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}%
            }%
        }%
```

If NOT formatting for a word processor, add a comment to avoid an empty :

```
        {\LWR@htmlcomment{}}%
```

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}
```

92 \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}
```

93 \LaTeX and other logos

Logos for HTML and print modes:

Some of these logos may be redefined in a later package, so after loading other packages, and at the beginning of the document, their definitions are finally set by \LWR@formatted.

For CSS conversions, see:
http://edward.oconnor.cx/2007/08/tex-poshlet
http://nitens.org/taraborelli/texlogo
and the spacing described in the metafont package documentation.

for HTML & PRINT:
```
\begin{warpall}
\newrobustcmd*{\Xe}
    {X\hspace{-.1667em}\raisebox{-.5ex}{E}}

\AtBeginDocument{
\@ifpackageloaded{graphics}{
    \@ifpackageloaded{metalogo}{}{
        \renewrobustcmd*{\Xe}
            {X\hspace{-.1667em}\raisebox{-.5ex}{\reflectbox{E}}}
    }
}{
    \PackageWarningNoLine{lwarp}{Load graphicx or graphics
        for improved XeTeX logo}
}
}

\providerobustcmd*{\XeTeX}{\mbox{\Xe\hspace{-.125em}\TeX}}
\providerobustcmd*{\XeLaTeX}{\mbox{\Xe\hspace{-.125em}\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{warpall}
```

for HTML output:

```
\begin{warpHTML}
```

The print-mode versions of the following may be changed by metalogo, so their print formatting is recorded `\AtBeginDocument`.

`\TeX` TEX

`latexlogo` is a css class used to properly typeset the E and A in LATEX and friends.

`latexlogofont` is a css class used to select the font for the rest of the logo in LATEX, LuaTEX, ConTEXt, etc.

```
\newrobustcmd*{\LWR@HTML@TeX}
{%
    \InlineClass{latexlogofont}%
    {%
        \InlineClass{latexlogo}%
        {%
            T%
            \InlineClass{latexlogosub}{e}%
            X%
        }%
    }%
}
\AtBeginDocument{\LWR@formatted{TeX}}% may have been patched by metalogo
```

`\LaTeX` LATEX, LATEX 2_ε
`\LaTeXe`

```
\newrobustcmd*{\LWR@HTML@LaTeX}
{%
    \InlineClass{latexlogofont}%
    {%
        \InlineClass{latexlogo}%
        {%
            L%
            \InlineClass{latexlogosup}{a}%
            T%
            \InlineClass{latexlogosub}{e}%
            X%
        }%
    }%
}

\AtBeginDocument{\LWR@formatted{LaTeX}}% may have been patched by metalogo

```

```
12007
12008 \newrobustcmd*{\LWR@HTML@LaTeXe}
12009 {%
12010     \LaTeX%
12011     \InlineClass{latexlogofont}{%
12012         \InlineClass{latexlogotwoe}{%
12013             2%
12014             \InlineClass{latexlogotwoesub}{\HTMLunicode{03B5}}%
12015         }%
12016     }%
12017 }
12018 \AtBeginDocument{\LWR@formatted{LaTeXe}}% may have been patched by metalogo
```

\LuaTeX \LuaLaTeX

LuaTEX, LuaLATEX

```
12019 \newrobustcmd*{\LWR@HTML@LuaTeX}{\InlineClass{latexlogofont}{Lua}\TeX}
12020 \AtBeginDocument{\LWR@formatted{LuaTeX}}% may have been patched by metalogo
12021
12022 \newrobustcmd*{\LWR@HTML@LuaLaTeX}{\InlineClass{latexlogofont}{Lua}\LaTeX}
12023 \AtBeginDocument{\LWR@formatted{LuaLaTeX}}% may have been patched by metalogo
```

\XeTeX \XeLaTeX

XƎTEX, XƎLATEX

xetexlogo is a css class which aligns the backwards E in XƎTEX and spaces TEX appropriately.

xelatexlogo is a css class which aligns the backwards E in XƎLATEX and spaces LATEX appropriately.

```
12024 \newrobustcmd*{\LWR@HTML@Xe}
12025     {%
12026         X%
12027         \InlineClass{xelatexlogosub}{\HTMLunicode{18e}}%
12028     }
12029 \AtBeginDocument{\LWR@formatted{Xe}}% may have been patched by metalogo
12030
12031 \newrobustcmd*{\LWR@HTML@XeTeX}{\InlineClass{xelatexlogo}{\Xe}\TeX}
12032 \AtBeginDocument{\LWR@formatted{XeTeX}}% may have been patched by metalogo
12033
12034 \newrobustcmd*{\LWR@HTML@XeLaTeX}{\InlineClass{xelatexlogo}{\Xe}\LaTeX}
12035 \AtBeginDocument{\LWR@formatted{XeLaTeX}}% may have been patched by metalogo
```

\ConTeXt

ConTEXt

```
12036 \newrobustcmd*{\LWR@HTML@ConTeXt}{%
12037     \InlineClass{latexlogofont}{Con}\TeX{}%
12038     \InlineClass{latexlogofont}{t}%
12039 }
12040 \LWR@formatted{ConTeXt}
```

\BibTeX \MakeIndex

BIBTEX, *MakeIndex*

```
\newrobustcmd*{\LWR@HTML@BibTeX}
    {\InlineClass{latexlogofont}{B\textsc{ib}}\TeX}
\LWR@formatted{BibTeX}

\newrobustcmd*{\LWR@HTML@MakeIndex}
    {\InlineClass{latexlogofont}{\textit{MakeIndex}}}
\LWR@formatted{MakeIndex}
```

`\AmS` $\mathcal{AMS}$

amslogo is a CSS class used for the $\mathcal{AMS}$ logo.

```
\AtBeginDocument{%
\newrobustcmd*{\LWR@HTML@AmS}
{%
    \InlineClass{amslogo}{%
        \textit{%
            A%
            \InlineClass{latexlogosub}{M}%
            S%
        }%
    }%
}%
\LWR@formatted{AmS}
}
```

`\MiKTeX` MiKTEX

```
\newrobustcmd*{\LWR@HTML@MiKTeX}{\InlineClass{latexlogofont}{MiK}\TeX}
\LWR@formatted{MiKTeX}
```

`\LyX` LyX

lyxlogo is a CSS class used for the LyX logo.

```
\newrobustcmd*{\LWR@HTML@LyX}{\InlineClass{lyxlogo}{LyX}}
\LWR@formatted{LyX}

\end{warpHTML}
```

94 \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}
```

95 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}
```

96 Loading MEMOIR class patches

Load patches to memoir.

for HTML output:
```
\begin{warpHTML}
\@ifclassloaded{memoir}{\RequirePackage{lwarp-patch-memoir}}{}
\end{warpHTML}
```

97 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}
```

98 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}
```

99 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}
```

100 babel and polyglossia warnings

lwarp prints a message instructing the user how to avoid the following error.

(These are not \PackageWarnings because there may not be a problem.)

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

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

for HTML output:

```
\begin{warpHTML}
\AtBeginDocument{

\@ifpackageloaded{polyglossia}{
    \PackageWarningNoLine{lwarp}
    {%
        Polyglossia has been loaded.  Lwarp also uses cleveref.\MessageBreak
        See the cleveref documentation regarding\MessageBreak
        polyglossia support%
    }
    \typeout{If the error}
    \typeout{\space\space``! Undefined control sequence.
        \space ... \space \protect\@begindocumenthook''}
    \typeout{occurs here, use the polyglossia macro:}
    \typeout{\space\space\protect\setmainlanguage\protect{...\protect}}
    \typeout{---}
}{
    \@ifpackageloaded{babel}{
        \PackageWarningNoLine{lwarp}
        {%
            Babel has been loaded.  Lwarp also uses cleveref.\MessageBreak
            See the cleveref documentation regarding\MessageBreak
            babel support%
        }
    }{}
}

}
\end{warpHTML}
```

101 MathJax warnings

\LWR@mathjaxwarn {⟨*packagename*⟩} {⟨*More text.*⟩}

To be done \AtBeginDocument.

```
\newcommand*{\LWR@mathjaxwarn}[2]{%
    \@ifpackageloaded{lwarp-#1}{%
        \ifblank{#2}{%
            \PackageWarningNoLine{lwarp}
                {%
                    Lwarp provides emulation for MathJax when used\MessageBreak
                     with the #1 package%
                }
        }{%
            \PackageWarningNoLine{lwarp}
                {%
                    Lwarp provides emulation for MathJax when used\MessageBreak
                     with the #1 package.\MessageBreak
                     #2%
                }
        }%
    }{}%
}

\AtBeginDocument{
    \ifbool{mathjax}{
        \LWR@mathjaxwarn{arydshln}
            {In a math array, do not use the optional argument\MessageBreak
            for \protect\cdashline.\space\space
            Furthermore, \protect\cline\space is not\MessageBreak
            supported by MathJax}
        \LWR@mathjaxwarn{autonum}
            {MathJax does not support equation+.\MessageBreak
            You may use the warpprint and warpHTML\MessageBreak
            environments to isolate the package load\MessageBreak
            and the equation+ environments}
        \LWR@mathjaxwarn{backnaur}
            {You may enclose its uses inside\MessageBreak
            lateximage environments to force SVG output,\MessageBreak
            but this also changes the print output}
        \LWR@mathjaxwarn{bigdelim}
            {Delimiters appear only of the first line}
        \LWR@mathjaxwarn{booktabs}
            {In a math array, do not use (trim) for \protect\cmidrule}
        \LWR@mathjaxwarn{breqn}
            {Each environment becomes an SVG image}
        \LWR@mathjaxwarn{delarray}
            {Enclose its uses inside lateximage\MessageBreak
            environments to force SVG output}
        \LWR@mathjaxwarn{jkmath}{}
        \LWR@mathjaxwarn{mathspec}{}
        \LWR@mathjaxwarn{mathtools}
            {Avoid starred macros.\MessageBreak
```

```
            See the Lwarp manual for other limitations}
        \LWR@mathjaxwarn{multirow}
            {Multirow works as expected in text mode, but\MessageBreak
            limited emulation is provided for MathJax math.\MessageBreak
            \protect\multirow\space ignores all arguments except\MessageBreak
            the text}
        \LWR@mathjaxwarn{pb-diagram}
            {Enclose its uses inside lateximage environments\MessageBreak
            to force SVG output}
        \LWR@mathjaxwarn{physics}
            {The third-party extension is not yet used.\MessageBreak
            Avoid starred macros and automatic delimiters.\MessageBreak
            Use all mandatory arguments, adding empty as needed.\MessageBreak
            See the Lwarp manual for details}
        \LWR@mathjaxwarn{unicode-math}
            {Not all characters are encoded correctly.\MessageBreak
            Some symbol fonts are not supported by MathJax,\MessageBreak
            and are only approximated}
        \LWR@mathjaxwarn{witharrows}
            {Arrows can only point to the next line.\MessageBreak
            Text is only placed on a single line}
        \LWR@mathjaxwarn{xy}
            {xy works in text, but in math you must\MessageBreak
            enclose its uses inside lateximage\MessageBreak
            environments to force SVG output}
    }{}
}
```

File 2 **lwarp-2in1.sty**

§ 102 Package 2in1

Pkg 2in1 2in1 is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{2in1}
```

File 3 **lwarp-2up.sty**

§ 103 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**

§ 104 Package a4

Pkg a4 a4 is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{a4}[2004/04/15]

\newcommand*{\WideMargins}{}
```

File 5 **lwarp-a4wide.sty**

§ 105 Package a4wide

Pkg a4wide a4wide is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{a4wide}[1994/08/30]
```

File 6 **lwarp-a5comb.sty**

§ 106 Package a5comb

Pkg a5comb a5comb is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{a5comb}
```

File 7 **lwarp-abstract.sty**

§ 107 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**

§ 108 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-afterpage.sty**

§ 109 Package afterpage

(*Emulates or patches code by* David Carlisle.)

Pkg afterpage afterpage is emulated.

for HTML output: Discard all options for lwarp-afterpage:

```
\LWR@ProvidesPackageDrop{afterpage}[2014/10/28]

\newcommand{\afterpage}[1]{#1}
```

File 10 **lwarp-accessibility.sty**

§ 110 Package accessibility

Pkg accessibility accessibility is emulated.

for HTML output: Discard all options for lwarp-accessibility:

```
\LWR@ProvidesPackageDrop{accessibility}[2019/10/14]

\newcommand{\alt}[1]{\ThisAltText{#1}}
\newcommand{\newhref}[3]{\ThisAltText{#2}\href{#1}{#3}}%
\providecommand{\thead}[1]{\textbf{#1}}
```

File 11 **lwarp-accsupp.sty**

§ 111 Package accsupp

Pkg accsupp accsupp is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{accsupp}[2018/03/28]

\newcommand*{\BeginAccSupp}[1]{}
\newcommand*{\EndAccSupp}[1]{}
```

File 12 **lwarp-acro.sty**

§ 112 Package acro

(*Emulates or patches code by* Clemens Niederberger.)

Pkg acro acro is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{acro}[2019/10/12]
```

\DeclareAcronym is used in the preamble, where lwarp has not yet made the dollar active, so temporarily enable lwarp math catcode just for this definition:

```
\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
\@ifpackagelater{acro}{2019/09/23}%
{% v2.10 or later
\cs_gset_protected:Npn \__acro_typeset:nn #1#2
  {
    \mode_if_horizontal:F { \leavevmode }
    \group_begin:
      \use:x
        {
```

```
          \bool_if:cTF {l__acro_custom_#1_format_bool}
            { \exp_not:v {l__acro_custom_#1_format_tl} }
            { \exp_not:v {l__acro_#1_format_tl} }
            {\exp_not:N\LWR@textcurrentfont{#2}}%   lwarp
        }
    \group_end:
  }

\cs_gset_protected:Npn \__acro_ending_format:nn #1#2
  {
    \bool_if:NTF \l__acro_include_endings_format_bool
      {
        \str_case:nn {#1}
          {
            {long}
            {
              \bool_if:NTF \l__acro_custom_long_format_bool
                { \l__acro_custom_long_format_tl }
                {
                  \bool_if:NTF \l__acro_first_instance_bool
                    { \l__acro_first_long_format_tl }
                    { \l__acro_long_format_tl }
                }
            }
            {short}
            {
              \bool_if:NTF \l__acro_custom_short_format_bool
                { \l__acro_custom_short_format_tl }
                { \l__acro_short_format_tl }
            }
            {alt}
            {
              \bool_if:NTF \l__acro_custom_alt_format_bool
                { \l__acro_custom_alt_format_tl }
                { \l__acro_alt_format_tl }
            }
          }
      }
      { \use:n }
      {\exp_not:N\LWR@textcurrentfont{#2}}% lwarp
  }
}% v2.10 or later
{% before v2.10
\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:
  }
}% before v2.10
\ExplSyntaxOff
```

File 13 **lwarp-acronym.sty**

§ 113 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 14 **lwarp-adjmulticol.sty**

§ 114 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 15 **lwarp-addlines.sty**

§ 115 Package addlines

(*Emulates or patches code by* Will Robertson.)

Pkg addlines addlines is ignored.

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 16 **lwarp-afterpage.sty**

§ 116 Package afterpage

(*Emulates or patches code by* David Carlisle.)

Pkg afterpage afterpage is emulated.

for HTML output: Discard all options for lwarp-afterpage:

```
\LWR@ProvidesPackageDrop{afterpage}[2014/10/28]
\newcommand{\afterpage}[1]{#1}
```

File 17 **lwarp-algorithm2e.sty**

§ 117 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 `latexmk`ages 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 18 **lwarp-algorithmicx.sty**

§ 118 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 506.1.

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 19 **lwarp-alltt.sty**

§ 119 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 20 **lwarp-amsmath.sty**

§ 120 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
```

For nesting $\mathcal{AMS}$ environments:

```
\newcounter{LWR@amsmathdepth}
\setcounter{LWR@amsmathdepth}{0}
```

The following $\mathcal{AMS}$ environments are patched in-place:

\LWR@amsmathenv@@before {⟨*environment name*⟩}

Embeds the environment inside a `lateximage`.

```
\newcommand*{\LWR@amsmathenv@@before}[1]{%
    \begin{BlockClass}{displaymathnumbered}
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[\LWR@amsmathbodynumbered{#1}]*
    \LWR@applyxfakebold%
}
```

\LWR@amsmathenv@before {⟨*environment name*⟩}

Embeds the environment with MathJax or a `lateximage`.

```
\newcommand*{\LWR@amsmathenv@before}[1]{%
    \ifnumequal{\value{LWR@amsmathdepth}}{0}{%
        \LWR@stoppars%
       \ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
        {
            \LWR@syncmathjax
            \boolfalse{LWR@amsmultline}
            \ifstrequal{#1}{multline}{\booltrue{LWR@amsmultline}}{}
            \ifstrequal{#1}{multline*}{\booltrue{LWR@amsmultline}}{}
```

⚠ autonum's "+" environments are not supported by MathJax.

```
            \LWR@beginhideamsmath
        }
        {
            \LWR@amsmathenv@@before{#1}
        }
    }{}
    \addtocounter{LWR@amsmathdepth}{1}
}
```

\LWR@amsmathenv@@after

Embeds the environment inside a `lateximage`.

```
\newcommand*{\LWR@amsmathenv@@after}{%
    \end{lateximage}\end{BlockClass}\LWR@startpars%
}
```

\LWR@amsmathenv@after {⟨*environment name*⟩}

Embeds the environment with MathJax or a `lateximage`.

```
\newcommand*{\LWR@amsmathenv@after}[1]{%
    \ifnumequal{\value{LWR@amsmathdepth}}{1}{%
       \ifboolexpr{bool{mathjax} or ( bool{FormatWP} and bool{WPMarkMath} ) }%
        {
            \LWR@endhideamsmath
            \boolfalse{LWR@amsmultline}
            \LWR@addmathjax{#1}{\the\@envbody}
        }
        {\LWR@amsmathenv@@after}
```

Clear the single-use alt text:

```
        \gdef\LWR@ThisAltText{}%
    }{}
    \addtocounter{LWR@amsmathdepth}{-1}
}
```

Env multline

```
\BeforeBeginEnvironment{multline}{\LWR@amsmathenv@before{multline}}

\AfterEndEnvironment{multline}{\LWR@amsmathenv@after{multline}}
```

Env multline*

```
\BeforeBeginEnvironment{multline*}{\LWR@amsmathenv@before{multline*}}

\AfterEndEnvironment{multline*}{\LWR@amsmathenv@after{multline*}}

```

Env gather

```
\BeforeBeginEnvironment{gather}{\LWR@amsmathenv@before{gather}}

\AfterEndEnvironment{gather}{\LWR@amsmathenv@after{gather}}
```

Env gather*

```
\BeforeBeginEnvironment{gather*}{\LWR@amsmathenv@before{gather*}}

\AfterEndEnvironment{gather*}{\LWR@amsmathenv@after{gather*}}
```

Env align

```
\BeforeBeginEnvironment{align}{\LWR@amsmathenv@before{align}}

\AfterEndEnvironment{align}{\LWR@amsmathenv@after{align}}
```

Env align*

```
\BeforeBeginEnvironment{align*}{\LWR@amsmathenv@before{align*}}

\AfterEndEnvironment{align*}{\LWR@amsmathenv@after{align*}}
```

Env flalign

```
\BeforeBeginEnvironment{flalign}{\LWR@amsmathenv@before{flalign}}

\AfterEndEnvironment{flalign}{\LWR@amsmathenv@after{flalign}}
```

Env flalign*

```
\BeforeBeginEnvironment{flalign*}{\LWR@amsmathenv@before{flalign*}}

\AfterEndEnvironment{flalign*}{\LWR@amsmathenv@after{flalign*}}
```

Env alignat

```
\BeforeBeginEnvironment{alignat}{\LWR@amsmathenv@before{alignat}}

\AfterEndEnvironment{alignat}{\LWR@amsmathenv@after{alignat}}
```

Env alignat*

```
\BeforeBeginEnvironment{alignat*}{\LWR@amsmathenv@before{alignat*}}

\AfterEndEnvironment{alignat*}{\LWR@amsmathenv@after{alignat*}}
```

For MathJax:

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\intertext}[1]{\\ \text{#1}\notag \\}}
\end{warpMathJax}
```

File 21 **lwarp-amsthm.sty**

§ 121 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}.\MessageBreak
        Package amsthm may be loaded by something else,\MessageBreak
        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 22 **lwarp-anonchap.sty**

§ 122 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 23 **lwarp-anysize.sty**

§ 123 Package anysize

(*Emulates or patches code by* Michael Salzenberg, Thomas Esser.)

Pkg anysize anysize is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{anysize}[1994/08/13]
```

```
\def\papersize#1#2{}
\def\marginsize#1#2#3#4{}
```

File 24 **lwarp-appendix.sty**

§ 124 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 25 **lwarp-ar.sty**

§ 125 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}
```

For MathJax:

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\AR}{\mathit{A\!\!R}}}
\CustomizeMathJax{\newcommand{\ARb}{\boldsymbol{A\!\!R}}}
\end{warpMathJax}
```

File 26 **lwarp-arabicfront.sty**

§ 126 Package arabicfront

Pkg arabicfront arabicfront is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{arabicfront}[2006/09/03]
```

File 27 **lwarp-array.sty**

§ 127 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}

\providecommand*{\LWR@HTML@tabularnewline}{\LWR@tabularendofline}
\LWR@formatted{tabularnewline}
```

For MathJax:

```
\CustomizeMathJax{
\newcommand{\multicolumn}[3]{#3}% only uses one cell
}
```

File 28 **lwarp-arydshln.sty**

§ 128 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 `latexmage`.

CSS is not able to display a double-dashed border, so a single-dashed rule is displayed as a single-dashed border, and a double-dashed rule is displayed as a thicker single-dashed border.

For MathJax, limited emulation is provided for math mode.

for HTML output:

array is required to allow `\newcolumn` below.

```
\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 `latexmage`, 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 `latexmage`:

```
\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}}


\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\firsthdashline}[1][]{\hdashline}}
\CustomizeMathJax{\let\lasthdashline\firsthdashline}
\CustomizeMathJax{\let\cdashline\cline}
\end{warpMathJax}
```

File 29 **lwarp-asymptote.sty**

§ 129 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-~\PackageDiagramAltText]%
}
\AfterEndEnvironment{asy}{\end{lateximage}}

\xpatchcmd{\asyinclude}
    {\begingroup}
    {\begin{lateximage}[-asymptote-~\PackageDiagramAltText]}
    {}
    {\LWR@patcherror{asymptote}{asyinclude-begingroup}}

\xpatchcmd{\asyinclude}
    {\endgroup}
    {\end{lateximage}}
    {}
    {\LWR@patcherror{asymptote}{asyinclude-endgroup}}
```

File 30 **lwarp-atbegshi.sty**

§ 130 Package

atbegshi

(*Emulates or patches code by* Heiko Oberdiek.)

Pkg atbegshi atbegshi is ignored.

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 31 **lwarp-attachfile.sty**

§ 131 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 32 **lwarp-attachfile2.sty**

§ 132 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 33 **lwarp-authblk.sty**

§ 133 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 66.8.

(*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 34 **lwarp-autonum.sty**

§ 134 Package

autonum

Pkg autonum autonum is ignored.

⚠ numbering All equations are numbered in HTML output.

MathJax does not support the "+" environments.

for HTML output:

```
\LWR@ProvidesPackageDrop{autonum}[2015/01/18]

\RequirePackage{amsmath}


\newenvironment{equation+}{\equation}{\endequation}


\newenvironment{gather+}{\gather}{\endgather}

\BeforeBeginEnvironment{gather+}{\LWR@amsmathenv@@before{gather+}}

\AfterEndEnvironment{gather+}{\LWR@amsmathenv@@after}
```

```
13
14
15 \newenvironment{multline+}{\multline}{\endmultline}
16
17 \BeforeBeginEnvironment{multline+}{\LWR@amsmathenv@@before{multline+}}
18
19 \AfterEndEnvironment{multline+}{\LWR@amsmathenv@@after}
20
21
22 \newenvironment{flalign+}{\flalign}{\endflalign}
23
24 \BeforeBeginEnvironment{flalign+}{\LWR@amsmathenv@@before{flaline+}}
25
26 \AfterEndEnvironment{flalign+}{\LWR@amsmathenv@@after}
27
28
29 \newenvironment{align+}{\align}{\endalign}
30
31 \BeforeBeginEnvironment{align+}{\LWR@amsmathenv@@before{aline+}}
32
33 \AfterEndEnvironment{align+}{\LWR@amsmathenv@@after}
34
35
36 \newenvironment{alignat+}{\alignat}{\endalignat}
37
38 \BeforeBeginEnvironment{alignat+}{\LWR@amsmathenv@@before{alineat+}}
39
40 \AfterEndEnvironment{alignat+}{\LWR@amsmathenv@@after}
41
42
43 \newenvironment{split+}{\split}{\endsplit}
```

File 35 **lwarp-awesomebox.sty**

§ 135 Package awesomebox

(*Emulates or patches code by* Étienne Deparis.)

Pkg awesomebox awesomebox is patched for use by lwarp.

for HTML output:

```
1 \LWR@ProvidesPackagePass{awesomebox}[2019/07/27]

2 \newcommand*{\LWR@awesomebox@boxborders}{}%
3 \newcommand*{\LWR@awesomebox@contentsborders}{}%
4
5 \newcommand*{\LWR@awesomebox@ruleborders}{%
6     border-top: 1px solid black ;
7     border-bottom: 1px solid black%
8 }
9
10 % \awesomebox[1:vrulecolor][2:hrule][3:title]{4:vrulewidth}{5:icon}{6:iconcolor}{7:content}
11 \RenewDocumentCommand \awesomebox { O{abvrulecolor} O{} o m m m +m }{%
12     \begin{awesomeblock}[#1][#2][#3]{#4}{#5}{#6}
```

```
    #7
    \end{awesomeblock}
}

% \begin{awesomeblock}[1:vrulecolor][2:hrule][3:title]{4:vrulewidth}{5:icon}{6:iconcolor}
% <contents>
% \end{awesomeblock}
\RenewDocumentEnvironment{awesomeblock}{ O{abvrulecolor} O{} o m m m }
{%
    \LWR@forceminwidth{#4}%
    \convertcolorspec{named}{#1}{HTML}\LWR@tempcolor%
    \renewcommand*{\LWR@awesomebox@boxborders}{}%
    \renewcommand*{\LWR@awesomebox@contentsborders}{}%
    \ifdefstrequal{\abShortLine}{#2}{%
      \renewcommand*{\LWR@awesomebox@contentsborders}{\LWR@awesomebox@ruleborders}%
    }{}%
    \ifdefstrequal{\abLongLine}{#2}{%
      \renewcommand*{\LWR@awesomebox@boxborders}{\LWR@awesomebox@ruleborders}%
    }{}%
    \begin{BlockClass}[\LWR@awesomebox@boxborders]{awesomebox}
    \begin{BlockClass}[%
        margin-left: 2\% ;
        vertical-align: top
    ]{minipage}
        \color{#6}\Huge #5
    \end{BlockClass}
    \begin{BlockClass}[%
        width:75\% ;
        vertical-align: top ;
        padding-left: 1em ;
        \LWR@awesomebox@contentsborders ;
        border-left: \LWR@printlength{\LWR@atleastonept} %
            solid \LWR@origpound\LWR@tempcolor%
    ]{minipage}
        \IfValueTF{#3}{#3\newline}{}
}
{%
    \end{BlockClass}
    \end{BlockClass}
}
```

File 36 **lwarp-axessibility.sty**

§ 136 Package

axessibility

Pkg axessibility

axessibility is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{axessibility}

\newcommand{\wrap}[1]{}
```

File 37 **lwarp-axodraw2.sty**

§ 137 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-~\PackageDiagramAltText]%
}

\AfterEndEnvironment{axopicture}{\end{lateximage}}
```

File 38 **lwarp-backnaur.sty**

§ 138 Package backnaur

(*Emulates or patches code by* Adrian P. Robson.)

Pkg backnaur backnaur is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{backnaur}[2019/06/18]

\renewenvironment{bnf}{\eqnarray}{\endeqnarray}
\renewenvironment{bnf*}{\csuse{eqnarray*}}{\csuse{endeqnarray*}}
```

File 39 **lwarp-backref.sty**

§ 139 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 40 **lwarp-balance.sty**

§ 140 Package balance

(*Emulates or patches code by* Patrick W. Daly.)

Pkg balance balance is ignored.

for HTML output: Discard all options for lwarp-balance:

```
\LWR@ProvidesPackageDrop{balance}[1999/02/23]

\newcommand*{\balance}{}
\newcommand*{\nobalance}{}
```

File 41 **lwarp-bbding.sty**

§ 141 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 42 **lwarp-biblatex.sty**

§ 142 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 43 **lwarp-bibunits.sty**

§ 143 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 44 **lwarp-bigdelim.sty**

§ 144 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 class of ldelim or rdelim, and the default CSS sets this to font-size:200%

⚠ use \mrowcell \ldelim and \rdelim use \multirow, so \mrowcell must be used in the proper number of empty cells in the same column below \ldelim or \rdelim, but not in cells which are above or below the delimiter:

```
\begin{tabular}{lll}
<empty> & a & b \\
\ldelim{\{}{3}{.25in}[left ] & c & d \\
\mrowcell & e & f \\
\mrowcell & g & h \\
<empty> & i & j \\
\end{tabular}
```

<-> a b

left { c d
e f
g h

<-> i j

For MathJax, limited emulation is provided which merely prints the delimter and optional text in the first row.

for HTML output: First, remove the temporary definitions of \ldelim and \rdelim, which were previously defined for tabular scanning in case bigdelim was not loaded:

```
\let\ldelim\relax
\let\rdelim\relax
```

Next, load the package's new definitions:

```
\LWR@ProvidesPackagePass{bigdelim}[2018/08/03]
```

\ldelim {⟨*1:delimiter*⟩} {⟨*2:#rows*⟩} {⟨*3:width*⟩} [⟨*4:text*⟩]
\rdelim

```
\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}
```

Limited emulation for MathJax. The delimiter is printed on the first row, along with any optional text.

```
\begin{warpMathJax}
\CustomizeMathJax{
% \ldelim ( [n]{width}[text]
    \newcommand{\LWRldelim}[1][]{\text{#1}~\LWRbigdelim}
    \def\ldelim#1#2#3{\def\LWRbigdelim{#1}\LWRldelim}
% \rdelim ) [n]{width}[text]
    \newcommand{\LWRrdelim}[1][]{\LWRbigdelim~\text{#1}}
    \def\rdelim#1#2#3{\def\LWRbigdelim{#1}\LWRrdelim}
}
\end{warpMathJax}
```

File 45 **lwarp-bigfoot.sty**

§ 145 Package

bigfoot

Pkg bigfoot

bigfoot is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{bigfoot}[2015/08/30]

\RequirePackage{manyfoot}
\RequirePackage{perpage}

\def\RestyleFootnote#1#2{}
\def\FootnoteSpecific#1{}
\def\DefineFootnoteStack#1{}
\def\PushFootnoteMark#1{}
\def\PopFootnoteMark#1{}
\def\hfootfraction{0.9}
\def\vtypefraction{0.7}
\def\FootnoteMinimum{1sp}
\def\FootnoteMainMinimum{0pt}
\newcount\bigfoottolerance
\bigfoottolerance=100
\providecommand\footnotecarryratio{2}
```

File 46 **lwarp-bigstrut.sty**

§ 146 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%
}


\CustomizeMathJax{
    \newcommand{\bigstrut}[1][]{}
}
```

File 47 **lwarp-bitpattern.sty**

§ 147 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-~\PackageDiagramAltText]}
    {}
    {\LWR@patcherror{bitpattern}{bitpattern}}

\xpatchcmd{\bp@Done}
    {\endgroup}
    {\end{lateximage}}
    {}
    {\LWR@patcherror{bitpattern}{bp@Done}}
```

File 48 **lwarp-blowup.sty**

§ 148 Package blowup

Pkg blowup blowup is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{blowup}[2018/01/02]

\newcommand*\blowUp[1]{}
```

File 49 **lwarp-bm.sty**

§ 149 Package bm

(*Emulates or patches code by* David Carlisle, Frank Mittelbach.)

Pkg bm bm is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{bm}[2019/07/24]
```

\DeclareBoldMathCommand must only be used in the preamble, since it adds to the MathJax setup code.

```
\begin{warpMathJax}
\LetLtxMacro\LWR@orig@DeclareBoldMathCommand\DeclareBoldMathCommand

\renewcommand\DeclareBoldMathCommand[3][bold]{%
    \LWR@orig@DeclareBoldMathCommand[#1]{#2}{#3}%
    \CustomizeMathJax{\newcommand{#2}{\boldsymbol{#3}}}%
}

\@onlypreamble\DeclareBoldMathCommand

\CustomizeMathJax{\newcommand{\bm}[1]{\boldsymbol{#1}}}
\end{warpMathJax}
```

File 50 **lwarp-booklet.sty**

§ 150 Package booklet

(*Emulates or patches code by* Peter Wilson.)

Pkg booklet booklet is ignored.

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 51 **lwarp-bookmark.sty**

§ 151 Package bookmark

(*Emulates or patches code by* Heiko Oberdiek.)

Pkg bookmark bookmark is ignored.

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 52 **lwarp-booktabs.sty**

§ 152 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 `lateximage`.

⚠ `\cmidrule` For MathJax, emulation is provided in math mode, but `\cmidrule` trim must not be used.

for HTML output:

If booktabs has already been loaded before lwarp, such as by memoir, use it as-is. If not, the lwarp core will have placed some dummy macros which should be removed before loading the actual booktabs definitions.

```
\@ifpackageloaded{booktabs}{}{
    \LetLtxMacro\toprule\relax
    \LetLtxMacro\midrule\relax
    \LetLtxMacro\cmidrule\cline
    \LetLtxMacro\bottomrule\relax
    \LetLtxMacro\addlinespace\relax
    \LetLtxMacro\morecmidrules\relax
    \LetLtxMacro\specialrule\relax
}
```

Next, load the booktabs package:

```
\LWR@ProvidesPackagePass{booktabs}[2019/10/08]
```

Adjust to work even if xltabular is loaded:

```
% \def\LWR@HTML@@BLTrule{\@BTnormal}
%
% \LWR@formatted{@BLTrule}
\LetLtxMacro\@BLTrule\@BTnormal


\DeclareDocumentCommand{\LWR@HTML@toprule}{o d()}%
    {%
        \IfValueTF{#1}%
            {\LWR@docmidrule[#1](){1-\arabic{LWR@tabletotalLaTeXcols}}}%
            {%
                \ifbool{FormatWP}%
                {\LWR@docmidrule[#1](){1-\arabic{LWR@tabletotalLaTeXcols}}}%
                {\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}}}%
                {\defaddtocounter{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}}}%
            {\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}
```

For MathJax:

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\toprule}[1][]{\hline}}
\CustomizeMathJax{\let\midrule\toprule}
\CustomizeMathJax{\let\bottomrule\toprule}
\CustomizeMathJax{\newcommand{\cmidrule}[2][]{}}% avoid (trim)
\CustomizeMathJax{\newcommand{\morecmidrules}{}}
\CustomizeMathJax{\newcommand{\specialrule}[3]{\hline}}
\CustomizeMathJax{\newcommand{\addlinespace}[1][]{}}
\end{warpMathJax}
```

File 53 **lwarp-bophook.sty**

§ 153 Package

bophook

Pkg bophook bophook is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{bophook}[2001/03/29]

\newcommand*{\AtBeginPage}[1]{}
\newcommand*{\PageLayout}[1]{}
```

File 54 **lwarp-bounddvi.sty**

§ 154 Package bounddvi

Pkg bounddvi bounddvi is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{bounddvi}[2016/12/28]
```

File 55 **lwarp-boxedminipage2e.sty**

§ 155 Package boxedminipage2e

(*Emulates or patches code by* Scott Pakin.)

Pkg boxedminipage2e boxedminipage2e is emulated for HTML, and used as-is for `latexi`mages.

for HTML output: Discard all options for lwarp-boxedminipage2e:

```
\LWR@ProvidesPackagePass{boxedminipage2e}[2015/03/09]

\newenvironment{LWR@HTML@boxedminipage}{%
    \LWR@stoppars%
    \begin{BlockClass}{framebox}%
    \minipage%
}
{%
    \endminipage%
    \end{BlockClass}%
    \LWR@startpars%
}
\LWR@formattedenv{boxedminipage}
```

File 56 **lwarp-braket.sty**

§ 156 Package braket

(*Emulates or patches code by* Donald Arseneau.)

Pkg braket braket works as-is for HTML with SVG math. For MathJax, emulation is provided by MathJax macros.

for HTML output:

```
\LWR@ProvidesPackagePass{braket}% No date is provided by the file.

\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\bra}[1]{\langle#1\vert}}
\CustomizeMathJax{\newcommand{\Bra}[1]{\left\langle#1\right\vert}}
```

```
\CustomizeMathJax{\newcommand{\ket}[1]{\vert#1\rangle}}
\CustomizeMathJax{\newcommand{\Ket}[1]{\left\vert#1\right\rangle}}
\CustomizeMathJax{\newcommand{\braket}[1]{\langle#1\rangle}}
\CustomizeMathJax{\newcommand{\Braket}[1]{\left\langle#1\right\rangle}}
\CustomizeMathJax{\newcommand{\set}[1]{\{#1\}}}
\CustomizeMathJax{\newcommand{\Set}[1]{\left\{#1\right\}}}
\end{warpMathJax}
```

File 57 **lwarp-breakurl.sty**

§ 157 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%
    \LWR@linkcatcodes%
    \LWR@burlaltb%
}

\LetLtxMacro\urlalt\burlalt
```

File 58 **lwarp-breqn.sty**

§ 158 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, and in fact does not work in the print version either.

While using MathJax, breqn objects are converted to SVG images.

for HTML output:

```
\LWR@ProvidesPackagePass{breqn}[2017/01/27]

\setkeys{breqn}{spread={5pt}}
```

```
3
4 \def\eqnumside{R}
5 % \def\eqnumplace{T}
6
7 \BeforeBeginEnvironment{dmath}{
8     \begin{BlockClass}{displaymathnumbered}
9     \LWR@newautoidanchor%
10    \booltrue{LWR@indisplaymathimage}%
11    \begin{lateximage}[-breqn dmath- \MathImageAltText]
12 }
13
14 \AfterEndEnvironment{dmath}{
15    \end{lateximage}\end{BlockClass}
16 }
17
18 \BeforeBeginEnvironment{dmath*}{
19    \begin{BlockClass}{displaymath}
20    \LWR@newautoidanchor%
21    \booltrue{LWR@indisplaymathimage}%
22    \begin{lateximage}[-breqn dmath*- \MathImageAltText]
23 }
24
25 \AfterEndEnvironment{dmath*}{
26    \end{lateximage}\end{BlockClass}
27 }
28
29 \BeforeBeginEnvironment{dseries}{
30    \begin{BlockClass}{displaymathnumbered}
31    \LWR@newautoidanchor%
32    \booltrue{LWR@indisplaymathimage}%
33    \begin{lateximage}[-breqn dseries- \MathImageAltText]
34 }
35
36 \AfterEndEnvironment{dseries}{
37    \end{lateximage}\end{BlockClass}
38 }
39
40 \BeforeBeginEnvironment{dseries*}{
41    \begin{BlockClass}{displaymath}
42    \LWR@newautoidanchor%
43    \booltrue{LWR@indisplaymathimage}%
44    \begin{lateximage}[-breqn dseries*- \MathImageAltText]
45 }
46
47 \AfterEndEnvironment{dseries*}{
48    \end{lateximage}\end{BlockClass}
49 }
50
51 \BeforeBeginEnvironment{dgroup}{
52    \begin{BlockClass}{displaymath}
53    \LWR@newautoidanchor%
54    \booltrue{LWR@indisplaymathimage}%
55    \begin{lateximage}[-breqn dgroup- \MathImageAltText]
56 }
57
```

```
\AfterEndEnvironment{dgroup}{
    \end{lateximage}\end{BlockClass}
}

\BeforeBeginEnvironment{dgroup*}{
    \begin{BlockClass}{displaymath}
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[-breqn dgroup*- \MathImageAltText]
}

\AfterEndEnvironment{dgroup*}{
    \end{lateximage}\end{BlockClass}
}
```

File 59 **lwarp-bsheaders.sty**

§ 159 Package bsheaders

Pkg bsheaders bsheaders is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{bsheaders}[1997/10/06]
```

File 60 **lwarp-bxpapersize.sty**

§ 160 Package bxpapersize

Pkg bxpapersize bxpapersize is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{bxpapersize}[2017/10/08]

\providecommand*\papersizesetup{\bxpapersizesetup}
\newcommand*\bxpapersizesetup[1]{}
```

File 61 **lwarp-bytefield.sty**

§ 161 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-~\PackageDiagramAltText]%
}
```

```
5
6 \AfterEndEnvironment{bytefield}{\end{lateximage}}
```

File 62 **lwarp-cancel.sty**

§ 162 Package

cancel

Pkg cancel

cancel is used as-is for SVG math, and emulated for HTML text output.

for HTML output:

```
1 \LWR@origRequirePackage{lwarp-xcolor}% for \convertcolorspec
2 \LWR@ProvidesPackagePass{cancel}[2013/04/12]
```

\cancelto is math-only, so is used as-is.

```
3 \LetLtxMacro\LWR@origcancel\cancel
4 \LetLtxMacro\LWR@origbcancel\bcancel
5 \LetLtxMacro\LWR@origxcancel\xcancel
6
7 \appto\LWR@restoreorigformatting{%
8 \LetLtxMacro\cancel\LWR@origcancel%
9 \LetLtxMacro\bcancel\LWR@origbcancel%
10 \LetLtxMacro\xcancel\LWR@origxcancel%
11 }
```

\LWR@cancelcolor {⟨*text*⟩} {⟨*color*⟩} {⟨*class*⟩} {⟨*colorstyle*⟩} {⟨*FormatWPstyle*⟩}

Add colors if not empty:

```
12 \newcommand{\LWR@cancelcolor}[5]{%
13 \ifcsempty{#2}%
14 {\InlineClass(#5){#3}{#1}}%
15 {\LWR@htmlspanclass[#5;#4:\LWR@origpound\LWR@tempcolor]{#3}{#1}}%
16 }
```

\cancel {⟨*text*⟩}

```
17 \DeclareRobustCommand{\cancel}[1]{%
18 \begingroup%
19 \CancelColor%
20 \LWR@findcurrenttextcolor%
21 \color{black}%
22 \LWR@cancelcolor{#1}{LWR@tempcolor}{sout}{text-decoration-color}%
23     {text-decoration:line-through}%
24 \endgroup%
25 }
26
27 \LetLtxMacro\bcancel\cancel
28 \LetLtxMacro\xcancel\cancel
```

File 63 **lwarp-canoniclayout.sty**

§ 163 Package canoniclayout

Pkg canoniclayout canoniclayout is ignored.

for HTML output:

```
1 \LWR@ProvidesPackageDrop{canoniclayout}[2011/11/05]

2 \newcommand*{\currentfontletters}{}
3 \newcommand*{\charactersperpage}{}
```

File 64 **lwarp-caption.sty**

§ 164 Package caption

(*Emulates or patches code by* Axel Sommerfeldt.)

Pkg caption caption is patched for use by lwarp.

for HTML output:

```
1 \typeout{---}
2 \typeout{Packages lwarp and caption:}
3 \typeout{If a ‘‘Missing \protect\begin\protect{document\protect}’’ error occurs here,}
4 \typeout{try using: \space \protect\usepackage\protect{caption\protect}\space\protect\captionsetup{opti
5 \typeout{instead of: \protect\usepackage[options]\protect{caption\protect}.}
6 \typeout{---}
7
8 \LWR@ProvidesPackagePass{caption}[2019/09/01]

9 \renewcommand\caption@ibox[3]{%
10   \@testopt{\caption@iibox{#1}{#2}{#3}}{%
11 %         \wd\@tempboxa%
12         \linewidth% lwarp
13   }%
14 % \LWR@traceinfo{caption@ibox: done}%
15 }

16 \long\def\caption@iibox#1#2#3[#4]{%
17   \@testopt{\caption@iiibox{#1}{#2}{#3}{#4}}\captionbox@hj@default
18 }

19 \long\def\caption@iiibox#1#2#3#4[#5]#6{%
20 %   \setbox\@tempboxa\hbox{#6}%
21   \begingroup
22   #1*% set \caption@position
23   \caption@iftop{%
24     \LWR@traceinfo{caption@iiibox top}%
25     \endgroup
26     \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
        #3%                 lwarp
        \else%              lwarp
      \@makecaption{\csname fnum@#1\endcsname}%
                   {\ignorespaces\caption@makeanchor{#3}}%
        \fi%                lwarp

% \par
      \caption@if@minipage\@minipagetrue\@minipagefalse}%
  \caption@end%
}
```

\caption@@@make {⟨*caption label*⟩} {⟨*caption text*⟩}

```
\renewcommand\caption@@@make[2]{%
\LWR@traceinfo{caption@@@make}%
%   \sbox\@tempboxa{#1}%
%   \ifdim\wd\@tempboxa=\z@
%     \let\caption@lsep\relax
%   \fi
  \caption@ifempty{#2}{%
    \let\caption@lsep\@empty
    \let\caption@tfmt\@firstofone
  }%
  \caption@applyfont
```

\caption@fmt with plain format is defined as {#1#2#3\par}:

```
%     \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}%
%       \caption@ifstrut
%         {\ifhmode\@finalstrut\strutbox\fi}%
%         {}%
      }}%
\LWR@traceinfo{caption@@@make done}%
```

```
}
```

\caption@@make@ {⟨⟩} {⟨⟩}

```
\renewcommand{\caption@@make@}[2]{%
  \caption@stepthecounter
  \caption@beginhook
      \caption@@@make{#1}{#2}%
  \caption@endhook
}
```

\caption@makecaption

```
\long\def\caption@makecaption#1#2{%
%   \caption@iftop
%     {\vskip\belowcaptionskip}%
%     {\caption@rule\vskip\abovecaptionskip}%
  \caption@@make{#1}{#2}%
%   \caption@iftop
%     {\vskip\abovecaptionskip\caption@rule}%
%     {\vskip\belowcaptionskip}%
}
```

```
% \DeclareCaptionBox{none}{#2}
\DeclareCaptionBox{parbox}{%
#2%
}
\DeclareCaptionBox{colorbox}{%
#2%
}
```

File 65 **lwarp-cases.sty**

§ 165 Package

cases

(*Emulates or patches code by* Donald Arseneau.)

Pkg cases cases is patched for use by lwarp.

While using MathJax, cases objects are converted to svg math images.

for HTML output:

```
\LWR@ProvidesPackagePass{cases}[2002/05/02]

\BeforeBeginEnvironment{numcases}{
    \begin{BlockClass}{displaymathnumbered}
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[-cases- \MathImageAltText]
}

\AfterEndEnvironment{numcases}{
    \end{lateximage}\end{BlockClass}
```

```
}

\BeforeBeginEnvironment{subnumcases}{
    \begin{BlockClass}{displaymathnumbered}
    \LWR@newautoidanchor%
    \booltrue{LWR@indisplaymathimage}%
    \begin{lateximage}[-cases- \MathImageAltText]
}

\AfterEndEnvironment{subnumcases}{
    \end{lateximage}\end{BlockClass}
}
```

File 66 **lwarp-changebar.sty**

§ 166 Package changebar

Pkg changebar changebar is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{changebar}[2018/03/09]

\newcommand*{\cbstart}{}
\newcommand*{\cbend}{}
\newenvironment*{\changebar}{}{}
\newcommand*{\cbdelete}{}
\newcommand*{\nochnagebars}{}
\newcommand*{\cbcolor}[1]{}
\newlength{\changebarwidth}
\newlength{\deletebarwidth}
\newlength{\changebarsep}
\newcounter{changebargrey}
```

File 67 **lwarp-changelayout.sty**

§ 167 Package changelayout

(*Emulates or patches code by* Ahmed Musa.)

Pkg changelayout changelayout is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{changelayout}[2009/10/07]

\renewrobustcmd\cpl@backtodefaults{}

\renewrobustcmd\cpl@checkifoddpage{%
  \cpl@oddpagefalse%
}

\renewrobustcmd\changepagelayout[1]{%
  \setkeys[KV]{changelay}{#1}%
```

```
}

\renewrobustcmd{\changetextlayout}[1]{\changepagelayout{#1}}

\renewrobustcmd\adjustpagelayout[1]{%
  \setkeys[KV@X]{changelay}{#1}%
}

\renewrobustcmd{\adjusttextlayout}[1]{\adjustpagelayout{#1}}

\renewrobustcmd\adjusttextwidth[1]{%
  \setkeys[KV]{changelay}{#1}%
   \begin{BlockClass}[color:\LWR@colorstyle{named}{\cpl@textcolor}]{changelayout}
        \color{\cpl@textcolor}%
        \cpl@content
    \end{BlockClass}
}
```

File 68 **lwarp-changepage.sty**

§ 168 Package changepage

(*Emulates or patches code by* Peter Wilson.)

Pkg changepage changepage is ignored.

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 69 **lwarp-changes.sty**

§ 169 Package changes

(*Emulates or patches code by* Ekkart Kleinod.)

Pkg changes changes is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{changes}[2019/01/26]
```

\BaseJobname is added to the label in case xr or xr-hyper are used.

```
\renewcommand{\ChangesListline}[4]{%
    \IfSubStr{\Changes@loc@show}{#1}{%
        \LWR@startpars%
        #2: #3 \qquad
        \nameref{\BaseJobname-autopage-#4}%
        \LWR@stoppars%
    }{}%
}

\renewcommand{\Changes@summaryline}[4]{%
    \IfSubStr{\Changes@loc@show}{#1}{%
       \ifthenelse{\not\equal{\Changes@loc@style}{compactsummary} \or #2 > 0}%
            {%
                #3:~#2#4%
            }{}%
    }{}%
}

\xpatchcmd{\listofchanges}
    {{\changesauthorname: \changesanonymousname}}
    {{\LWR@textcurrentcolor{\changesauthorname: \changesanonymousname}}}
    {}
    {\LWR@patcherror{changes}{listofchanges A}}

\xpatchcmd{\listofchanges}
    {\changesauthorname: \Changes@InID}
    {\LWR@textcurrentcolor{\changesauthorname: \Changes@InID}}
    {}
    {\LWR@patcherror{changes}{listofchanges B}}

\xpatchcmd{\listofchanges}
    {(\Changes@InName)}
    {\LWR@textcurrentcolor{(\Changes@InName)}}
    {}
    {\LWR@patcherror{changes}{listofchanges C}}

\xpatchcmd{\listofchanges}
    {\parbox{\Changes@summary@width}}
    {}
    {}
    {\LWR@patcherror{changes}{listofchanges D}}

\xpatchcmd{\Changes@Markup@comment}
    {%
        \ifthenelse{\isAnonymous{#2}}%
            {\textbf{[\arabic{Changes@commentCount#2}]:} }%
            {\textbf{[#3~\arabic{Changes@commentCount#2}]:} }%
        #1%
    }
```

```
    {%
        \LWR@textcurrentcolor{%         lwarp
        \ifthenelse{\isAnonymous{#2}}%
            {\textbf{[\arabic{Changes@commentCount#2}]:} }%
            {\textbf{[#3~\arabic{Changes@commentCount#2}]:} }%
        #1%
        }%                              lwarp
    }
    {}
    {\LWR@patcherror{changes}{\Changes@Markup@comment A}}

\xpatchcmd{\Changes@Markup@comment}
    {%
        \uwave{%
            \ifthenelse{\isAnonymous{#2}}%
                {\textbf{[\arabic{Changes@commentCount#2}]:} }%
                {\textbf{[#3~\arabic{Changes@commentCount#2}]:} }%
            #1%
        }%
    }
    {%
        \LWR@textcurrentcolor{%         lwarp
        \uwave{%
            \ifthenelse{\isAnonymous{#2}}%
                {\textbf{[\arabic{Changes@commentCount#2}]:} }%
                {\textbf{[#3~\arabic{Changes@commentCount#2}]:} }%
            #1%
        }%
        }%                              lwarp
    }
    {}
    {\LWR@patcherror{changes}{\Changes@Markup@comment B}}

\xpatchcmd{\Changes@output}
    {\Changes@Markup@author{\Changes@output@author{#2}{left}}}
   {\LWR@textcurrentcolor{\Changes@Markup@author{\Changes@output@author{#2}{left}}}}
    {}
    {\LWR@patcherror{changes}{Changes@output A}}

\xpatchcmd{\Changes@output}
    {%
        \ifthenelse{\equal{#1}{added}}{\Changes@Markup@added{#3}}{}%
        \ifthenelse{\equal{#1}{deleted}}{\Changes@Markup@deleted{#4}}{}%
        \ifthenelse{\equal{#1}{replaced}}{%
            {\Changes@Markup@added{#3}}\allowbreak\Changes@Markup@deleted{#4}%
        }{}%
        \ifthenelse{\equal{#1}{highlight}}{\Changes@Markup@highlight{#3}}{}%
    }
    {%
        \LWR@textcurrentcolor{%
        \ifthenelse{\equal{#1}{added}}{\Changes@Markup@added{#3}}{}%
        \ifthenelse{\equal{#1}{deleted}}{\Changes@Markup@deleted{#4}}{}%
        \ifthenelse{\equal{#1}{replaced}}{%
            {\Changes@Markup@added{#3}}\allowbreak\Changes@Markup@deleted{#4}%
        }{}%
```

```
        \ifthenelse{\equal{#1}{highlight}}{\Changes@Markup@highlight{#3}}{}%
        }%
    }
    {}
    {\LWR@patcherror{changes}{Changes@output B}}

\xpatchcmd{\Changes@output}
    {\Changes@Markup@author{\Changes@output@author{#2}{right}}}
   {\LWR@textcurrentcolor{\Changes@Markup@author{\Changes@output@author{#2}{right}}}}
    {}
    {\LWR@patcherror{changes}{Changes@output C}}
```

File 70 **lwarp-chappg.sty**

§ 170 Package chappg

(*Emulates or patches code by* Robin Fairbairns.)

Pkg chappg chappg is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{chappg}[2006/05/09]

\renewcommand{\pagenumbering}[2][]{}
\providecommand{\chappgsep}{--}
```

File 71 **lwarp-chapterbib.sty**

§ 171 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 72 **lwarp-chemfig.sty**

§ 172 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 `latexmage`:

```
\begin{lateximage}[-chemfig-~\PackageDiagramAltText]
\chemfig{. . . }
\polymerdelim[. . . ]{. . . }
\end{lateximage}
```

The images are not hashed because they depend on external settings which may be changed at any time, and are unlikely to be reused inline anyhow.

for HTML output:

```
\LWR@ProvidesPackagePass{chemfig}

\catcode`\_=11

\@ifpackagelater{chemfig}{2019/04/18}%
{% 2019/04/18 or newer

    \LetLtxMacro\LWR@chemfig@origchemfig\chemfig

    \renewcommand*{\chemfig}[2][]{%
        \begin{lateximage}[-chemfig-~\PackageDiagramAltText]%
        \LWR@chemfig@origchemfig[#1]{#2}%
        \end{lateximage}%
    }

    \GlobalLetLtxMacro\LWR@chemfig@origCF_lewisc\CF_lewisc
    \gdef\CF_lewisc#1,#2\_nil{%
    \begin{lateximage}[-chemfig-~\PackageDiagramAltText]%
    \LWR@chemfig@origCF_lewisc#1,#2\_nil
    \end{lateximage}
    }

    \gpreto{\schemestart}{%
        \begin{lateximage}[-chemfig-~\PackageDiagramAltText]%
    }
    \gappto{\CF_schemestop}{\end{lateximage}}

}% 2019/04/18 or newer
{% older than 2019/04/18

    \LetLtxMacro\LWR@chemfig@origchemfig\chemfig

    \DeclareDocumentCommand\chemfig{s O{} O{} m}{%
        \begin{lateximage}[-chemfig-~\PackageDiagramAltText]%
        \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-~\PackageDiagramAltText]%
    \LWR@chemfig@origCF@lewis@b{#1}{#2}%
```

```
    \end{lateximage}%
    }

    \preto{\schemestart}{%
        \begin{lateximage}[-chemfig-~\PackageDiagramAltText]%
    }
    \appto{\CF@schemestop}{\end{lateximage}}

}% older than 2019/04/18

\catcode`\_=8%



\LetLtxMacro\LWR@chemfig@origchemleft\chemleft

\def\chemleft#1#2\chemright#3{%
\begin{lateximage}[-chemfig-~\PackageDiagramAltText]%
\LWR@chemfig@origchemleft#1#2\chemright#3%
\end{lateximage}%
}

\LetLtxMacro\LWR@chemfig@origchemup\chemup

\def\chemup#1#2\chemdown#3{%
\begin{lateximage}[-chemfig-~\PackageDiagramAltText]%
\LWR@chemfig@origchemup#1#2\chemdown#3%
\end{lateximage}%
}
```

File 73 **lwarp-chemformula.sty**

§ 173 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}[2019/10/13]

\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.

```
\@ifpackagelater{chemformula}{2019/10/13}{
\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:NV
            \l__chemformula_compound_tl
            \l__chemformula_catcodes_tl
          \chemformula_write:V \l__chemformula_compound_tl
        }
    \group_end:
    }
    \endgroup
  }
}% later than 2019/10/13
{% earlier than 2019/10/13
\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
  }
}% earlier than 2019/10/13
```

\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
      }
  }

\ExplSyntaxOff
```

File 74 **lwarp-chemgreek.sty**

§ 174 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:

```
\LWR@ProvidesPackagePass{chemgreek}[2016/02/10]

\ExplSyntaxOn

\cs_gset_protected:Npn \chemgreek_text:n #1
  { { \text {#1} } }

\appto\LWR@restoreorigformatting{%
\cs_set_protected:Npn \chemgreek_text:n #1%
  { \ensuremath { \text {#1} } }%
}

\ExplSyntaxOff
```

File 75 **lwarp-chemmacros.sty**

§ 175 Package chemmacros

(*Emulates or patches code by* Clemens Niederberger.)

Pkg chemmacros chemmacros is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{chemmacros}[2017/08/28]
```

SVG file hashing assumes that the relevent options are constant for the entire document.

§ 175.1 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}
```

§ 175.2 Code

§ 175.3 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
```

§ 175.4 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 86.

```
\DeclareDocumentEnvironment{redoxreaction}{m m}
{\begin{lateximage}[-chemmacros- redoxreaction]}
{\end{lateximage}}
```

```
\ExplSyntaxOn
```

§ 175.5 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
```

§ 175.6 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
```

§ 175.7 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
```

§ 175.8 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
```

§ 175.9 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
```

§ 175.10 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
```

§ 175.11 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
```

§ 175.12 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
```

§ 175.13 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
        \ifdefvoid{\LWR@ThisAltText}{%                lwarp
            \ThisAltText{-chemmacros-~reaction}%      lwarp
        }{}%                                          lwarp
        \chemmacros_add_reaction_description:n {##1}
        \__chemmacros_begin_reaction:
        \chemmacros_reaction_read:nnw {#2} {#4}
      }
      {
        \__chemmacros_end_reaction:
        \gdef\LWR@ThisAltText{}%                      lwarp
      }
  }
\cs_generate_variant:Nn \chemmacros_declare_reaction_env:nnnn {nnnV}

\RenewChemReaction {reaction}   {equation}
\RenewChemReaction {reaction*}  {equation*}
\RenewChemReaction {reactions}  {align}
\RenewChemReaction {reactions*} {align*}

}{}% \@ifchemmacrosmoduleloaded
}% AtBeginDocument
```

§ 175.14 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
```

§ 175.15 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
```

§ 175.16 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
        {
          \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}
      \tl_if_blank:VF \g__chemmacros_nmr_element_coupled_tl
        {
          \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
    \group_end:
  }
\EndChemCompatibility


\cs_gset_protected:Npn \chemmacros_nmr_position:n #1
```

```
  {
    \chemmacros_chemformula:x
      {
        \exp_not:V \g__chemmacros_nmr_element_tl
        \bool_if:NF \l__chemmacros_nmr_position_side_bool
          {
            \tl_if_eq:NnTF \l__chemmacros_nmr_position_tl {^}% lwarp
            { \textsuperscript{\exp_not:n { {#1} }} }% lwarp
            { \textsubscript{\exp_not:n { {#1} }} }% lwarp
%             \exp_not:V \l__chemmacros_nmr_position_tl
%             \exp_not:n { {#1} }
          }
      }
    \bool_if:NT \l__chemmacros_nmr_position_side_bool
      {
        \tl_use:N \l__chemmacros_nmr_position_tl
        \__chemmacros_nmr_position:n {#1}
      }
  }

\cs_gset_protected:Npn \__chemmacros_nmr_coupling:w (#1;#2)
  {
    \tl_set:Nn \l__chemmacros_nmr_coupling_bonds_tl
      {
        \l__chemmacros_nmr_coupling_bonds_pre_tl
        #1
        \l__chemmacros_nmr_coupling_bonds_post_tl
      }
    \bool_if:NTF \l__chemmacros_nmr_coupling_nuclei_sub_bool
      {
        \tl_set:Nn \l__chemmacros_nmr_coupling_nuclei_tl
          {
%             \c_math_subscript_token
            \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

% lwarp
                    { \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 }
      {
% lwarp
        \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
```

§ 175.17 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 76 **lwarp-chemnum.sty**

§ 176 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 77 **lwarp-chkfloat.sty**

§ 177 Package chkfloat

Pkg chkfloat chkfloat is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{chkfloat}[2012/08/19]
```

File 78 **lwarp-chngpage.sty**

§ 178 Package chngpage

(*Emulates or patches code by* Peter Wilson.)

Pkg chngpage chngpage is ignored.

for HTML output: Discard all options for lwarp-chngpage:

```
\LWR@ProvidesPackageDrop{chngpage}[2009/10/20]
\LWR@origRequirePackage{lwarp-changepage}
```

File 79 **lwarp-cite.sty**

§ 179 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 80 **lwarp-CJK.sty**

§ 180 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 81 **lwarp-CJKutf8.sty**

§ 181 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 82 **lwarp-clrdblpg.sty**

§ 182 Package clrdblpg

Pkg clrdblpg clrdblpg is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{clrdblpg}[2018/04/21]
```

File 83 **lwarp-cmdtrack.sty**

§ 183 Package cmdtrack

Pkg cmdtrack cmdtrack is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{cmdtrack}[2012/12/18]

\newcommand{\untrack}[1]{}
```

File 84 **lwarp-color.sty**

§ 184 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 85 **lwarp-colortbl.sty**

§ 185 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]{}

\AtBeginDocument{\LWR@expandableformatted{doublerulesepcolor}}
```

[⟨*model*⟩] {⟨*color*⟩}

\LWR@doublerulesepcolornexttoken The version for use inside a `tabular`.

```
\newcommand{\LWR@HTML@doublerulesepcolornexttoken}[2][named]{\LWR@getmynexttoken}

\AtBeginDocument{\LWR@expandableformatted{doublerulesepcolornexttoken}}
```

File 86 **lwarp-continue.sty**

§ 186 Package continue

Pkg continue continue is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{continue}}[2018/12/09]

\newcommand*{\flagcont}{}
\newcommand*{\flagend}{}
\newcommand*{\flagword}{}
\newcommand*{\preflagword}{}
\newcommand*{\postflagword}{}
\newlength\contsep
\newlength\contdrop
```

File 87 **lwarp-copyrightbox.sty**

§ 187 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:

```
\LWR@ProvidesPackageDrop{copyrightbox}[2011/11/27]

\newcommand{\copyrightbox}[3][r]{%
\begin{BlockClass}[
    display: inline-flex;
    flex-direction: column ;
]{copyrightbox}
\begin{BlockClass}{copyrightboxcontents}
#2
\end{BlockClass}
\begin{BlockClass}{copyrightboxnote}
#3
\end{BlockClass}
\end{BlockClass}
}

\newcommand{\CRB@setcopyrightfont}{}
\newcommand{\CRB@setcopyrightparagraphstyle}{}
```

File 88 **lwarp-crop.sty**

§ 188 Package crop

(*Emulates or patches code by* Melchior FRANZ.)

Pkg crop crop is ignored.

for HTML output: Discard all options for lwarp-crop:

```
\LWR@ProvidesPackageDrop{crop}[2003/05/20]

\newcommand*{\crop}[1][]{}
\newcommand*{\cropdef}[6][]{}
```

File 89 **lwarp-ctable.sty**

§ 189 Package ctable

(*Emulates or patches code by* Wybo Dekker.)

Pkg ctable ctable is patched for use by lwarp.

⚠ Misplaced alignment tab character &

Use `\StartDefiningTabulars` before one or more `\ctables`, and `\StopDefiningTabulars` after. These change the meaning of the ampersand & character.

for HTML output:

```
\LWR@ProvidesPackagePass{ctable}[2015/10/17]
```

The following is in the original:

```
\newcommand{\LWR@HTML@ctable}[4][]{%
   \let\@CTtaborfig  \@dfltCTtaborfig
   \let\@CTalign     \@dfltCTalign
   \let\@CTsideways  \@dfltCTsideways
   \let\@CTcontinued \empty
   \let\@CTpos       \@dfltCTpos
   \let\@CTcaption   \empty
   \let\@CTcap       \undefined
   \let\@CTlabel     \empty
   \let\@CTbotcap    \@dfltCTbotcap
   \let\@CTstarred   \@dfltCTstarred
   \let\@CTsuper     \@dfltCTsuper
   \let\@CTnotespar  \@dfltCTnotespar
   \let\@CTdoinside  \@dfltCTdoinside
   \let\@CTbgopacity \@dfltCTbgopacity
   \@CTframerule     \@dfltCTframerule
   \@CTcaptionskip   \@dfltCTcaptionskip
   \@CTframesep      \@dfltCTframesep
```

```
\@CTwidth         \@dfltCTwidth
\@CTmaxwidth      \@dfltCTmaxwidth
\@CTmincapwidth   \@dfltCTmincapwidth
\@CTfooterwidth   \@dfltCTfooterwidth
\def\@CTfgactual {@dfltCTframefg}%
\def\@CTbgactual {@dfltCTframebg}%
\def\@CTbeg      {\begin{\@CTsideways\@CTtaborfig\@CTstarred}}%
\def\@CTbegin    {\@CTbeg}%
\def\@CTend      {\end{\@CTsideways\@CTtaborfig\@CTstarred}}%
\setkeys{CT}{#1}%
\ifx\@CTcap\undefined\let\@CTcap\@CTcaption\fi
\ifx\@CTcap\empty
  \if@CTcaptionloaded\else
    \PackageWarningNoLine{lwarp-ctable}{\MessageBreak
       An empty cap= option prevents lot/loc entry only\MessageBreak
       if the caption package is loaded!}
  \fi
\fi
\if@CTinmemoir\else
   \ifx\@CTbotcap\undefined
      \PackageError{lwarp-ctable}{\MessageBreak
         You can, currently, use the sidecap option only with\MessageBreak
         memoir documents. Use topcap or botcap only}
   \fi
\fi
\ifdim\@CTwidth=0pt\else
   \ifdim\@CTmaxwidth=0pt\else
      \PackageError{lwarp-ctable}{\MessageBreak
        You may not use the width and maxwidth options together\MessageBreak
         Use either width or maxwidth}
   \fi
\fi
\ifx\@CTpos\empty
   \ifx\@CTsideways\empty\else
   \PackageError{lwarp-ctable}{\MessageBreak
      You may not use the pos and sideways options together\MessageBreak
      Rotated tables and figures are always typeset on a separate page}
   \fi
\fi
\ifx\@CTcaption\empty
   \ifx\@CTlabel\empty\else
      \PackageError{lwarp-ctable}{\MessageBreak
         You may not label a captionless table\MessageBreak
         Such a label can't be referenced}
   \fi
\fi
```

Some of the original, regarding computing the width of `\CT@t`, is removed here.

```
\@CTbegin
   \ifx\@CTcontinued\empty\else\addtocounter{\@CTtaborfig}{-1}\fi
   \@CTalign
```

lwarp's patches begin here:

```
    \begin{center}
        \setlength{\fboxrule}{\@CTframerule}
        \setlength{\fboxsep}{\@CTframesep}
        \LWR@forceminwidth{\fboxrule}%  lwarp
        \convertcolorspec{named}{\@CTbgactual}{HTML}\LWR@tempcolor% lwarp
        \begin{BlockClass}[%                              lwarp
            border:
                \LWR@printlength{\LWR@atleastonept}
                solid
                \LWR@colorstyle{named}{\@CTfgactual} ; %
            padding:\LWR@printlength{\fboxsep} ; %
            \ifdefstring{\LWR@tempcolor}{FFFFFF}{}{%
                background: \LWR@colorstyle{named}{\@CTbgactual} ; %
            }%
        ]{fminipage}%       lwarp
         \ifx\@CTbotcap\@CTfalse\@CTcaption\vskip\@CTcaptionskip\fi
         \ifx\@CTbotcap\undefined%
             \begin{sidecaption}[\@CTcap]{\@CTcaption}[\@CTlabel]
         \fi
         \@CTdoinside
         \begin{tabularx}{\linewidth}{#2}%      lwarp
            #4%
         \end{tabularx}%                    lwarp
         \def\@CTfootnotes{#3}%
         \ifx#3\empty\else{% append footnotes, if any
            \begin{BlockClass}{tnotes}%     lwarp
            #3
            \end{BlockClass}%               lwarp
         }
         \fi
         \ifx\@CTbotcap\undefined\end{sidecaption}\fi
         \ifx\@CTbotcap\@CTtrue\vskip\@CTcaptionskip\@CTcaption\fi
        \end{BlockClass}
     \end{center}
    \@CTend
}
\LWR@formatted{ctable}
```

Required to properly detect the toprule:

```
\LetLtxMacro\FL\toprule
```

Table notes are redefined for HTML:

```
\newcommand{\LWR@HTML@tmark}[1][a]{%
   \textsuperscript{\textrm{\textit{#1}}}
}
\LWR@formatted{tmark}

\newcommand{\LWR@HTML@tnote}[2][a]{%
    \tmark[#1]\,#2\par
}
\LWR@formatted{tnote}
```

File 90 **lwarp-cuted.sty**

§ 190 Package cuted

(*Emulates or patches code by* Sigitas Tolušis.)

Pkg cuted cuted is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{cuted}[2012/10/04]

\newenvironment{strip}{}{}
\newskip\stripsep
\def\oldcolsbreak#1{}
```

File 91 **lwarp-cutwin.sty**

§ 191 Package cutwin

(*Emulates or patches code by* Peter Wilson and Alan Hoenig.)

Pkg cutwin cutwin is 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 92 **lwarp-dblfloatfix.sty**

§ 192 Package dblfloatfix

Pkg dblfloatfix dblfloatfix is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{dblfloatfix}[2012/12/31]
```

File 93 **lwarp-dblfnote.sty**

§ 193 Package dblfnote

(*Emulates or patches code by* Hiroshi Nakashima.)

Pkg dblfnote dblfnote is ignored.

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 94 **lwarp-dcolumn.sty**

§ 194 Package dcolumn

Pkg dcolumn dcolumn is emulated by the lwarp core.

```
\LWR@ProvidesPackageDrop{dcolumn}[2014/10/28]
```

File 95 **lwarp-diagbox.sty**

§ 195 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 `latexmage`:

```
\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 96 **lwarp-dingbat.sty**

§ 196 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 97 **lwarp-dprogress.sty**

§ 197 Package dprogress

Pkg dprogress dprogress is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{dprogress}[2008/02/21]
```

File 98 **lwarp-draftcopy.sty**

§ 198 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 99 **lwarp-draftfigure.sty**

§ 199 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 100 **lwarp-draftwatermark.sty**

§ 200 Package draftwatermark

(*Emulates or patches code by* Sergio Callegari.)

Pkg draftwatermark draftwatermark is ignored.

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 101 **lwarp-easy-todo.sty**

§ 201 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 102 **lwarp-ebook.sty**

§ 202 Package

ebook

(*Emulates or patches code by* Jørgen Steensgaard.)

Pkg ebook ebook is ignored.

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 103 **lwarp-ed.sty**

§ 203 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 104 **lwarp-ellipsis.sty**

§ 204 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 105 **lwarp-embrac.sty**

§ 205 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 106 **lwarp-emptypage.sty**

§ 206 Package emptypage

Pkg emptypage emptypage is ignored.

for HTML output: Discard all options for lwarp-emptypage:

```
\LWR@ProvidesPackageDrop{emptypage}[2010/05/30]
```

File 107 **lwarp-endfloat.sty**

§ 207 Package endfloat

Pkg endfloat endfloat is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{endfloat}[2019/04/15]

\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{}
\NewDocumentCommand{\addtodelayedfloat}{s m m}{}
\providecommand{\efloatbegin}{}
\providecommand{\efloatend}{}
\providecommand{\efloatbeginlist}{}
\providecommand{\efloatendlist}{}
```

File 108 **lwarp-endheads.sty**

§ 208 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 109 **lwarp-endnotes.sty**

§ 209 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 110 **lwarp-enumerate.sty**

§ 210 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:
```
1 \LWR@ProvidesPackagePass{enumerate}[2015/07/23]
```

File 111 **lwarp-enumitem.sty**

§ 211 Package enumitem

(*Emulates or patches code by* Javier Bezos.)

Pkg enumitem enumitem is supported with minor adjustments.

for HTML output:
```
1 \LWR@ProvidesPackagePass{enumitem}[2018/11/30]
```

for HTML output:
```
2 \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.

```
3 \let\LWR@enumitem@orignewlist\newlist
4
5 \renewcommand*{\newlist}[3]{%
6 \LWR@enumitem@orignewlist{#1}{#2}{#3}%
7 \AtBeginEnvironment{#1}{\@nameuse{LWR@#2start}}%
8 \AtEndEnvironment{#1}{\@nameuse{LWR@#2end}}%
9 }
10
11 \def\DrawEnumitemLabel{}
```

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

File 112 **lwarp-epigraph.sty**

§ 212 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
\LWR@stoppars%
\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][]{}


\newif\ifepigraphnoindent
\newcommand\epigraphnoindent{\epigraphnoindenttrue}
```

File 113 **lwarp-epsfig.sty**

§ 213 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 114 **lwarp-epstopdf.sty**

§ 214 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 115 **lwarp-epstopdf-base.sty**

§ 215 Package

epstopdf-base

Pkg epstopdf-base

Images with an .eps extension will be converted to .pdf. The HTML output uses the .svg version, so use

⚠ convert to .svg

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 lateximage.

```
\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 116 **lwarp-eqlist.sty**

§ 216 Package eqlist

Pkg eqlist eqlist is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{eqlist}[2002/08/15]
```

```
\newenvironment{eqlist}[1][]{\description}{\enddescription}
\newenvironment{eqlist*}[1][]{\description}{\enddescription}
\newenvironment{Eqlist}[2][]{\description}{\enddescription}
\newenvironment{Eqlist*}[2][]{\description}{\enddescription}
\newcommand*{\longitem}[1][]{\item[#1]}
\newcommand*{\eqlistinit}{}
\newcommand*{\eqliststarinit}{}
\newcommand*{\eqlistinitpar}{}
\def\eqlistlabel#1{#1}
\newcommand{\eqlistauto}[1]{}
\newcommand{\eqlistnoauto}{}
```

File 117 **lwarp-eqparbox.sty**

§ 217 Package eqparbox

(*Emulates or patches code by* Scott Pakin.)

Pkg eqparbox eqparbox is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{eqparbox}[2017/09/03]
```

```
\NewDocumentCommand{\LWR@HTML@eqparbox}{O{t} O{} O{t} m +m}{%
    {%
        \minipagefullwidth%
```

```
        \parbox[#1][#2][#3]{\linewidth}{#5}%
    }%
}
\LWR@formatted{eqparbox}

\NewDocumentCommand{\LWR@HTML@eqmakebox}{o o m}{%
    \makebox[#2]{#3}%
}
\LWR@formatted{eqmakebox}

\NewDocumentCommand{\LWR@HTML@eqframebox}{o o m}{%
    \framebox[#2]{#3}%
}
\LWR@formatted{eqframebox}

\NewDocumentEnvironment{LWR@HTML@eqminipage}{O{t} O{} O{t} m}
{%
    \begingroup%
    \minipagefullwidth%
    \minipage[#1][#2][#3]{\linewidth}%
}%
{%
    \endminipage%
    \endgroup%
}

\newcommand*{\LWR@HTML@eqboxwidth}[1]{.25\linewidth}
\LWR@formatted{eqboxwidth}

\newcommand*{\LWR@HTML@eqsetminwidth}[2]{}
\newcommand*{\LWR@HTML@eqsetmaxwidth}[2]{}

\newcommand*{\LWR@HTML@eqsetminwidthto}[2]{}
\newcommand*{\LWR@HTML@eqsetmaxwidthto}[2]{}
```

File 118 **lwarp-errata.sty**

§ 218 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 119 **lwarp-eso-pic.sty**

§ 219 Package eso-pic

(*Emulates or patches code by* Rolf Niepraschk.)

Pkg eso-pic eso-pic is ignored.

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 120 **lwarp-eurosym.sty**

§ 220 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 121 **lwarp-everypage.sty**

§ 221 Package everypage

(*Emulates or patches code by* Sergio Callegari.)

Pkg everypage everypage is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{everypage}[2007/06/20]

\newcommand*{\AddEverypageHook}[1]{}
\newcommand*{\AddThispageHook}[1]{}
```

File 122 **lwarp-everyshi.sty**

§ 222 Package everyshi

(*Emulates or patches code by* Martin Schröder.)

Pkg everyshi ignored.

for HTML output: Discard all options for lwarp-everyshi:

```
\LWR@ProvidesPackageDrop{everyshi}[2001/05/15]

\newcommand*{\EveryShipout}[1]{}
\newcommand*{\AtNextShipout}[1]{}
```

File 123 **lwarp-extramarks.sty**

§ 223 Package extramarks

(*Emulates or patches code by* Piet van Oostrum.)

Pkg extramarks extramarks is ignored.

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 124 **lwarp-fancybox.sty**

§ 224 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.

fancybox, fancyvrb

⚠ \VerbatimFootnotes

⚠ sectioning or displaymath

If using fancybox or fancyvrb with \VerbatimFootnotes, and using footnotes in a sectioning command or display math, use \footnotemark and \footnotetext:

```
\subsection[Subsection Name]
    {Subsection Name\protect\footnotemark}
\footnotetext{A footnote with \verb+verbtim+.}
```

and likewise for equations or display math.

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

```
\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@footnotebox=\vbox\bgroup%
```

Add to any current footnotes:

```
    \unvbox\LWR@footnotebox%
```

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{}%
```

The footnote text will follow after \V@@@footnotetext has completed.

```
  \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}


\let\LWR@origendSbox\endSbox

\def\endSbox{\endLWR@nestspan\LWR@origendSbox}
```

Beqnarray is adapted for MathJax or enclosed inside a lateximage:

```
\RenewEnviron{Beqnarray}
{\LWR@eqnarrayfactor}

\csgpreto{Beqnarray*}{\boolfalse{LWR@numbereqnarray}}
```

\GenericCaption is enclosed in an HTML block:

```
\renewcommand{\GenericCaption}[1]{%
\LWR@figcaption%
\LWR@isolate{#1}%
\endLWR@figcaption%
}
```

Btrivlist is enclosed in an HTML block. This is a tabular, and does not use \item.

\trivlist {⟨*l/c/r*⟩} [⟨*t/c/b*⟩]

```
\RenewDocumentEnvironment{Btrivlist}{m o}
{%
    \LWR@stoppars%
    \begin{BlockClass}{Btrivlist}%
    \tabular{#1}%
}
{%
    \endtabular%
    \end{BlockClass}%
    \LWR@startpars%
}
```

Btrivlist is also neutralized when used inside a span:

```
\AtBeginEnvironment{LWR@nestspan}{%
\RenewDocumentEnvironment{Btrivlist}{m o}{}{}%
}
```

lwarp's handling of \item is patched to accept fancybox's optional arguments:

```
\let\LWRFB@origitemizeitem\LWR@itemizeitem
\let\LWRFB@origdescitem\LWR@descitem

\RenewDocumentCommand{\LWR@itemizeitem}{d()o}{%
    \IfValueTF{#2}{%
        \LWRFB@origitemizeitem[#2]%
```

```
    }{%
        \LWRFB@origitemizeitem%
    }%
}

\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}%
}
```

```
124
125 \let\LWRFB@LUseVerbatim\LUseVerbatim
126
127 \renewcommand*{\LUseVerbatim}[1]{%
128 \LWR@atbeginverbatim{3}{LVerbatim}%
129 \noindent%
130 \LWRFB@LUseVerbatim{#1}%
131 \LWR@afterendverbatim{.5}%
132 }
133
134 \def\@BUseVerbatim[#1]#2{%
135 \LWR@atbeginverbatim{3}{BVerbatim}%
136 \LWRFB@UseVerbatim{#2}%
137 \LWR@afterendverbatim{.5}%
138 }
```

File 125 **lwarp-fancyhdr.sty**

§ 225 Package

fancyhdr

(*Emulates or patches code by* Piet van Oostrum.)

Pkg fancyhdr fancyhdr is ignored.

for HTML output: Discard all options for lwarp-fancyhdr:

```
1 \LWR@ProvidesPackageDrop{fancyhdr}[2019/01/31]

2 \newcommand*{\fancyhead}[2][]{}
3 \newcommand*{\fancyfoot}[2][]{}
4 \newcommand*{\fancyhf}[2][]{}
5 \newcommand*{\fancypagestyle}[2]{}
6 \newcommand*{\lhead}[2][]{}
7 \newcommand*{\chead}[2][]{}
8 \newcommand*{\rhead}[2][]{}
9 \newcommand*{\lfoot}[2][]{}
10 \newcommand*{\cfoot}[2][]{}
11 \newcommand*{\rfoot}[2][]{}
12 \newcommand*{\headrulewidth}{}
13 \newcommand*{\footrulewidth}{}
14 \newcommand*{\headrule}{}
15 \newcommand*{\footrule}{}
16 \newlength{\headwidth}
17 \newcommand*{\fancyheadoffset}[2][]{}
18 \newcommand*{\fancyfootoffset}[2][]{}
19 \newcommand*{\fancyhfoffset}[2][]{}
20 \newcommand*{\iffloatpage}[2]{#2}
21 \newcommand*{\ifftopfloat}[2]{#2}
22 \newcommand*{\iffbotfloat}[2]{#2}
23 \newcommand*{\iffootnote}[2]{#2}
```

File 126 **lwarp-fancyref.sty**

§ 226 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{lwarp-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 127 **lwarp-fancytabs.sty**

§ 227 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 128 **lwarp-fancyvrb.sty**

§ 228 Package

fancyvrb

(*Emulates or patches code by* Timothy Van Zandt.)

Pkg fancyvrb fancyvrb is supported with some patches.

HTML classes

The fancy verbatim environment is placed inside a of class fancyvrb. The label is placed inside a of class fancyvrblabel. The verbatim text itself is placed inside a of class verbatim.

fancybox, fancyvrb

⚠ \VerbatimFootnotes

⚠ sectioning or displaymath

If using fancybox or fancyvrb with \VerbatimFootnotes, and using footnotes in a sectioning command or display math, use \footnotemark and \footnotetext:

```
\subsection[Subsection Name]
    {Subsection Name\protect\footnotemark}
\footnotetext{A footnote with \verb+verbtim+.}
```

and likewise for equations or display math.

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

```
\RequirePackage{xcolor}% for \convertcolorspec
```

```
2
3 \LWR@ProvidesPackagePass{fancyvrb}[2008/02/07]
```

Initial default patch for fancyvrb:

```
4 \fvset{frame=none}%
```

After the preamble is loaded, after any patches to `Verbatim`:

```
5 \AfterEndPreamble{
6 \LWR@traceinfo{Patching fancyvrb.}
```

`\VerbatimFootnotes` Patched to use the new version.

```
7 \def\VerbatimFootnotes{%
8 \let\@footnotetext\V@footnotetext%
9 \let\footnote\V@footnote%
10 \let\LWR@footnotetext\V@footnotetext% lwarp
11 }
```

`\V@@footnotetext` Patches in a subset of lwarp's `\LWR@footnotetext` to the fancyvrb version of `\V@@footnotetext`.

```
12 \def\V@@footnotetext{%
13 \LWR@traceinfo{V@footnotetext}%
14 \global\setbox\LWR@footnotebox=\vbox\bgroup%
```

Add to any current footnotes:

```
15     \unvbox\LWR@footnotebox%
```

Remember the footnote number for `\ref`:

```
16     \protected@edef\@currentlabel{%
17         \csname p@footnote\endcsname\@thefnmark%
18     }% @currentlabel
```

Use HTML superscripts in the footnote even inside a `lateximage`:

```
19     \renewrobustcmd{\textsuperscript}[1]{\LWR@htmlspan{sup}{##1}}%
```

Use paragraph tags if in a tabular data cell or a `lateximage`:

```
20     \ifthenelse{%
21         \boolean{LWR@doingstartpars} \AND%
22         \cnttest{\value{LWR@lateximagedepth}}{=}{0}%
23     }%
24         {}%
25         {\LWR@htmltagc{\LWR@tagregularparagraph}\LWR@orignewline}%
```

Append the footnote mark to the list:

```
26     \@makefntext{}%
```

The footnote text will follow after `\V@@@footnotetext` has completed.

```
27   \bgroup%
28   \aftergroup{\V@@@footnotetext}%
29   \ignorespaces%
30 }%
```

```
\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{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@{
\BlockClass[\LWR@FVstyle]{fancyvrb}
\LWR@stoppars
\ifx\FV@LabelPositionTopLine\relax\else
    \ifx\FV@LabelBegin\relax\else
        \FancyVerbRuleColor{\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}\LWR@orignewline%
    \fi
\fi
\LWR@atbeginverbatim{0}{verbatim}%
% }%
}

\newcommand*{\LWR@fvendnone}{%
\LWR@traceinfo{fvendnone}%
% \hbox to\z@{
\LWR@afterendverbatim{0}%
\LWR@stoppars%
\ifx\FV@LabelPositionBottomLine\relax\else
    \ifx\FV@LabelEnd\relax\else
        \FancyVerbRuleColor{\LWR@FVfindbordercolor}
        \LWR@htmltagc{%
            div class="fancyvrblabel" % extra space
                style="color: \LWR@origpound\LWR@tempcolor"%
        }
        \LWR@print@textrm{\FV@LabelEnd}
        \LWR@htmltagc{/div}\LWR@orignewline%
    \fi
\fi
\endBlockClass
}

\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 ; % space
\LWR@FVfindbordercolor\LWR@indentHTMLtwo%
border#1: \strip@pt\dimexpr \FV@FrameRule\relax\relax pt % space
solid {\FancyVerbRuleColor{\LWR@origpound\LWR@tempcolor}} ; % space
}

\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@indentHTMLtwo%
    \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@indentHTMLtwo%
    \LWR@FVborderstyle{-top}%
    \LWR@indentHTMLtwo%
    \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@indentHTMLtwo%
    \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@indentHTMLtwo%
    \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@indentHTMLtwo%
    \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
        \fi
      \else
%         \setbox\z@\hbox{\strut\enspace\FV@LabelEnd\enspace\strut}%
        \ifx\FV@LabelPositionBottomLine\relax
```

```
        \else
        \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 129 **lwarp-figcaps.sty**

§ 229 Package figcaps

(*Emulates or patches code by* Patrick W. Daly.)

Pkg `figcaps` figcaps is ignored.

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 130 **lwarp-figsize.sty**

§ 230 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 131 **lwarp-fitbox.sty**

§ 231 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}
}
```

```
7
8 \newcommand*{\fitboxset}[1]{}
9
10 \newdimen\fitboxnatheight
11 \newdimen\fitboxnatwidth
12
13 \newcommand\SetFitboxLayout[3][]{}
```

File 132 **lwarp-fix2col.sty**

§ 232 Package

fix2col

Pkg fix2col fix2col is ignored.

for HTML output:

```
1 \LWR@ProvidesPackageDrop{fix2col}[2015/11/13]
```

File 133 **lwarp-fixme.sty**

§ 233 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:

```
1 \LWR@ProvidesPackagePass{fixme}[2017/03/05]
```

Restore lwarp's version of `\@wrindex`, ignoring the fixme package's `target` option:

```
2 \let\@wrindex\LWR@wrindex
```

Float-related macros required by lwarp:

```
3 \newcommand{\ext@fixme}{lox}
4
5 \renewcommand{\l@fixme}[2]{\hypertocfloat{1}{fixme}{lox}{#1}{#2}}
```

Other modifications:

```
6 \def\FXFaceInlineHTMLStyle{font-weight:bold}
```

```
7
8 \renewcommand*\FXLayoutInline[3]{ %
9 \InlineClass[\FXFaceInlineHTMLStyle]{fixmeinline}%
10     {\@fxtextstd{#1}{#2}{#3}}%
11 }
12
13 \def\FXFaceEnvHTMLStyle{font-weight:bold}
14
15 \renewcommand*\FXEnvLayoutPlainBegin[2]{%
16 \BlockClass[\FXFaceEnvHTMLStyle]{fixmebold}
17 \ignorespaces#2 \fxnotename{#1}: \ignorespaces}
18
19 \renewcommand*\FXEnvLayoutPlainEnd[2]{\endBlockClass}
20
21 \renewcommand*\FXEnvLayoutSignatureBegin[2]{%
22 \BlockClass[\FXFaceEnvHTMLStyle]{fixmebold}
23 \fxnotename{#1}: \ignorespaces}
24
25 \renewcommand*\FXEnvLayoutSignatureEnd[2]{\@fxsignature{#2}\endBlockClass}
26
27 \def\FXFaceSignatureHTMLStyle{font-style:italic}
28
29 \DeclareRobustCommand*\@fxsignature[1]{%
30 \ifthenelse{\equal{#1}{}}%
31     {}%
32     { -- {\InlineClass[\FXFaceSignatureHTMLStyle]{fixmesignature}{#1}}}%
33 }
34
35
36 \def\FXFaceTargetHTMLStyle{font-style:italic}
37
38 \renewcommand\FXTargetLayoutPlain[2]{%
39     \InlineClass[\FXFaceTargetHTMLStyle]{fixmetarget}{#2}%
40 }
```

File 134 **lwarp-fixmetodonotes.sty**

§ 234 Package

fixmetodonotes

(*Emulates or patches code by* Gioele Barabucci.)

Pkg fixmetodonotes fixmetodonotes is patched for use by lwarp.

for HTML output:

```
1 \LWR@ProvidesPackagePass{fixmetodonotes}[2013/04/28]

2 \renewcommand{\NOTES@addtolist}[2]{%
3     \refstepcounter{NOTES@note}%
4 %   \phantomsection% REMOVED
5     \addcontentsline{notes}{NOTES@note}{%
6         \protect\numberline{\theNOTES@note}{{#1}: {#2}}%
7     }%
8 }
9
```

```
\renewcommand{\NOTES@marker}[2]{\fbox{%
    \textcolor{#2}{% WAS \color
        \textbf{#1}}%
    }}

\renewcommand{\NOTES@colorline}[2]{%
  \bgroup%
    \ULon{\LWR@backgroundcolor{#1}{#2}}%
}
```

File 135 **lwarp-flafter.sty**

§ 235 Package

flafter

Pkg flafter flafter is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{flafter}[2018/01/08]
\providecommand\fl@trace[1]{}
```

File 136 **lwarp-flippdf.sty**

§ 236 Package

flippdf

Pkg flippdf flippdf is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{flippdf}[2006/06/30]

\newcommand\FlipPDF{}
\newcommand\UnFlipPDF{}
```

File 137 **lwarp-float.sty**

§ 237 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 75.2 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 138 **lwarp-floatflt.sty**

§ 238 Package

floatflt

(*Emulates or patches code by* Mats Dahlgren.)

Pkg floatflt floatflt is 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}
{%
    \begin{LWR@setvirtualpage}*%
    \ifblank{#3}{%
        \LWR@BlockClassWP{%
            float:right; %
            width: 1.5in; % reasonable dummy width for word processor
            margin:10pt%
        }{}%
        {marginblock}%
    }{%
        \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%
    \end{LWR@setvirtualpage}%
}
```

Env floatingfigure [⟨*placement*⟩] {⟨*width*⟩}

```
\DeclareDocumentEnvironment{floatingfigure}{o m}
  {\begin{KFLTfloatflt@marginfloat}{figure}{#2}}
```

```
31  {\end{KFLTfloatflt@marginfloat}}
```

Env floatingtable [⟨*placement*⟩]

```
32 \DeclareDocumentEnvironment{floatingtable}{o}
33  {\begin{KFLTfloatflt@marginfloat}{table}{}}
34  {\end{KFLTfloatflt@marginfloat}}
```

File 139 **lwarp-floatpag.sty**

§ 239 Package

floatpag

(*Emulates or patches code by* Vytas Statulevičius and Sigitas Tolušis.)

Pkg floatpag floatpag is ignored.

for HTML output: Discard all options for lwarp-floatpag:

```
1 \LWR@ProvidesPackageDrop{floatpag}[2012/05/29]

2 \newcommand*{\floatpagestyle}[1]{}
3 \newcommand*{\rotfloatpagestyle}[1]{}
4 \newcommand*{\thisfloatpagestyle}[1]{}
```

File 140 **lwarp-floatrow.sty**

§ 240 Package

floatrow

(*Emulates or patches code by* Olga Lapko.)

Pkg floatrow floatrow is emulated.

for HTML output:

```
1 \LWR@ProvidesPackageDrop{floatrow}[2008/08/02]
```

⚠ Misplaced alignment tab character &

Use \StartDefiningTabulars and \StopDefiningTabulars before and after defining macros using \ttabbox with a tabular inside. See section 8.10.1.

⚠ subfig package

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

⚠ \FBwidth, \FBheight

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

- Use fixed lengths. lwarp will scale the HTML lengths appropriately.
- Use warpprint and warpHTML environments to select appropriate values for each case.
- Inside a warpHTML environment, manually change \FBwidth or \FBheight before the \ffigbox or \ttabbox. Use \FBwidth or \FBheight normally afterwards; it

will be used as expected in print output, and will use your custom-selected value in HTML output. This custom value will be used repeatedly, until it is manually changed to a new value.

After everything has loaded, remember whether subcaption was loaded. If not, it is assumed that subfig is used instead:

```
\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 `<div>` of class `floatrow`.

```
\newenvironment*{floatrow}[1][2]
{%
    \begin{LWR@setvirtualpage}*%
    \BlockClass{floatrow}%
}
{
    \endBlockClass%
    \end{LWR@setvirtualpage}%
}
```

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}{}}%
{%
    \DeclareFloatingEnvironment[
        placement=\LWR@frowkeyplacement,
        fileext=\LWR@frowkeyfileext
    ]{#1}%
}%
{%
    \DeclareFloatingEnvironment[
        placement=\LWR@frowkeyplacement,
        fileext=\LWR@frowkeyfileext,
        within=\LWR@frowkeywithin
    ]{#1}%
%     \LWR@traceinfo{finished newfloat #1}%
}%
```

Rename the float if a name was given:

```
\ifthenelse{\equal{\LWR@frowkeyname}{}}%
    {}%
    {%
        \SetupFloatingEnvironment{#1}{name={\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 `<div>` 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 141 **lwarp-fltrace.sty**

§ 241 Package fltrace

Pkg fltrace fltrace is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{fltrace}[2018/01/08]
```

```
\def\tracefloats{}
\def\tracefloatsoff{}
\def\tracefloatvals{}
```

File 142 **lwarp-flushend.sty**

§ 242 Package flushend

(*Emulates or patches code by* Sigitas Tolušis.)

Pkg flushend flushend is ignored.

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 143 **lwarp-fnbreak.sty**

§ 243 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 144 **lwarp-fncychap.sty**

§ 244 Package fncychap

(*Emulates or patches code by* Ulf A. Lindgren.)

Pkg fncychap fncychap is ignored.

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 145 **lwarp-fnlineno.sty**

§ 245 Package fnlineno

Pkg fnlineno fnlineno is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{fnlineno}[2011/01/07]
```

File 146 **lwarp-fnpara.sty**

§ 246 Package fnpara

Pkg fnpara fnpara is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{fnpara}
```

File 147 **lwarp-fnpos.sty**

§ 247 Package fnpos

(*Emulates or patches code by* Hiroshi Nakashima.)

Pkg fnpos fnpos is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{fnpos}[1999/07/14]
```

```
\newcommand*{\makeFNbottom}{}
\newcommand*{\makeFNmid}{}
\newcommand*{\makeFNbelow}{}
\newcommand*{\makeFNabove}{}
```

File 148 **lwarp-fontawesome.sty**

§ 248 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]{%
    \LWR@findcurrenttextcolor%
   \begin{lateximage}*[icon][fontawesomexetex#1SZ\LWR@font@size{}CL\LWR@tempcolor]%
    \csuse{\LWR@font@size}%
    \LWR@orig@FA%
    \LWR@orig@symbol{#1}%
    \end{lateximage}%
}

\RenewDocumentCommand{\FA}{}{%
    \LetLtxMacro\symbol\LWR@fontawesome@xelatex@symbol%
}

\else

\newcommand*{\LWR@fontawesome@symbolX}[2]{%
    \LWR@findcurrenttextcolor%
   \begin{lateximage}*[icon #1][fontawesome#2#1SZ\LWR@font@size{}CL\LWR@tempcolor]%
    \csuse{\LWR@font@size}%
    \fontencoding{U}\fontfamily{fontawesome#2}\selectfont%
    \LWR@orig@symbol{#1}%
    \end{lateximage}%
}

\newcommand*{\LWR@fontawesome@symbolone}[1]{%
    \LWR@fontawesome@symbolX{#1}{one}%
}

\newcommand*{\LWR@fontawesome@symboltwo}[1]{%
    \LWR@fontawesome@symbolX{#1}{two}%
}

\newcommand*{\LWR@fontawesome@symbolthree}[1]{%
    \LWR@fontawesome@symbolX{#1}{three}%
```

```
}

\renewrobustcmd\FAone{%
    \LetLtxMacro\symbol\LWR@fontawesome@symbolone%
}

\renewrobustcmd\FAtwo{%
    \LetLtxMacro\symbol\LWR@fontawesome@symboltwo%
}

\renewrobustcmd\FAthree{%
    \LetLtxMacro\symbol\LWR@fontawesome@symbolthree%
}
\fi
```

File 149 **lwarp-fontawesome5.sty**

§ 249 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{
    \LWR@findcurrenttextcolor
  \cs_if_exist:cTF{c__fontawesome_slot_#2_tl}{
    \begin{lateximage}*[#2][fontawesome5#1SZ\LWR@font@size{}CL\LWR@tempcolor]
    \csuse{\LWR@font@size}
    \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 150 **lwarp-fontaxes.sty**

§ 250 Package fontaxes

(*Emulates or patches code by* Andreas Bühmann, Michael Ummels.)

Pkg fontaxes fontaxes is emulated for HTML, and used as-is for print output.

Functionality for small caps is in the lwarp core. Swashes and figure styles are ignored for HTML.

for HTML output:

```
\LWR@ProvidesPackagePass{fontaxes}[2014/03/23]

\newrobustcmd{\LWR@HTML@sscshape}{\LWR@HTML@scshape}
\LWR@formatted{sscshape}
\newcommand*{\LWR@null@sscshape}{}

\newrobustcmd{\LWR@HTML@textssc}[1]{\textsc{#1}}
\LWR@formatted{textssc}
\newcommand*{\LWR@null@textssc}[1]{#1}

\ifdef{\LWR@HTML@swshape}{}{% duplicated by nfssext-cfr
    \newcommand{\LWR@HTML@swshape}{}
    \LWR@formatted{swshape}
    \newcommand*{\LWR@null@swshape}{}

    \newcommand{\LWR@HTML@textsw}[1]{#1}
    \LWR@formatted{textsw}
    \newcommand*{\LWR@null@textsw}[1]{#1}
}

\appto{\LWR@nullfonts}{%
\LetLtxMacro\sscshape\LWR@null@sscshape%
\LetLtxMacro\textssc\LWR@null@textssc%
\LetLtxMacro\swshape\LWR@null@swshape%
\LetLtxMacro\textsw\LWR@null@textsw%
}
```

File 151 **lwarp-fontenc.sty**

§ 251 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 152 **lwarp-footmisc.sty**

§ 252 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 153 **lwarp-footnote.sty**

§ 253 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%
    \ignorespaces%
}
```

Removed print-version formatting, added closing paragraph tag:

```
\def\fn@endfntext{%
    \LWR@htmltagc{/\LWR@tagregularparagraph}%
    \LWR@orignewline%
  \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 154 **lwarp-footnotebackref.sty**

§ 254 Package

footnotebackref

Pkg footnotebackref footnotebackref is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{footnotebackref}[2012/07/01]
```

File 155 **lwarp-footnotehyper.sty**

§ 255 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 156 **lwarp-footnoterange.sty**

§ 256 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 157 **lwarp-footnpag.sty**

§ 257 Package footnpag

Pkg footnpag footnpag is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{footnpag}
```

File 158 **lwarp-foreign.sty**

§ 258 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 159 **lwarp-forest.sty**

§ 259 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-~\PackageDiagramAltText]%     lwarp
  \forest@next{#3}%
    \end{lateximage}%               lwarp
}
```

File 160 **lwarp-framed.sty**

§ 260 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*⟩} {⟨*bottitle*⟩} {⟨*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 161 **lwarp-ftcap.sty**

§ 261 Package ftcap

Pkg ftcap ftcap is ignored.

for HTML output:

```
1 \LWR@ProvidesPackageDrop{ftcap}
```

File 162 **lwarp-ftnright.sty**

§ 262 Package ftnright

Pkg ftnright ftnright is ignored.

for HTML output: Discard all options for lwarp-ftnright:

```
1 \LWR@ProvidesPackageDrop{ftnright}[2014/10/28]
```

File 163 **lwarp-fullminipage.sty**

§ 263 Package fullminipage

Pkg fullminipage fullminipage is ignored.

for HTML output:

```
1 \LWR@ProvidesPackageDrop{fullminipage}[2014/07/06]
2 \newenvironment{fullminipage}[1][]{}{}
```

File 164 **lwarp-fullpage.sty**

§ 264 Package fullpage

Pkg fullpage fullpage is ignored.

for HTML output: Discard all options for lwarp-fullpage:

```
1 \LWR@ProvidesPackageDrop{fullpage}[1994/06/01]
```

File 165 **lwarp-fullwidth.sty**

§ 265 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 166 **lwarp-fwlw.sty**

§ 266 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 167 **lwarp-gentombow.sty**

§ 267 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 168 **lwarp-geometry.sty**

§ 268 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@ProvidesPackageDropA{geometry}{2018/04/16}
```

If geometry is never loaded by the user, it will be loaded by lwarp `\AtBeginDocument`. If this is the case, the page layout should not be changed but the user macros should still be nullified.

```
\ifbool{LWR@allowanothergeometry}{%
```

Assign and set the selected geometry with `reset` prepended. `\AtEndPreamble` lwarp will save this, then set its own geometry.

```
    \edef\LWR@tempone{reset,\@ptionlist{\@currname.\@currext}}%
    \expandafter\LWR@origgeometry\expandafter{\LWR@tempone}%
}{}% LWR@allowanothergeometry
```

The user-level commands are nullified:

```
\renewcommand*{\geometry}[1]{}
\renewcommand*{\newgeometry}[1]{}
\renewcommand*{\restoregeometry}{}
\renewcommand*{\savegeometry}[1]{}
\renewcommand*{\loadgeometry}[1]{}
```

File 169 **lwarp-gloss.sty**

§ 269 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]
```

\BaseJobname is added to the label in case xr or xr-hyper are used.

```
\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{\BaseJobname-autopage-#2}}}%
```

File 170 **lwarp-glossaries.sty**

§ 270 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 171 **lwarp-gmeometric.sty**

§ 271 Package gmeometric

Pkg gmeometric gmeometric is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{gmeometric}[2008/11/22]
```

File 172 **lwarp-graphics.sty**

§ 272 Package graphics

(*Emulates or patches code by* D. P. Carlisle.)

Pkg graphics graphics is emulated.

for HTML output:

```
\LWR@ProvidesPackagePass{graphics}[2017/06/25]
```

§ 272.1 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
```

§ 272.2 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}{\ImageAltText}
```

Set the actions of each of the key/value combinations for `\includegraphics`. Many are ignored.

If an optional width was given, set an HTML style:

```
\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}{%
    \ifthenelse{\equal{#1}{1}}{}{%  must expand #1
        \PackageWarning{lwarp}{%
            It is recommended to use\MessageBreak
            [width=xx\protect\linewidth]\MessageBreak
            instead of [scale=yy],%
        }%
    }%
    \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}{}
```

§ 272.3 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]{%
    \edef\LWR@tempone{#2}%
    \setcounter{LWR@tempcountone}{-1*\real{\LWR@tempone}} % space
    #1transform:rotate(\arabic{LWR@tempcountone}deg); % space
}
```

\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);
}
```

§ 272.4 \includegraphics

\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@ig@wpimagesizes * [⟨*2: options*⟩] [⟨*3: options*⟩] {⟨*4: filename*⟩}

If formatting for a word processor, find and set the actual image size, without rotation, using PDF instead of SVG to find the original bounding box:

```
\newcommand*{\LWR@ig@wpimagesizes}[4]{%
    \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@originincludegraphics*[#2][#3]{#4}%
                }%
            }%
            {%
                \IfValueTF{#2}%
                {%
                    \global\sbox{\LWR@imagesizebox}{%
                        \LWR@originincludegraphics*[#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
}
```

\LWR@ig@htmltag For the HTML reference, add the graphicspath, filename, extension, alt tag, style, and class.

```
\newcommand*{\LWR@ig@htmltag}{%
    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="\AltTextOpen\LWR@igalt\AltTextClose" \LWR@orignewline%
}% end of image tags
```

`\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.

```
    \begin{LWR@setvirtualpage}%
```

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}%

    \ifdefvoid{\LWR@ThisAltText}{%
        \edef\LWR@igalt{\ImageAltText}%
    }{%
        \edef\LWR@igalt{\LWR@ThisAltText}%
    }%
```

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:

```
    \LWR@ig@wpimagesizes{#1}{#2}{#3}{#4}%
```

Create the HTML reference with the graphicspath, filename, extension, alt tag, style, and class:

```
    \LWR@traceinfo{LWR@includegraphicsb: about to create href}%
    \href{\LWR@parsedfilename}%
    {% start of href
        \LWR@traceinfo{LWR@includegraphicsb: about to LWR@htmltag}%
        \LWR@htmltag{\LWR@ig@htmltag}%
    }% end of href
```

Return to original page size and font size:

```
    \end{LWR@setvirtualpage}%
```

Clear the single-use `alt` text:

```
    \gdef\LWR@ThisAltText{}%
    \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
```

§ 272.5 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 173 **lwarp-graphicx.sty**

§ 273 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:

```
1 \LWR@ProvidesPackagePass{graphicx}[2017/06/01]
```

File 174 **lwarp-grffile.sty**

§ 274 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:

```
1 \LWR@ProvidesPackagePass{grffile}[2017/06/30]
```

File 175 **lwarp-grid.sty**

§ 275 Package

grid

Pkg grid

grid is ignored.

for HTML output:

```
1 \LWR@ProvidesPackageDrop{grid}[2009/06/16]
2 \newenvironment*{gridenv}{}{}
```

File 176 **lwarp-grid-system.sty**

§ 276 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 177 **lwarp-gridset.sty**

§ 277 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 178 **lwarp-hang.sty**

§ 278 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 179 **lwarp-hanging.sty**

§ 279 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 180 **lwarp-hypbmsec.sty**

§ 280 Package hypbmsec

Pkg hypbmsec hypbmsec is emulated by the lwarp core.

for HTML output:

```
\LWR@ProvidesPackageDrop{hypbmsec}[2016/05/16]
```

File 181 **lwarp-hypcap.sty**

§ 281 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 182 **lwarp-hypdestopt.sty**

§ 282 Package hypdestopt

Pkg hypdestopt hypdestopt is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{hypdestopt}[2016/05/21]
```

File 183 **lwarp-hypernat.sty**

§ 283 Package hypernat

Pkg hypernat hypernat is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{hypernat}[2001/07/09]
```

File 184 **lwarp-hyperref.sty**

§ 284 Package

hyperref

(*Emulates or patches code by* Sebastian Rahtz, Heiko Oberdiek.)

Pkg hyperref hyperref is emulated.

for HTML output:

```
% \LWR@ProvidesPackageDrop{hyperref}% not allowed
% \ProvidesPackage{lwarp-#1-#2}% not allowed
\PackageInfo{lwarp}{%
Using the lwarp HTML version of package ‘hyperref’,\MessageBreak
and discarding options.\MessageBreak
(Not using \protect\ProvidesPackage, so that other packages\MessageBreak
do not attempt to patch lwarp’s version of ‘hyperref’.)\MessageBreak}

\DeclareOption*{}


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


\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%
    \LWR@linkcatcodes%
    \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%
    \LWR@linkcatcodes%
    \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}%
        "%
        \LWR@addlinktitle%
    }%
    \endgroup%
    \LWR@hyperreffinish%
}

\newrobustcmd*{\LWR@hyperrefb}{%
    \begingroup%
    \LWR@linkcatcodes%
    \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%
```

```
    \LWR@linkcatcodes%
    \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%
    \LWR@linkcatcodes%
    \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%
    \LWR@linkcatcodes%
    \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*⟩}

```
125 \DeclareRobustCommand{\PushButton}[2][]{}
```

\Submit [⟨*parameters*⟩] {⟨*label*⟩}

```
126 \DeclareRobustCommand{\Submit}[2][]{}
```

\Reset [⟨*parameters*⟩] {⟨*label*⟩}

```
127 \DeclareRobustCommand{\Reset}[2][]{}
```

\Gauge [⟨*parameters*⟩] {⟨*label*⟩}

```
128 \DeclareRobustCommand{\Gauge}[2][]{}
```

\LayoutTextField {⟨*label*⟩} {⟨*field*⟩}

```
129 \newcommand*{\LayoutTextField}[2]{}
```

\LayoutChoiceField {⟨*label*⟩} {⟨*field*⟩}

```
130 \newcommand*{\LayoutChoiceField}[2]{}
```

\LayoutCheckField {⟨*label*⟩} {⟨*field*⟩}

```
131 \newcommand*{\LayoutCheckField}[2]{}
```

\MakeRadioField {⟨*width*⟩} {⟨*height*⟩}

```
132 \newcommand*{\MakeRadioField}[2]{}
```

\MakeCheckField {⟨*width*⟩} {⟨*height*⟩}

```
133 \newcommand*{\MakeCheckField}[2]{}
```

\MakeTextField {⟨*width*⟩} {⟨*height*⟩}

```
134 \newcommand*{\MakeTextField}[2]{}
```

\MakeChoiceField {⟨*width*⟩} {⟨*height*⟩}

```
135 \newcommand*{\MakeChoiceField}[2]{}
```

\MakeFieldButton {⟨*text*⟩}

```
136 \newcommand{\MakeFieldButton}[1]{}
```

File 185 **lwarp-hyperxmp.sty**

§ 285 Package hyperxmp

Pkg hyperxmp hyperxmp is ignored.

for HTML output: Discard all options for lwarp-hyperxmp:

```
\LWR@ProvidesPackageDrop{hyperxmp}[2018/11/27]
```

File 186 **lwarp-hyphenat.sty**

§ 286 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 187 **lwarp-idxlayout.sty**

§ 287 Package idxlayout

(*Emulates or patches code by* Thomas Titz.)

Pkg idxlayout idxlayout is 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 188 **lwarp-ifoddpage.sty**

§ 288 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
```

```
5
6 \DeclareRobustCommand{\checkoddpage}{\oddpagetrue\oddpageoronesidetrue}
7
8 \def\oddpage@page{1}
9
10 \def\@ifoddpage{%
11         \expandafter\@firstoftwo
12 }
13
14 \def\@ifoddpageoroneside{%
15         \expandafter\@firstoftwo
16 }
```

File 189 **lwarp-imakeidx.sty**

§ 289 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 8.6.18 for how to setup *lwarpmk* to process the indexes with imakeidx, both with and without shell escape.

for HTML output:

```
1 \LWR@ProvidesPackagePass{imakeidx}[2016/10/15]
```

Use the new HTML suffix:

```
2 \catcode`\_=12%
3 \define@key{imki}{name}{\def\imki@name{#1_html}}
4 \catcode`\_=8%
```

\printindex The HTML version of `\printindex`:

```
5 \catcode`\_=12%
6
7 \renewcommand*{\printindex}[1][\imki@jobname]{%
8 \LWR@orignewpage%
9 \LWR@startpars%
10 \ifstrequal{#1}{\imki@jobname}{%
11   \@ifundefined{#1@idxfile}{%
12         \imki@error{#1}%
13     }{%
14         \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{lwarp-imakeidx}{Undefined index file `#1'}%
            \begingroup
            \@sanitize
            \imki@nowrindex%
        }%
        {%
            \edef\@idxfile{#1}%
            \begingroup
            \@sanitize
            \@wrindex\@idxfile%
        }%
    }%
    {%
        \@ifundefined{#1_html@idxfile}%
        {%
            \PackageWarning{lwarp-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 190 **lwarp-index.sty**

§ 290 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 191 **lwarp-inputtrc.sty**

§ 291 Package inputtrc

(*Emulates or patches code by* Uwe Lück.)

Pkg inputtrc inputtrc is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{inputttrc}[2012/10/10]
```

Patched to remove extraneous spaces, which sometimes showed up in logos inside a `lateximage`.

```
\renewcommand*{\IT@prim@input}[1]{%
  \typeout{\IT@indent\IT@currfile INPUTTING #1}%
%% ... TODO: option to write to ‘.log’ only.
  \xdef\IT@filestack{{\IT@currfile}\IT@filestack}%
  \xdef\IT@currfile{#1}%
  \expandafter \gdef\expandafter \IT@indent\expandafter{%
    \IT@indent \IT@indent@unit}%                lwarp
  \@@input#1%                                   lwarp
  \expandafter\IT@pop@indent\IT@indent   \@nil% lwarp
  \expandafter\IT@pop@file  \IT@filestack\@nil% lwarp
  \IT@maybe@returnmessage%% v0.2                lwarp
}
```

File 192 **lwarp-intopdf.sty**

§ 292 Package intopdf

Pkg intopdf intopdf is emulated.

The filespec, MIME type, and description are ignored for now.

for HTML output:

```
\LWR@ProvidesPackageDrop{intopdf}[2019/05/28]
```

```
\NewDocumentCommand{\attachandlink}{o m o m m}{%
    \href{#2}{#5}%
}
```

File 193 **lwarp-karnaugh-map.sty**

§ 293 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} \& |(00001
          01 \& |(000100)|  \phantom{0} \& |(000101)|  \phantom{0} \& |(000111)|  \phantom{0} \& |(00011
          11 \& |(001100)|  \phantom{0} \& |(001101)|  \phantom{0} \& |(001111)|  \phantom{0} \& |(00111
          10 \& |(001000)|  \phantom{0} \& |(001001)|  \phantom{0} \& |(001011)|  \phantom{0} \& |(00101
     \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{lwarp-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);
      \draw[color=black, ultra thin] (0,-5) grid (4,-1);
      \draw[color=black, ultra thin] (5,-5) grid (9,-1);
    \fi
    % labels
    % for all dimensions
  \node[above] at (\@karnaughmap@var@mapsizex@*0.5,\@karnaughmap@var@mapsizey@+0.9) {\small{#5}};
    \node[left] at (-0.9,\@karnaughmap@var@mapsizey@*0.5) {\small{#6}};
```

```
      % when there are 2 sub maps
      \ifnum\@karnaughmap@var@mapsizez@=2
        \node[above] at (7,4.9) {\small{#5}};
        % extra sub maps labels
        \node[below] at (2,-0.1) {\small{#7$=0$}};
        \node[below] at (7,-0.1) {\small{#7$=1$}};
      \fi
      % when there are 4 sub maps
      \ifnum\@karnaughmap@var@mapsizez@=4
        \node[above] at (7,4.9) {\small{#5}};
        \node[left] at (-0.9,-3) {\small{#6}};
        % extra sub maps labels
        \node[below] at (2,-0.1) {\small{#7$=00$}};
        \node[below] at (7,-0.1) {\small{#7$=01$}};
        \node[below] at (2,-5.1) {\small{#7$=10$}};
        \node[below] at (7,-5.1) {\small{#7$=11$}};
      \fi
      % data
      \matrix[
        matrix of nodes,
        ampersand replacement=\&,
        column sep={1cm,between origins},
        row sep={1cm,between origins},
    ] at (\@karnaughmap@var@mapsizex@*0.5+\@karnaughmap@local@maprealignmentx@,\@karnaughmap@var@mapsiz
        \@karnaughmap@local@matrixtemplate@%
      };
}{
    \end{tikzpicture}
  \endgroup
}
```

File 194 **lwarp-keyfloat.sty**

§ 294 Package keyfloat

(*Emulates or patches code by* Brian Dunn.)

Pkg keyfloat keyfloat is supported with a considerable amount of hacking. (It's a mashup of lwarp, keyfloat, and tocdata.)

⚠ keywrap If placing a `\keyfig[H]` inside a `keywrap`, use an absolute width for `\keyfig`, 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:

```
\LWR@ProvidesPackagePass{keyfloat}[2019/09/23]

\@ifpackagelater{keyfloat}{2019/09/23}{}{
    \PackageError{lwarp-keyfloat}
    {%
        The keyfloat package is out of date.\MessageBreak
        Update to keyfloat v2.01 2019/09/23 or later%
    }
    {%
```

```
        Please update the keyfloat package.  It's worth it!%
    }
}
```

After keyfloat has loaded:

```
\AtBeginDocument{

\providecommand*{\KFLT@LWR@hook@boxouter}{}
\renewcommand*{\KFLT@LWR@hook@boxouter}{%
    \ifbool{KFLT@keywrap}{%
    }{%
        \ifnumequal{\value{KFLT@keyfloatdepth}}{0}{%
            \setlength{\linewidth}{6in}%
            \setlength{\textwidth}{6in}%
            \setlength{\textheight}{9in}%
        }{}%
    }%
    \normalcolor%
}

\LetLtxMacro\KFLT@LWR@hook@keysubfloats\KFLT@LWR@hook@boxouter

\let\KFLT@LWR@hook@keyfloatsminipage\relax
\let\endKFLT@LWR@hook@keyfloatsminipage\relax
\newenvironment*{KFLT@LWR@hook@keyfloatsminipage}[1]{}{}

\LetLtxMacro\KFLT@LWR@hook@keyfloats\KFLT@LWR@hook@boxouter

\renewcommand*{\KFLT@maybeendfloatrow}{%
    \ifnumless{\value{KFLT@thiscol}}{\value{KFLT@numcols}}%
    {}% thiscol < numcols
    {% >=
        \defcounter{KFLT@thiscol}{0}%
    }%
}%

\renewcommand{\KFLT@trackrows}%
{%
```

If are nested inside a keyfloats or a subfloat:

```
    \ifboolexpr{%
        test {\ifnumgreater{\value{KFLT@keyfloatdepth}}{0}} or%
        bool{KFLT@inkeysubfloats}%
    }%
    {% nested
```

Tracks row start and end:

```
        \KFLT@maybestartfloatrow%
```

Possibly fill space between columns:

```
        \ifnumgreater{\value{KFLT@thiscol}}{1}%
            {%
%                 \hfill%
            }%
            {}%
    }% nested
    {}% not nested
}


\RenewDocumentCommand{\KFLT@onefigureimage}{m}
{%
\LWR@traceinfo{KFLT@onefigureimage}%
% \begin{lrbox}{\KFLT@envbox}%
\ifthenelse{\NOT\equal{\KFLT@lw}{}}%
    {%
        \KFLT@frame{\includegraphics%
        [scale=\KFLT@s,width=\KFLT@imagewidth]{#1}}%
    }%
    {% not linewidth
        \ifthenelse{\dimtest{\KFLT@w}{>}{0pt}}%
        {% width is given
            \ifthenelse{\dimtest{\KFLT@h}{>}{0pt}}%
            {% w and h
                \KFLT@frame{\includegraphics%
                [scale=\KFLT@s,%
                width=\KFLT@imagewidth,height=\KFLT@h]{#1}}%
            }% w and h
            {% only w
                \KFLT@frame{\includegraphics%
                [scale=\KFLT@s,width=\KFLT@imagewidth]{#1}}%
            }% only w
        }% width is given
        {% width is not given
            \ifthenelse{\dimtest{\KFLT@h}{>}{0pt}}%
            {%
                \KFLT@frame{\includegraphics%
                [scale=\KFLT@s,height=\KFLT@h]{#1}}%
            }%
            {%
                \KFLT@frame{\includegraphics%
                [scale=\KFLT@s]{#1}}%
            }%
        }% 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%
    \minipagefullwidth%
    \ifboolexpr{bool{KFLT@ft} or bool{KFLT@f}}{%
        \fminipage{\KFLT@imagewidth}%
    }{%
        \minipage{\KFLT@imagewidth}%
    }%
}
{%
    \ifboolexpr{bool{KFLT@ft} or bool{KFLT@f}}{%
        \endfminipage%
    }{%
        \endminipage%
    }%
    \LWR@startpars%
    \LWR@traceinfo{KFLT@boxinner: done}%
}

\newcommand*{\LWR@KFLT@settextalign}[1]{%
    \def\LWR@KFLT@textalign{justify}%
    \ifcsstring{KFLT@#1textalign}{\centering}%
        {\def\LWR@KFLT@textalign{center}}%
        {}%
    \ifcsstring{KFLT@#1textalign}{\raggedleft}%
        {\def\LWR@KFLT@textalign{right}}%
        {}%
    \ifcsstring{KFLT@#1textalign}{\raggedright}%
        {\def\LWR@KFLT@textalign{left}}%
        {}%
}

\renewcommand{\KFLT@addtext}[1]
{%
```

Is there text to add?

```
    \ifcsempty{KFLT@#1t}%
    {}% no text
    {% text to add
        {% local
```

Add some space, then create a to contain the text:

```
        \addvspace{\smallskipamount}%
        \LWR@KFLT@settextalign{#1}%
        \begin{BlockClass}[text-align:\LWR@KFLT@textalign]{floatnotes}%
```

Set the alignment and some text parameters:

```
%         \csuse{KFLT@#1textalign}%
%         \footnotesize%
```

```
        \setlength{\parskip}{1.5ex}%
        \setlength{\parindent}{0em}%
```

Typeset the actual text:

```
        \csuse{KFLT@#1t}%
```

Close it all out with a little more space:

```
        \end{BlockClass}%
%         \par\addvspace{2ex}%
        }% local
    }% text to add
}

\@ifpackageloaded{tocdata}
{}
{% tocdata not loaded

    \newcommand*{\LWR@KFLT@setnamealign}[1]{%
        \def\LWR@KFLT@textalign{justify}%
        \ifstrequal{#1}{\centering}%
            {\def\LWR@KFLT@textalign{center}}%
            {}%
        \ifstrequal{#1}{\raggedleft}%
            {\def\LWR@KFLT@textalign{right}}%
            {}%
        \ifstrequal{#1}{\raggedright}%
            {\def\LWR@KFLT@textalign{left}}%
            {}%
    }

    \renewcommand*{\KFLT@@addartisttext}[3]{%

```

Add space and create the name inside a :

```
%         \addvspace{\medskipamount}%
    %     \begin{minipage}{\linewidth}%
        \LWR@KFLT@setnamealign{#3}%
        \begin{BlockClass}[text-align:\LWR@KFLT@textalign]{floatnotes}%

```

Text alignment is #3, and depends on artist or author:

```
    %     #3%

```

#1 is empty or 'subgrp'
#2 is empty for artist, 'u' for author:

```
        \footnotesize\textsc{%
            \KFLT@optionalname{\csuse{KFLT@#1a#2p}}%
            \KFLT@optionalname{\csuse{KFLT@#1a#2f}}%
```

```
            \csuse{KFLT@#1a#2l}%
            \csuse{KFLT@#1a#2s}%
        }%
    %     \end{minipage}%
        \end{BlockClass}
%         \par\addvspace{2ex}%
    }

}% tocdata not loaded
```

Env KFLT@marginfloat [⟨*offset*⟩] {⟨*type*⟩}

```
\DeclareDocumentEnvironment{KFLT@marginfloat}{O{-1.2ex} m}
{%
    \uselengthunit{PT}%
    \LWR@BlockClassWP%
        {float:right; width:2in; margin:10pt}%
        {}%
        {marginblock}%
    \captionsetup{type=#2}%
    \minipage{\LWR@usersmarginparwidth}%
    \setlength{\marginparwidth}{.95\LWR@usersmarginparwidth}%
}
{%
    \endminipage%
    \endLWR@BlockClassWP%
}

\DeclareDocumentEnvironment{marginfigure}{o}
    {\begin{KFLT@marginfloat}{figure}}
    {\end{KFLT@marginfloat}}

\DeclareDocumentEnvironment{margintable}{o}
    {\begin{KFLT@marginfloat}{table}}
    {\end{KFLT@marginfloat}}
```

Env keywrap {⟨*width*⟩} {⟨*keyfloat*⟩}

```
\DeclareDocumentEnvironment{keywrap}{m +m}
{%
    \begin{LWR@setvirtualpage}*
    \setlength{\LWR@templengthone}{#1}%
    \begin{LWR@BlockClassWP}%
        {%
         float:right; width:\LWR@printlength{\LWR@templengthone};  % extra space
            margin:10pt%
        }%
        {}%
        {marginblock}%
    \setlength{\linewidth}{.95\LWR@templengthone}%
    \booltrue{KFLT@keywrap}%
    #2%
    \end{LWR@BlockClassWP}%
    \end{LWR@setvirtualpage}%
```

```
}
{}

}% AtBeginDocument
```

File 195 **lwarp-layaureo.sty**

§ 295 Package layaureo

Pkg layaureo layaureo is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{layaureo}[2004/09/16]
```

File 196 **lwarp-layout.sty**

§ 296 Package layout

Pkg layout layout is ignored.

for HTML output: Discard all options for lwarp-layout:

```
\LWR@ProvidesPackageDrop{layout}[2014/10/28]

\NewDocumentCommand{\layout}{s}{}
```

File 197 **lwarp-layouts.sty**

§ 297 Package layouts

Pkg layouts layouts is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{layouts}[2009/09/02]

\newif\ifoddpagelayout
  \oddpagelayouttrue
\newif\iftwocolumnlayout
  \twocolumnlayoutfalse
\newif\ifdrawmarginpars
  \drawmarginparstrue
\newif\ifdrawparameters
  \drawparameterstrue
\newif\iflistaspara
  \listasparatrue
\newif\ifruninhead
  \runinheadfalse
\newif\ifprintparameters
```

```
  \printparameterstrue
\newif\ifdrawdimensions
  \drawdimensionsfalse
\newif\ifprintheadings
  \printheadingstrue
\newcommand{\testdrawdimensions}{}
\newcommand{\testprintparameters}{}
\newcommand{\setlabelfont}[1]{}
\newcommand{\setparametertextfont}[1]{}
\newcommand{\setvaluestextsize}[1]{}
\newcommand{\setlayoutscale}[1]{}
\newcommand{\setuplayouts}{}
\newcommand{\printinunitsof}[1]{}
\newcommand{\prntlen}[1]{}
\newcommand{\trypaperwidth}[1]{}
\newcommand{\trypaperheight}[1]{}
\newcommand{\tryhoffset}[1]{}
\newcommand{\tryvoffset}[1]{}
\newcommand{\trytopmargin}[1]{}
\newcommand{\tryheadheight}[1]{}
\newcommand{\tryheadsep}[1]{}
\newcommand{\trytextheight}[1]{}
\newcommand{\tryfootskip}[1]{}
\newcommand{\tryoddsidemargin}[1]{}
\newcommand{\tryevensidemargin}[1]{}
\newcommand{\trytextwidth}[1]{}
\newcommand{\trymarginparsep}[1]{}
\newcommand{\trymarginparwidth}[1]{}
\newcommand{\trymarginparpush}[1]{}
\newcommand{\trycolumnsep}[1]{}
\newcommand{\trycolumnseprule}[1]{}
\newcommand{\setfootbox}[2]{}
\newcommand{\currentpage}{}
\newcommand{\drawpage}{(draw page)}
\newcommand{\pagediagram}{(page diagram)}
\newcommand{\pagedesign}{(page design)}
\newcommand{\pagevalues}{(page values)}
\newcommand{\trystockwidth}[1]{}
\newcommand{\trystockheight}[1]{}
\newcommand{\trytrimedge}[1]{}
\newcommand{\trytrimtop}[1]{}
\newcommand{\tryuppermargin}[1]{}
\newcommand{\tryspinemargin}[1]{}
\newcommand{\currentstock}{}
\newcommand{\drawstock}{(draw stock)}
\newcommand{\stockdiagram}{(stock diagram)}
\newcommand{\stockdesign}{(stock design)}
\newcommand{\stockvalues}{(stock values)}
\newcommand{\tryitemindent}[1]{}
\newcommand{\trylabelwidth}[1]{}
\newcommand{\trylabelsep}[1]{}
\newcommand{\tryleftmargin}[1]{}
\newcommand{\tryrightmargin}[1]{}
\newcommand{\trylistparindent}[1]{}
\newcommand{\trytopsep}[1]{}
```

```
\newcommand{\tryparskip}[1]{}
\newcommand{\trypartopsep}[1]{}
\newcommand{\tryparsep}[1]{}
\newcommand{\tryitemsep}[1]{}
\newcommand{\currentlist}{}
\newcommand{\drawlist}{(draw list)}
\newcommand{\listdiagram}{(list diagram)}
\newcommand{\listdesign}{(list design)}
\newcommand{\listvalues}{(list values)}
\newcommand{\tryfootins}[1]{}
\newcommand{\tryfootnotesep}[1]{}
\newcommand{\tryfootnotebaseline}[1]{}
\newcommand{\tryfootruleheight}[1]{}
\newcommand{\tryfootrulefrac}[1]{}
\newcommand{\currentfootnote}{}
\newcommand{\drawfootnote}{(draw footnote)}
\newcommand{\footnotediagram}{(footnote diagram)}
\newcommand{\footnotedesign}{(footnote design)}
\newcommand{\footnotevalues}{(footnote values)}
\newcommand{\tryparindent}[1]{}
\newcommand{\tryparlinewidth}[1]{}
\newcommand{\tryparbaselineskip}[1]{}
\newcommand{\currentparagraph}{}
\newcommand{\drawparagraph}{(draw paragraph)}
\newcommand{\paragraphdiagram}{(paragraph diagram)}
\newcommand{\paragraphdesign}{(paragraph design)}
\newcommand{\paragraphvalues}{(paragraph values)}
\newcommand{\trybeforeskip}[1]{}
\newcommand{\tryafterskip}[1]{}
\newcommand{\tryindent}[1]{}
\newcommand{\currentheading}{}
\newcommand{\drawheading}[1]{(draw heading)}
\newcommand{\headingdiagram}[1]{(heading diagram)}
\newcommand{\headingdesign}[1]{(heading design)}
\newcommand{\headingvalues}{(heading values)}
\newcommand{\trytextfloatsep}[1]{}
\newcommand{\tryfloatsep}[1]{}
\newcommand{\tryintextsep}[1]{}
\newcommand{\trytopfigrule}[1]{}
\newcommand{\trybotfigrule}[1]{}
\newcommand{\currentfloat}{}
\newcommand{\drawfloat}{(draw float)}
\newcommand{\floatdiagram}{(float diagram)}
\newcommand{\floatdesign}{(float design)}
\newcommand{\floatvalues}{(float values)}
\newcommand{\trytotalnumber}[1]{}
\newcommand{\trytopnumber}[1]{}
\newcommand{\trybottomnumber}[1]{}
\newcommand{\trytopfraction}[1]{}
\newcommand{\trytextfraction}[1]{}
\newcommand{\trybottomfraction}[1]{}
\newcommand{\currentfloatpage}{}
\newcommand{\drawfloatpage}{(draw floatpage)}
\newcommand{\floatpagediagram}{(floatpage diagram)}
\newcommand{\floatpagedesign}{(floatpage design)}
```

```
\newcommand{\floatpagevalues}{(floatpage values)}
\newcommand{\trytocindent}[1]{}
\newcommand{\trytocnumwidth}[1]{}
\newcommand{\trytoclinewidth}[1]{}
\newcommand{\trytocrmarg}[1]{}
\newcommand{\trytocpnumwidth}[1]{}
\newcommand{\trytocdotsep}[1]{}
\newcommand{\currenttoc}{}
\newcommand{\drawtoc}{(draw toc)}
\newcommand{\tocdiagram}{(toc diagram)}
\newcommand{\tocdesign}{(toc design)}
\newcommand{\tocvalues}{(toc values)}
\newcommand{\drawaspread}[8][0]{(a spread)}
\newcommand{\drawfontframe}[1]{(font frame)}
\newcommand{\drawfontframelabel}[1]{}
```

File 198 **lwarp-leading.sty**

§ 298 Package leading

Pkg leading leading is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{leading}[2008/12/11]
\newcommand\leading[1]{}
```

File 199 **lwarp-letterspace.sty**

§ 299 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 200 **lwarp-lettrine.sty**

§ 300 Package lettrine

(*Emulates or patches code by* Daniel Flipo.)

Pkg lettrine lettrine is 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 201 **lwarp-lineno.sty**

§ 301 Package lineno

(*Emulates or patches code by* Stephan I. Böttcher.)

Pkg lineno lineno is partly emulated, but mostly ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{lineno}[2005/11/02]

\newcommand*\resetlinenumber[1][\@ne]{}

\def\linenumbers{%
     \@ifnextchar[{\resetlinenumber}%]
                  {\@ifstar{\resetlinenumber}{}}%
     }
```

```
8
9 \newcommand*{\nolinenumbers}{}
10
11 \@namedef{linenumbers*}{\par\linenumbers*}
12 \@namedef{runninglinenumbers*}{\par\runninglinenumbers*}
13
14 \def\endlinenumbers{\par}
15 \let\endrunninglinenumbers\endlinenumbers
16 \let\endpagewiselinenumbers\endlinenumbers
17 \expandafter\let\csname endlinenumbers*\endcsname\endlinenumbers
18 \expandafter\let\csname endrunninglinenumbers*\endcsname\endlinenumbers
19 \let\endnolinenumbers\endlinenumbers
20
21 \def\pagewiselinenumbers{\linenumbers\setpagewiselinenumbers}
22
23 \def\runninglinenumbers{\setrunninglinenumbers\linenumbers}
24
25 \def\setpagewiselinenumbers{}
26
27 \def\setrunninglinenumbers{}
28
29 \def\linenomath{}%
30 \@namedef{linenomath*}{}%
31 \def\endlinenomath{}
32 \expandafter\let\csname endlinenomath*\endcsname\endlinenomath
33
34 \let\linelabel\label
35
36 \def\switchlinenumbers{\@ifstar{}{}}
37 \def\setmakelinenumbers#1{\@ifstar{}{}}
38
39 \def\leftlinenumbers{\@ifstar{}{}}
40 \def\rightlinenumbers{\@ifstar{}{}}
41
42 \newcounter{linenumber}
43 \newcount\c@pagewiselinenumber
44 \let\c@runninglinenumber\c@linenumber
45
46 \def\runningpagewiselinenumbers{}
47 \def\realpagewiselinenumbers{}
48
49
50 \NewDocumentCommand\modulolinenumbers{s o}{}
51
52 \chardef\c@linenumbermodulo=5
53 \modulolinenumbers[1]
54
55 \newcommand*\firstlinenumber[1]{}
56
57 \newcommand\internallinenumbers{}
58 \let\endinternallinenumbers\endlinenumbers
59 \@namedef{internallinenumbers*}{\internallinenumbers*}
60 \expandafter\let\csname endinternallinenumbers*\endcsname\endlinenumbers
61
62 \newcommand*{\linenoplaceholder}[1]{% redefine per language
```

```
    (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 202 **lwarp-lips.sty**

§ 302 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 203 **lwarp-listings.sty**

§ 303 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}[2018/09/02]
```

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:

```
                    \LWR@stoppars%
                   \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{lwarp-listings}{Numbers #1 unknown}\@ehc}}

\end{warpHTML}
```

File 204 **lwarp-listliketab.sty**

§ 304 Package listliketab

Pkg listliketab listliketab is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{listliketab}[2005/01/09]

\newcommand*{\storestyleof}[1]{}
\newcommand*{\storeliststyle}{}
\newenvironment{listliketab}{}{}
```

File 205 **lwarp-lltjext.sty**

§ 305 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 206 **lwarp-longtable.sty**

§ 306 Package

longtable

(*Emulates or patches code by* David Carlisle.)

Pkg longtable longtable is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{longtable}[2014/10/28]
```

Use one of either \endhead or \endfirsthead for both print and HTML, and use a \warpprintonly macro to disable the other head phrase, and also the \endfoot and \endfirstfoot phrases. (See section 8.10.4 if using threeparttablex.)

```
\begin{longtable}{ [column specifiers] }
[ . . . ] \endfirsthead   % or \endhead, for print and HTML
\warpprintonly{          % not used in HTML
  [ . . . ] \endhead     % or \endfirsthead
  [ . . . ] \endfoot
  [ <lastfoot macros> ] \endlastfoot
}
. . .  table contents . . .
\warpHTMLonly{
  [ <lastfoot macros> ]   % HTML last footer, without \endfoot
                                                % or \endlastfoot.
}
\end{longtable}
```

⚠ Misplaced \noalign Use the \warpprintonly macro instead of the warpprint environment. Doing so helps avoid "Misplaced \noalign." when using \begin{warpprint}.

⚠ \kill \kill is ignored, place a \kill line inside

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

or place it inside \warpprintonly.

⚠ lateximage longtable is not supported inside a lateximage.

See:
http://tex.stackexchange.com/questions/43006/why-is-input-not-expandable

Used to detect more than one of \endhead and \endfirsthead in use for HTML at the same time.

```
\newbool{LWR@longtable@havehead}
\boolfalse{LWR@longtable@havehead}
```

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 ignored.

```
\newenvironment{longtable*}[2][]{%
    \LWR@floatbegin{table}%
    \setcaptiontype{\LTcaptype}%
    \caption@setoptions{longtable}%
    \caption@setoptions{@longtable}%
    \caption@LT@setup%
    \booltrue{LWR@starredlongtable}%
    \boolfalse{LWR@longtable@havehead}%
    \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}%
    \boolfalse{LWR@longtable@havehead}%
    \let\captionlistentry\LWR@LTcaptionlistentry%
    \tabular{#2}%
}
{\endtabular\LWR@floatend}
```

Provided for compatibility, but ignored:

```
\newcounter{LTchunksize}
```

Error for heads which should have been in `\warpprintonly`:

```
\newcommand*{\LWR@longtable@headerror}{%
    \PackageError{lwarp}
    {For longtable:\MessageBreak
    1: Keep either one of an \protect\endhead\space or
        \protect\endfirsthead\space\MessageBreak
        \space phrase as-is, to be used by both print and HTML.\MessageBreak
    2: Place any other \protect\end... phrases inside a\MessageBreak
        \space\protect\warpprintonly\space macro,
            to be ignored by HTML.\MessageBreak
    3: Add a final footer for HTML at the end of the table\MessageBreak
        \space inside a \protect\warpHTMLonly\space macro.
            This can be\MessageBreak
        \space a copy of an \protect\endfoot\space or
            \protect\endfirstfoot\space phrase,\MessageBreak
        \space but without the actual \protect\endfoot\space
            or \protect\endfirstfoot\MessageBreak
        \space macros.  If using threeparttablex, add\MessageBreak
        \space \protect\insertTableNotes\space here,
            optionally with\MessageBreak
        \space \protect\UseMinipageWidths\space in front.\MessageBreak
    See the Lwarp documentation regarding longtables\MessageBreak
    and threeparttablex}
    {See the Lwarp documentation regading longtables and threeparttablex.}
}
```

Error if more than one of `\endhead` or `\endfirsthead` is outside of `warpprintonly`.

```
\newcommand*{\LWR@longtable@maybeheaderror}{%
\ifbool{LWR@longtable@havehead}%
    {\LWR@longtable@headerror}%
    {%
        \booltrue{LWR@longtable@havehead}
        \LWR@tabularendofline% throws away options //[dim] and //*
    }%
}
```

Error if more than one of these is outside of warppprint.

```
\def\endhead{\LWR@longtable@maybeheaderror}
\def\endfirsthead{\LWR@longtable@maybeheaderror}
```

Error if ANY of these is outside of warppprint.

```
\def\endfoot{\LWR@longtable@headerror}
\def\endlastfoot{\LWR@longtable@headerror}
```

```
\providecommand*{\LWR@HTML@tabularnewline}{\LWR@tabularendofline}
\LWR@formatted{tabularnewline}
```

```
\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 207 **lwarp-lscape.sty**

§ 307 Package lscape

(*Emulates or patches code by* D. P. Carlisle.)

Pkg lscape lscape is ignored.

for HTML output: Discard all options for lwarp-lscape.

```
\LWR@ProvidesPackageDrop{lscape}[2000/10/22]
```

```
\newenvironment*{landscape}{}{}
```

File 208 **lwarp-ltablex.sty**

§ 308 Package ltablex

(*Emulates or patches code by* Anil K. Goel.)

Pkg ltablex ltablex is emulated by lwarp.

for HTML output:

Relies on tabularx.

```
\RequirePackage{longtable}
\RequirePackage{tabularx}

\LWR@ProvidesPackageDrop{ltablex}[2014/08/13]

\DeclareDocumentEnvironment{tabularx}{m o m}
{\longtable{#3}}
{\endlongtable}

\DeclareDocumentEnvironment{tabularx*}{m o m}
{\longtable{#3}}
{\endlongtable}

\newcommand*{\keepXColumns}{}
\newcommand*{\convertXColumns}{}
```

File 209 **lwarp-ltcaption.sty**

§ 309 Package ltcaption

(*Emulates or patches code by* Axel Sommerfeldt.)

Pkg ltcaption ltcaption is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{ltcaption}[2018/08/26]
```

\LTcaptype is already defined by lwarp.

longtable* is already defined by lwarp-longtable.

```
\newlength{\LTcapskip}
\newlength{\LTcapleft}
\newlength{\LTcapright}
\newcommand*{\LTcapmarginsfalse}{}
```

File 210 **lwarp-ltxgrid.sty**

§ 310 Package ltxgrid

Pkg ltxgrid ltxgrid is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{ltxgrid}[2010/07/25]

\newcommand*{\onecolumngrid}{}
\newcommand*{\twocolumngrid}{}
\newcommand*{\removestuff}{}
\newcommand*{\addstuff}[2]{}
\newcommand*{\replacestuff}[2]{}
```

File 211 **lwarp-ltxtable.sty**

§ 311 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:
```
\RequirePackage{tabularx,longtable}
\LWR@ProvidesPackageDrop{ltxtable}[1995/12/11]
```

\LTXtable {⟨*width*⟩} {⟨*file*⟩}
```
\newcommand*{\LTXtable}[2]{%
    \input{#2}%
}
```

File 212 **lwarp-lua-check-hyphen.sty**

§ 312 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 213 **lwarp-lua-visual-debug.sty**

§ 313 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 214 **lwarp-luacolor.sty**

§ 314 Package
# luacolor

Pkg luacolor luacolor is ignored.

for HTML output:
```
\LWR@ProvidesPackageDrop{luacolor}[2016/05/16]
```

```
\newcommand{\luacolorProcessBox}[1]{}
```

File 215 **lwarp-luatodonotes.sty**

§ 315 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 216 **lwarp-lyluatex.sty**

§ 316 Package

lyluatex

(*Emulates or patches code by* Fr. Jacques Peron, Urs Liska, Br. Samuel Springuel.)

Pkg lyluatex lyluatex is patched for use by lwarp.

For the first compile, to set *lwarpmk*'s configuration, use:

```
lualatex --shell-escape <filename>
```

⚠ images After compiling the document with **lwarpmk html**, use **lwarpmk limages** to convert the Lilypond images for HTML.

The option `insert=systems` results in an image per system. Each music image "system" CSS is placed inside a `<span>` of class `lyluatex`, which defaults to `display: inline-block`.

⚠ insert=fullpage The option `insert=fullpage` results in a single image per page of printed output. CSS Each music "fullpage" image is placed inside a `<div>` of class `lyluatex`. To match the number of measures per line with the printed version, use the geometry package to select the page geometry, or use the lyluatex options for page and staff sizes.

⚠ options To use `\linewidth` or `\textwidth` inside the package options for lyluatex, use the kvoptions-patch package first:

```
\usepackage{kvoptions-patch}
\usepackage[...,line-width-0.8\linewidth,...]{lyluatex}
```

⚠ raw-pdf If using `raw-pdf`, the resulting PDF images must be converted to SVG:

Enter ⇒ **lwarpmk pdftosvg tmp-ly/*.pdf**

for HTML output:

```
\LWR@origRequirePackage{luacode}

\LWR@ProvidesPackagePass{lyluatex}[2019/05/27]
```

User-redefinable `ALT` tag:

```
\newcommand*{\LyluatexImageAltText}{-lilypond-~\PackageDiagramAltText}
```

`\ly@compilescore` {⟨*Lilypond object*⟩}

```
\LetLtxMacro\LWR@orig@ly@compilescore\ly@compilescore

\renewcommand*{\ly@compilescore}[1]{%
```

A local group holds a number of changes:

```
    \begingroup%
```

The user's original geometry and font size are restored to match the print version. This allows for correct spacing in the musical score.

```
    \LWR@orignewpage%
    \LWR@origloadgeometry{LWR@usergeometry}%
    \LWR@print@normalsize%
```

A local group holds a redefined `\includegraphics` which is used by *lyluatex.lua* to insert the *Lilypond* score if `insert=systems` is used. This is now placed inside a `lateximage`, which itself is placed inside a of class `lyluatex`.

`\LWR@addbaselinemarker` preserves the left margins.

```
    \renewcommand{\includegraphics}[2][]{%
        \InlineClass{lyluatex}{%
            \begin{lateximage}[\LyluatexImageAltText]%
            \LWR@addbaselinemarker%
            \LWR@origincludegraphics{##2}%
            \end{lateximage}%
        }%
    }%
```

From the original:

```
    \ly@setunits%
    \directlua{
        ly_opts:set_option('currfiledir', [[\currfiledir]])
        ly_opts:set_option('twoside', '\ly@istwosided')
        #1
    }%
    \ly@resetunits%
    \ly@currentfonts%
```

The fullpage version is set inside an HTML :

```
    \directlua{
        if (ly.score.insert == 'fullpage') then
            tex.print{[[\string\begin{BlockClass}{lyluatex}]]}
        end
    }%
```

Generate the score:

```
    \directlua{ly.score:process()}%
```

Close the :

```
    \directlua{
        if (ly.score.insert == 'fullpage') then
            tex.print{[[\string\end{BlockClass}]]}
        end
    }%
```

Move to a new page and renew the regular page geometry:

```
    \LWR@orignewpage%
    \LWR@origrestoregeometry%
```

End of the local group.

```
    \endgroup%
}
```

In HTML the following generates an error, so is removed:

```
\xpatchcmd{\endly@bufferenv}
    {\hspace{0pt}\\}
    {}
    {}
    {\LWR@patcherror{lyluatex}{endly@bufferenv}}
```

File 217 **lwarp-magaz.sty**

§ 317 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 218 **lwarp-makeidx.sty**

§ 318 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 219 **lwarp-manyfoot.sty**

§ 319 Package

manyfoot

Pkg manyfoot

manyfoot is emulated.

bigfoot, manyfoot
⚠ verbatim

Verbatim footnotes are not yet supported.

If using the bigfoot package, and possibly also manyfoot, problems may occur with counter allocation because lwarp uses many counters, and there is a difference in how counters numbered 256 and up are handled in pdfLaTeX. With bigfoot this has been known to show up as an error related to one footnote insert being forbidden inside another. Another problem showed up as a input stack error, and which of these problems occurred depended on how many counters were allocated.

As a possible solution, try creating several new counters before defining bigfoot or manyfoot footnotes, hoping to shift the problematic counter above the 256 threshold. It may instead be necessary to use XeLaTeX or LuaLaTeX instead of pdfLaTeX.

lwarp's emulation of bigfoot uses manyfoot, so some of the bigfoot enhancements are included here.

The bigfoot "default" footnote is ignored, using the lwarp version instead.

for HTML output:

```
\LWR@ProvidesPackageDrop{manyfoot}[2005/09/11]

\RequirePackage{nccfoots}

\newcommand{\extrafootnoterule}{}

\let\defaultfootnoterule\footnoterule

\newcommand*{\SelectFootnoteRule}[2][0]{}

\newcommand{\footnoterulepriority}{1}

\newcommand{\SetFootnoteHook}[1]{}
\@onlypreamble\SetFootnoteHook

\newcommand{\SplitNote}{}

\newcommand*\ExtraParaSkip[1]{}

\newcommand*{\newfootnote}[2][plain]{%
    \ifstrequal{#2}{default}{}{% not "default"
        \expandafter\newbox\csname LWR@footnote#2box\endcsname%
        \appto{\LWR@printpendingfootnotes}{%
            \LWR@@printpendingfootnotes{footnote#2}%
        }
        \long\csdef{Footnotetext#2}##1##2{%
            \NCC@makefnmark{##1}%
```

```
            \LWR@@footnotetext{##2}{LWR@footnote#2box}%
        }%
        \long\csdef{Footnotetext#2+}##1##2{%
            \NCC@makefnmark{##1}%
            \LWR@@footnotetext{##2}{LWR@footnote#2box}%
        }%
    }% not "default"
}
\@onlypreamble\newfootnote

\newcommand*{\DeclareNewFootnote}[2][plain]{%
  \@ifnextchar[%
        {\LWR@manyfoot@declare{#1}{#2}}%
        {\LWR@manyfoot@declare{#1}{#2}[arabic]}%
}

\def\LWR@manyfoot@declare#1#2[#3]{%
\ifstrequal{#2}{default}{}{% not "default"
  \newfootnote[#1]{#2}%
  \newcounter{footnote#2}%
    \newcounter{footnote#2Reset}%
    \setcounter{footnote#2Reset}{0}%
    \csdef{thefootnote#2}{%
      \expandafter\noexpand\csname @#3\endcsname%
      \expandafter\noexpand\csname c@footnote#2\endcsname%
    }%
```

For bigfoot, the footnote commands may be appended with one or two plusses or one or two minuses, which are ignored in HTML.

```
  \expandafter\NewDocumentCommand\csname footnote#2\endcsname{t{+}t{+}t{-}t{-}}{%
        \stepcounter{footnote#2}%
        \protected@xdef\@thefnmark{\csname thefootnote#2\endcsname}%
        \@footnotemark%
        \csuse{Footnotetext#2}{\@thefnmark}% absorbs the footnote contents
    }%
    \csdef{footnotemark#2}{%
        \stepcounter{footnote#2}%
        \protected@xdef\@thefnmark{\csname thefootnote#2\endcsname}%
        \@footnotemark%
    }%
  \expandafter\NewDocumentCommand\csname footnotetext#2\endcsname{t{+}t{+}t{-}t{-}}{%
        \protected@xdef\@thefnmark{\csname thefootnote#2\endcsname}%
        \csuse{Footnotetext#2}{\@thefnmark}% absorbs the footnote contents
    }%
    \csdef{Footnotemark#2}{%
      \Footnotemark%
    }%
    \csdef{Footnote#2}##1{%
      \Footnotemark{##1}%
      \csuse{Footnotetext#2}{##1}%
    }%
}% not "default"
}
\@onlypreamble\DeclareNewFootnote
```

File 220 **lwarp-marginal.sty**

§ 320 Package marginal

Pkg marginal marginal is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{marginal}

\newcommand*{\showlostmarginals}{}
\newcommand*{\enlargefreelist}{}
\newcommand*{\onesidemarginals}{}
```

File 221 **lwarp-marginfit.sty**

§ 321 Package marginfit

Pkg marginfit marginfit is ignored.

for HTML output: Discard all options for lwarp-marginfit:

```
\LWR@ProvidesPackageDrop{marginfit}[2018/06/08]
```

File 222 **lwarp-marginfix.sty**

§ 322 Package marginfix

(*Emulates or patches code by* Stephen Hicks.)

Pkg marginfix marginfix is ignored.

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 223 **lwarp-marginnote.sty**

§ 323 Package marginnote

(*Emulates or patches code by* Markus Kohm.)

Pkg marginnote marginnote is 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 224 **lwarp-marvosym.sty**

§ 324 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 225 **lwarp-mathtools.sty**

§ 325 Package

mathtools

(*Emulates or patches code by* Morten Høgholm, Lars Madsen.)

Pkg mathtools mathtools is patched for use by lwarp. Emulation macros are provided for MathJax.

⚠ equation numbering showonlyrefs is disabled, as it conflicts with cleveref, which is used by lwarp. Equation numbers may not match the print version.

⚠ italic correction mathic is not emulated for HTML.

⚠ MathJax If using MathJax:

- Starred macros are not available. Starred environments do work.
- mathtools disallowspaces does not work for MathJax. Protect brackets which are not optional arguments, such as:

```
\begin{gathered}{}
[p]=1 . . .
\end{gathered}
```

- showonlyrefs does not work in MathJax, and will result in a difference in equation numbering compared to the print version.
- Due to MathJax limitations, the following do not render well: \overbracket, \underbracket, \overbrace, \underbrace, rcases, drcases, \Aboxed, and \ArrowBetweenLines.
- For the new cases-like environments, \text must be used to set the normal roman font if desired.
- alignat in MathJax requires math mode, but in LaTeXit doesn't. It may be required to use warpHTML and warpprint to isolate a version for each mode.
- \DeclareParedDelimiter and related must be in the preamble before \begin{document}. The starred versions of each macro are not created.

for HTML output:

```
\LWR@ProvidesPackagePass{mathtools}[2018/01/08]
\RequirePackage{graphicx}
\MHInternalSyntaxOn
```

Forces showonlyrefs off because lwarp uses cleveref, which is not compatible with showonlyrefs.

```
\renewcommand*\MT_showonlyrefs_true:{%
    \PackageWarningNoLine{lwarp}
```

```
    {%
        Mathtools \space showonlyrefs \space conflicts \space
        with \space cleveref,\MessageBreak
        which \space is \space used \space by \space lwarp, \space
        so \space showonlyrefs \space is\MessageBreak
        forced \space off. \space\space
        Equation \space numbers \space may \space not \space match%
    }
    \MT_showonlyrefs_false:
}
\mathtoolsset{showonlyrefs=false}
```

Forces math italic correction off. Not patched for lwarp.

```
\renewcommand*{\MT_mathic_true:}{\MT_mathic_false:}
\mathtoolsset{mathic=false}

\MHInternalSyntaxOff
```

For MathJax:

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\mathllap}[2][]{{#1#2}}}
\CustomizeMathJax{\newcommand{\mathrlap}[2][]{{#1#2}}}
\CustomizeMathJax{\newcommand{\mathclap}[2][]{{#1#2}}}
\CustomizeMathJax{\newcommand{\mathmbox}[1]{#1}}
\CustomizeMathJax{\newcommand{\clap}[1]{#1}}
\CustomizeMathJax{\newcommand{\LWRmathmakebox}[2][]{#2}}
\CustomizeMathJax{\newcommand{\mathmakebox}[1][]{\LWRmathmakebox}}
\CustomizeMathJax{\newcommand{\cramped}[2][]{{#1#2}}}
\CustomizeMathJax{\newcommand{\crampedllap}[2][]{{#1#2}}}
\CustomizeMathJax{\newcommand{\crampedrlap}[2][]{{#1#2}}}
\CustomizeMathJax{\newcommand{\crampedclap}[2][]{{#1#2}}}
\CustomizeMathJax{\newenvironment{crampedsubarray}[1]{}{}}
\CustomizeMathJax{\newcommand{\crampedsubstack}{}}
\CustomizeMathJax{\newcommand{\smashoperator}[2][]{#2}}
\CustomizeMathJax{\newcommand{\SwapAboveDisplaySkip}{}}

\CustomizeMathJax{\require{extpfeil}}
\CustomizeMathJax{\Newextarrow\xleftrightarrow{10,10}{0x2194}}
\CustomizeMathJax{\Newextarrow\xLeftarrow{10,10}{0x21d0}}
\CustomizeMathJax{\Newextarrow\xhookleftarrow{10,10}{0x21a9}}
\CustomizeMathJax{\Newextarrow\xmapsto{10,10}{0x21a6}}
\CustomizeMathJax{\Newextarrow\xRightarrow{10,10}{0x21d2}}
\CustomizeMathJax{\Newextarrow\xLeftrightarrow{10,10}{0x21d4}}
\CustomizeMathJax{\Newextarrow\xhookrightarrow{10,10}{0x21aa}}
\CustomizeMathJax{\Newextarrow\xrightharpoondown{10,10}{0x21c1}}
\CustomizeMathJax{\Newextarrow\xleftharpoondown{10,10}{0x21bd}}
\CustomizeMathJax{\Newextarrow\xrightleftharpoons{10,10}{0x21cc}}
\CustomizeMathJax{\Newextarrow\xrightharpoonup{10,10}{0x21c0}}
\CustomizeMathJax{\Newextarrow\xleftharpoonup{10,10}{0x21bc}}
\CustomizeMathJax{\Newextarrow\xleftrightharpoons{10,10}{0x21cb}}

\CustomizeMathJax{\newcommand{\LWRdounderbracket}[1]{\underline{#1}}}
\CustomizeMathJax{\newcommand{\LWRunderbracket}[2][]{\LWRdounderbracket{#2}}}
```

```
\CustomizeMathJax{\newcommand{\underbracket}[1][]{\LWRunderbracket}}
\CustomizeMathJax{\newcommand{\LWRdooverbracket}[1]{\overline{#1}}}
\CustomizeMathJax{\newcommand{\LWRoverbracket}[2][]{\LWRdooverbracket{#2}}}
\CustomizeMathJax{\newcommand{\overbracket}[1][]{\LWRoverbracket}}

\CustomizeMathJax{\newcommand{\LaTeXunderbrace}[1]{\underbrace{#1}}}
\CustomizeMathJax{\newcommand{\LaTeXoverbrace}[1]{\overbrace{#1}}}

\CustomizeMathJax{\newenvironment{matrix*}[1][]{\begin{matrix}}{\end{matrix}}}
\CustomizeMathJax{\newenvironment{pmatrix*}[1][]{\begin{pmatrix}}{\end{pmatrix}}}
\CustomizeMathJax{\newenvironment{bmatrix*}[1][]{\begin{bmatrix}}{\end{bmatrix}}}
\CustomizeMathJax{\newenvironment{Bmatrix*}[1][]{\begin{Bmatrix}}{\end{Bmatrix}}}
\CustomizeMathJax{\newenvironment{vmatrix*}[1][]{\begin{vmatrix}}{\end{vmatrix}}}
\CustomizeMathJax{\newenvironment{Vmatrix*}[1][]{\begin{Vmatrix}}{\end{Vmatrix}}}
\CustomizeMathJax{\newenvironment{smallmatrix*}[1][]{\begin{matrix}}{\end{matrix}}}
\CustomizeMathJax{\newenvironment{psmallmatrix*}[1][]{\begin{pmatrix}}{\end{pmatrix}}}
\CustomizeMathJax{\newenvironment{bsmallmatrix*}[1][]{\begin{bmatrix}}{\end{bmatrix}}}
\CustomizeMathJax{\newenvironment{Bsmallmatrix*}[1][]{\begin{Bmatrix}}{\end{Bmatrix}}}
\CustomizeMathJax{\newenvironment{vsmallmatrix*}[1][]{\begin{vmatrix}}{\end{vmatrix}}}
\CustomizeMathJax{\newenvironment{Vsmallmatrix*}[1][]{\begin{Vmatrix}}{\end{Vmatrix}}}
\CustomizeMathJax{\newenvironment{psmallmatrix}[1][]{\begin{pmatrix}}{\end{pmatrix}}}
\CustomizeMathJax{\newenvironment{bsmallmatrix}[1][]{\begin{bmatrix}}{\end{bmatrix}}}
\CustomizeMathJax{\newenvironment{Bsmallmatrix}[1][]{\begin{Bmatrix}}{\end{Bmatrix}}}
\CustomizeMathJax{\newenvironment{vsmallmatrix}[1][]{\begin{vmatrix}}{\end{vmatrix}}}
\CustomizeMathJax{\newenvironment{Vsmallmatrix}[1][]{\begin{Vmatrix}}{\end{Vmatrix}}}


\CustomizeMathJax{\newcommand{\LWRmultlined}[1][]{\begin{multline*}}}
\CustomizeMathJax{\newenvironment{multlined}[1][]{\LWRmultlined}{\end{multline*}}}

\CustomizeMathJax{\let\LWRorigshoveleft\shoveleft}
\CustomizeMathJax{\renewcommand{\shoveleft}[1][]{\LWRorigshoveleft}}
\CustomizeMathJax{\let\LWRorigshoveright\shoveright}
\CustomizeMathJax{\renewcommand{\shoveright}[1][]{\LWRorigshoveright}}

\CustomizeMathJax{\newenvironment{dcases}{\begin{cases}}{\end{cases}}}
\CustomizeMathJax{\newenvironment{dcases*}{\begin{cases}}{\end{cases}}}
\CustomizeMathJax{\newenvironment{rcases}{\begin{cases}}{\end{cases}}}
\CustomizeMathJax{\newenvironment{rcases*}{\begin{cases}}{\end{cases}}}
\CustomizeMathJax{\newenvironment{drcases}{\begin{cases}}{\end{cases}}}
\CustomizeMathJax{\newenvironment{drcases*}{\begin{cases}}{\end{cases}}}
\CustomizeMathJax{\newenvironment{cases*}{\begin{cases}}{\end{cases}}}

\CustomizeMathJax{\newcommand{\MoveEqLeft}[1][]{}}

\CustomizeMathJax{% special parsing to handle '&' in argument
    \def\LWRAboxed#1&#2&#3!|!{\fbox{\(#1\)}&\fbox{\(#2\)}}
    \newcommand{\Aboxed}[1]{\LWRAboxed#1&&!|!}
}

\CustomizeMathJax{
    \newcommand{\ArrowBetweenLines}[1][\Updownarrow]{#1}%   no starred
}

\CustomizeMathJax{\newcommand{\shortintertext}[1]{\\ \text{#1}\notag \\}}
```

```
\CustomizeMathJax{\newcommand{\vdotswithin}[1]{\hspace{.5em}\vdots}}
\CustomizeMathJax{\newcommand{\shortvdotswithin}[1]{ & \hspace{.5em}\vdots \\}}
\CustomizeMathJax{\newcommand{\MTFlushSpaceAbove}{}}
\CustomizeMathJax{\newcommand{\MTFlushSpaceBelow}{\\}}

% The following does not create a starred version of the macro:
\LetLtxMacro\LWR@mathtools@orig@DeclarePairedDelimiter\DeclarePairedDelimiter
\renewcommand{\DeclarePairedDelimiter}[3]{
    \LWR@mathtools@orig@DeclarePairedDelimiter{#1}{#2}{#3}
    \CustomizeMathJax{\newcommand{#1}[2][]{{##1#2##2##1#3}}}
}
\@onlypreamble\DeclareParedDelimiter

% (\cs{DeclarePairedDelimiterX} is already defined to use \cs{\DeclarePairedDelimiterXPP}.)

% The following does not create a starred version of the macro.
\LetLtxMacro\LWR@mathtools@orig@DeclarePairedDelimiterXPP\DeclarePairedDelimiterXPP
\DeclareDocumentCommand{\DeclarePairedDelimiterXPP}{m O{1} m m m m m}{
    \LWR@mathtools@orig@DeclarePairedDelimiterXPP{#1}[#2]{#3}{#4}{#5}{#6}{#7}
    \appto\LWR@customizedMathJax{\LWRbackslash(}%
    \appto\LWR@customizedMathJax{%
        \LWRbackslash{}newcommand\{\LWRbackslash\macrotocsname{#1}LWRsub\}%
    }%
    \appto\LWR@customizedMathJax{[#2]}%
    \appto\LWR@customizedMathJax{\{\{\LWRbackslash{}delimsize}%
    \LWR@subcustomizedmathjax{#3#4#7}%
    \appto\LWR@customizedMathJax{\LWRbackslash{}delimsize}%
    \LWR@subcustomizedmathjax{#5#6}%
    \appto\LWR@customizedMathJax{\}\}}%
    \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
    \appto\LWR@customizedMathJax{\LWRbackslash(}%
    \appto\LWR@customizedMathJax{%
        \LWRbackslash{}newcommand\{\LWRbackslash\macrotocsname{#1}\}[1][]%
    }%
    \appto\LWR@customizedMathJax{%
        \{
        \LWRbackslash{}def\LWRbackslash{}delimsize\{\#1\}
        \LWRbackslash\macrotocsname{#1}LWRsub
        \}%
    }%
    \appto\LWR@customizedMathJax{\LWRbackslash)\par}%
}
\@onlypreamble\DeclareParedDelimiterXPP
\@onlypreamble\DeclareParedDelimiterX

\CustomizeMathJax{\newcommand\lparen{(}}
\CustomizeMathJax{\newcommand\rparen{)}}
\CustomizeMathJax{\newcommand{\vcentcolon}{:}}
\CustomizeMathJax{\newcommand{\ordinarycolon}{:}}

\CustomizeMathJax{\newcommand\dblcolon{\vcentcolon\vcentcolon}}
\CustomizeMathJax{\newcommand\coloneqq{\vcentcolon=}}
\CustomizeMathJax{\newcommand\Coloneqq{\dblcolon=}}
\CustomizeMathJax{\newcommand\coloneq{\vcentcolon{-}}}
```

```
\CustomizeMathJax{\newcommand\Coloneq{\dblcolon{-}}}

\CustomizeMathJax{\newcommand\eqqcolon{=\vcentcolon}}
\CustomizeMathJax{\newcommand\Eqqcolon{=\dblcolon}}
\CustomizeMathJax{\newcommand\eqcolon{\mathrel{-}\vcentcolon}}
\CustomizeMathJax{\newcommand\Eqcolon{\mathrel{-}\dblcolon}}

\CustomizeMathJax{\newcommand\colonapprox{\vcentcolon\approx}}
\CustomizeMathJax{\newcommand\Colonapprox{\dblcolon\approx}}
\CustomizeMathJax{\newcommand\colonsim{\vcentcolon\sim}}
\CustomizeMathJax{\newcommand\Colonsim{\dblcolon\sim}}

\CustomizeMathJax{\newcommand{\nuparrow}{\cancel{\uparrow}}}
\CustomizeMathJax{\newcommand{\ndownarrow}{\cancel{\downarrow}}}
\CustomizeMathJax{\newcommand{\bigtimes}{{\Large\times}}}

\CustomizeMathJax{\newcommand{\prescript}[3]{{}^{#1}_{#2}#3}}


\CustomizeMathJax{\newenvironment{lgathered}{\begin{gathered}}{\end{gathered}}}
\CustomizeMathJax{\newenvironment{rgathered}{\begin{gathered}}{\end{gathered}}}

\LetLtxMacro\LWR@mathtools@orig@newgathered\newgathered
\renewcommand{\newgathered}[4]{%
    \LWR@mathtools@orig@newgathered{#1}{#2}{#3}{#4}%
    \appto\LWR@customizedMathJax{\LWRbackslash(}%
    \LWR@subcustomizedmathjax{%
        \newenvironment{#1}{\begin{gathered}}{\end{gathered}}%
    }%
    \appto\LWR@customizedMathJax{\LWRbackslash)}%
}
\@onlypreamble\newgathered

\CustomizeMathJax{\newcommand{\splitfrac}[2]{{}^{#1}_{#2}}}
\CustomizeMathJax{\let\splitdfrac\splitfrac}
\end{warpMathJax}
```

File 226 **lwarp-mcaption.sty**

§ 326 Package

mcaption

(*Emulates or patches code by* Stephan Hennig.)

Pkg mcaption mcaption is ignored.

for HTML output: Discard all options for lwarp-mcaption:

```
\LWR@ProvidesPackageDrop{mcaption}[2009/03/13]

\newenvironment{margincap}{}{}
\newcommand*{\margincapalign}{}
\newlength{\margincapsep}
```

File 227 **lwarp-mdframed.sty**

§ 327 Package mdframed

(*Emulates or patches code by* Marco Daniel, Elke Schubert.)

Pkg mdframed mdframed is loaded with options forced to `framemethod=none`.

§ 327.1 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.

§ 327.2 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}
```

§ 327.3 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
```

§ 327.4 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},
}
```

§ 327.5 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}%
}
```

§ 327.6 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}%
}
```

§ 327.7 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%
}
```

§ 327.8 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
```

§ 327.9 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 228 **lwarp-media9.sty**

§ 328 Package
# media9

Pkg media9 media9 is emulated.

The packages multimedia, movie15, and media9 are supported.

HTML5 <audio> and <video> objects are created for .mp3 and .mp4 files.

HTML5 <embed> objects are created for http and ftp links.

\href links are created for other media types. (Unfortunately, there is not much overlap between the file types supported for print output and the file types supported by HTML5.)

For media9, a multimedia object is inserted for each addresource=, as well as each flashvars source= and src=. This may result in duplicate objects.

Undesired objects may be nullified by placing them inside \warpprintonly or the warpprint environment.

Each HTML multimedia object includes the poster text, except for <embed> objects. For movie15, the text option is supported to specify the poster text.

The width, height, and totalheight options are supported. The HTML object is scaled according to the display width, correctly compensating for either tall or wide viewports.

Other options are ignored.

media9 \addmediapath is supported. It is assumed that the same path structure will exist for the HTML document.

HTML5 media controls are always specified for each <audio> and <video> object.

media9 slideshows are not supported.

\hyperlinkmovie, \movieref, and \mediabutton are not supported.

3D objects are not supported.

If using a YouTube™ video, use an "embedded" URL with .../embed/... instead of .../v/...

⚠ & in a URL Many special characters are converted to regular catcode 12 characters for use inside a URL. & is used in the flash variables fields, which are split with xparse \SplitList, which does not seem to work with a catcode 12 divider token, so & is not converted to catcode 12, and will not work in a URL with media9. Using & in a URL in a flashvars field may also cause parsing problems with print output, as well.

for HTML output:
```
1 \LWR@ProvidesPackageDrop{media9}[2019/02/21]
```

```
\LWR@origRequirePackage{lwarp-common-multimedia}

\RequirePackage{xkeyval}
```

`\addmediapath` {⟨*path*⟩}

Supported.

```
\newcommand*{\LWR@medianine@path}{}

\newcommand*{\addmediapath}[1]{\appto\LWR@medianine@path{{#1}}}
```

The options and poster text are reused in several places.

```
\newcommand*{\LWR@medianine@postertext}{}
\newcommand*{\LWR@medianine@options}{}
```

Each `addresource` can generate a multimedia object.

```
\define@key{LWR@medianine}{addresource}{%
    \expandafter\LWR@multimedia\expandafter[\LWR@medianine@options]
        {\LWR@medianine@postertext}
        {#1}
}
```

Each `flashvars source` can generate a multimedia object.

```
\newcommand*{\LWR@medianine@flashvarsb}[1]{%
    \IfBeginWith{#1}{source=}{%
        \StrGobbleLeft{#1}{7}[\LWR@tempone]%
        \expandafter\LWR@multimedia\expandafter[\LWR@medianine@options]%
            {\LWR@medianine@postertext}%
            {\LWR@tempone}%
    }{}%
    \IfBeginWith{#1}{src=}{%
        \StrGobbleLeft{#1}{4}[\LWR@tempone]%
        \expandafter\LWR@multimedia\expandafter[\LWR@medianine@options]%
            {\LWR@medianine@postertext}%
            {\LWR@tempone}%
    }{}%
}

\NewDocumentCommand{\LWR@medianine@flashvars}{ >{\SplitList{&}} m }{%
    \ProcessList {#1}{\LWR@medianine@flashvarsb}%
}

\define@key{LWR@medianine}{flashvars}{%
    \LWR@medianine@flashvars{#1}%
}
```

`\includemedia` [⟨*options*⟩] {⟨*poster text*⟩} {⟨*file or URL*⟩}

```
\newcommand*{\LWR@includemediab}[3][]{%
    \let\input@path\LWR@medianine@path%
    \renewcommand*{\LWR@medianine@options}{#1}%
```

```
    \renewcommand*{\LWR@medianine@postertext}{#2}%
    \setkeys*{LWR@medianine}{#1}%
    \IfBeginWith{#3}{http}{\LWR@multimedia[#1]{#2}{#3}}{%
    \IfBeginWith{#3}{HTTP}{\LWR@multimedia[#1]{#2}{#3}}{%
    \IfBeginWith{#3}{ftp}{\LWR@multimedia[#1]{#2}{#3}}{%
    \IfBeginWith{#3}{FTP}{\LWR@multimedia[#1]{#2}{#3}}{%
    }}}}%
    \endgroup%
}

\newrobustcmd*{\includemedia}{%
    \begingroup%
    \LWR@linkmediacatcodes%
    \LWR@includemediab%
}
```

`\mediabutton` `[`⟨*options*⟩`]` `{`⟨*text*⟩`}`

Ignored.

```
\newcommand*{\mediabutton}[2][]{}
```

File 229 **lwarp-memhfixc.sty**

§ 329 Package memhfixc

Pkg `memhfixc` memhfixc is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{memhfixc}[2013/05/30]
```

File 230 **lwarp-metalogo.sty**

§ 330 Package metalogo

(*Emulates or patches code by* Andrew Gilbert Moschou.)

Pkg `metalogo` metalogo is used in print mode, and emulated in HTML.

for HTML output:

```
\LWR@ProvidesPackagePass{metalogo}[2010/05/29]

\newcommand*{\LWR@HTML@setlogokern}[2]{}
\newcommand*{\LWR@HTML@setlogodrop}[2][XeTeX]{}
\newcommand*{\LWR@HTML@setLaTeXa}[1]{}
\newcommand*{\LWR@HTML@setLaTeXee}[1]{}
\newcommand*{\LWR@HTML@seteverylogo}[1]{}
\newcommand*{\LWR@HTML@everylogo}[1]{}

\LWR@formatted{setlogokern}
\LWR@formatted{setlogodrop}
\LWR@formatted{setLaTeXa}
\LWR@formatted{setLaTeXee}
```

```
\LWR@formatted{seteverylogo}
\LWR@formatted{everylogo}
```

File 231 **lwarp-metalogox.sty**

§ 331 Package metalogox

(*Emulates or patches code by* Brian Dunn.)

Pkg metalogox metalogox is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{metalogox}[2019/01/20]
```

\AtBeginDocument, adjust the logo setting according to the font which is active at that moment.

```
\AtBeginDocument{
    \let\LWR@metalogox@currentformatting\LWR@formatting
    \renewcommand*{\LWR@formatting}{print}%
    \autoadjustlogos*
    \let\LWR@formatting\LWR@metalogox@currentformatting
}
```

File 232 **lwarp-mhchem.sty**

§ 332 Package mhchem

(*Emulates or patches code by* Martin Hensel.)

Pkg mhchem mhchem is patched for use by lwarp.

without MathJax Without MathJax, mhchem expressions are converted to svg math. Inline expressions use hashed filenames to allow reuse, and assume that any mhchem options are global.

MathJax with mhchem extension For MathJax, the mhchem extension is used if the mhchem expression is used inside a math expression:

```
$\ce{C6H5-CHO}$
```

not inside math If *not* used inside a math expression, lwarp converts standalone mhchem expressions into svg math images.

MathJax without mhchem extension If the MathJax mhchem extension is not used, expressions inside math must be placed between \displaymathother and \displaymathnormal:

```
\displaymathother
\[ \ce{ . . . } \]     . . .     $ \ce { . . . } $
\displaymathnormal
```

⚠ nested math

When producing HTML output without the MathJax mhchem extension, lwarp does not support the use of nested dollar signs in mhchem expressions.

For some examples from the mhchem manual, change as follows:

```
$\ce{NaOH(aq,$\infty$)}$            % old
$\ce{NaOH(aq,\infty)}$              % new

$\ce{Fe(CN)_{$\frac{6}{2}$}$}$      % old
$\ce{Fe(CN)_{\frac{6}{2}}}$         % new

$\ce{NO_$x$}$                       % old
$\ce{NO_x}$                         % new

$\ce{NO_${x}$}$                     % old
$\ce{NO_{x}}$                       % new

$\ce{$cis${-}[PtCl2(NH3)2]}$        % old
$\ce{\mathit{cis}{-}[PtCl2(NH3)2]}$ % new
```

for HTML output:

```
\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]{%
\LWR@findcurrenttextcolor% sets \LWR@tempcolor
\ifbool{LWR@xfakebold}%
    {\def\LWR@tempone{Y}}%
    {\def\LWR@tempone{N}}%
\begin{lateximage}*[\textbackslash{}ce\{\LWR@HTMLsanitize{#1}\}]*%
    [%
        FM\LWR@f@family%
        SR\LWR@f@series%
        SH\LWR@f@shape%
        SHC\LWR@f@shapecaps%
        CL\LWR@tempcolor%
        FB\LWR@tempone% xfakebold
    ]%
\LWR@setcurrentfont%
\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]
{%
\LWR@findcurrenttextcolor% sets \LWR@tempcolor
\ifbool{LWR@xfakebold}%
    {\def\LWR@tempone{Y}}%
    {\def\LWR@tempone{N}}%
\begin{lateximage}*[\textbackslash{}cesplit\{\LWR@HTMLsanitize{#2}\}]*%
    [%
        FM\LWR@f@family%
        SR\LWR@f@series%
        SH\LWR@f@shape%
        SHC\LWR@f@shapecaps%
        CL\LWR@tempcolor%
        FB\LWR@tempone% xfakebold
    ]%
\LWR@setcurrentfont%
\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 233 **lwarp-microtype.sty**

§ 333 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 234 **lwarp-midfloat.sty**

§ 334 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 235 **lwarp-midpage.sty**

§ 335 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 236 **lwarp-minibox.sty**

§ 336 Package minibox

(*Emulates or patches code by* Will Robertson.)

Pkg minibox minibox is patched for use by lwarp.

Due to HTML limitations regarding paragraphs and s, miniboxes inline with other text will appear on their own line.

for HTML output:

```
\LWR@ProvidesPackagePass{minibox}[2013/06/21]
```

```
\ExplSyntaxOn
\newcommand\LWR@HTML@minibox[2][]{%
    \LWR@stoppars%
```

```
    \group_begin:
    \keys_set:nn {minibox} {#1}
    \bool_if:NTF \l_minibox_frame_bool
    {
        \setlength\fboxrule{\l_minibox_rule_dim}
        \setlength\fboxsep{\l_minibox_pad_dim}
        \fboxBlock{%
            \begin{tabular}[\l_minibox_tabular_valign_tl]%
              {\l_minibox_tabular_preamble_tl}
                {#2}
            \end{tabular}
        }%
    }
    {
        \begin{BlockClass}[display:inline-block]{minibox}
        \begin{tabular}[\l_minibox_tabular_valign_tl]%
          {\l_minibox_tabular_preamble_tl}
            {#2}
        \end{tabular}
        \end{BlockClass}
    }
    \group_end:
    \LWR@startpars%
}
\ExplSyntaxOff

\LWR@formatted{minibox}
```

File 237 **lwarp-minitoc.sty**

§ 337 Package minitoc

Pkg minitoc minitoc is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{minitoc}[2018/07/12]
```

mtcoff disables minitoc.

```
\usepackage{mtcoff}
```

File 238 **lwarp-morefloats.sty**

§ 338 Package morefloats

Pkg morefloats morefloats is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{morefloats}[2015/07/22]
```

File 239 **lwarp-moreverb.sty**

§ 339 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 240 **lwarp-movie15.sty**

§ 340 Package

movie15

Pkg movie15 movie15 is emualted.

The packages multimedia, movie15, and media9 are supported.

HTML5 <audio> and <video> objects are created for .mp3 and .mp4 files.

HTML5 <embed> objects are created for http and ftp links.

\href links are created for other media types. (Unfortunately, there is not much overlap between the file types supported for print output and the file types supported by HTML5.)

For media9, a multimedia object is inserted for each addresource=, as well as each flashvars source= and src=. This may result in duplicate objects.

Undesired objects may be nullified by placing them inside \warpprintonly or the warpprint environment.

Each HTML multimedia object includes the poster text, except for <embed> objects. For movie15, the text option is supported to specify the poster text.

The width, height, and totalheight options are supported. The HTML object is scaled according to the display width, correctly compensating for either tall or wide viewports.

Other options are ignored.

media9 \addmediapath is supported. It is assumed that the same path structure will exist for the HTML document.

HTML5 media controls are always specified for each <audio> and <video> object.

media9 slideshows are not supported.

\hyperlinkmovie, \movieref, and \mediabutton are not supported.

3D objects are not supported.

If using a YouTube™ video, use an "embedded" URL with .../embed/... instead of .../v/...

for HTML output:

```
\LWR@ProvidesPackageDrop{movie15}[2012/05/16]

\LWR@origRequirePackage{lwarp-common-multimedia}

\RequirePackage{xkeyval}

\newcommand*{\LWR@moviefifteen@text}{}

\define@key{LWR@moviefifteen}{text}{\renewcommand{\LWR@moviefifteen@text}{#1}}

\newcommand*{\LWR@includemovieb}[4][]{%
    \renewcommand{\LWR@moviefifteen@text}{(multimedia)}
    \setkeys*{LWR@moviefifteen}{#1}%
    \LWR@multimediab[#1,width=#2,height=#3]{\LWR@moviefifteen@text}{#4}%
}

\newrobustcmd*{\includemovie}{%
    \begingroup%
    \LWR@linkmediacatcodes%
    \LWR@includemovieb%
}


\newcommand*{\movieref}[3][]{}

\LetLtxMacro\movie\LWR@multimedia
% \LetLtxMacro\sound\LWR@multimedia% not in media15

\newcommand{\hyperlinkmovie}[3][]{}
```

File 241 **lwarp-mparhack.sty**

§ 341 Package

mparhack

Pkg mparhack

mparhack is ignored.

for HTML output:

Discard all options for lwarp-mparhack:

```
\LWR@ProvidesPackageDrop{mparhack}[2005/04/17]
```

File 242 **lwarp-multicap.sty**

§ 342 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 243 **lwarp-multicol.sty**

§ 343 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 244 **lwarp-multicolrule.sty**

§ 344 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 245 **lwarp-multimedia.sty**

§ 345 Package

multimedia

Pkg multimedia multimedia is emulated.

The packages multimedia, movie15, and media9 are supported.

HTML5 <audio> and <video> objects are created for .mp3 and .mp4 files.

HTML5 <embed> objects are created for http and ftp links.

\href links are created for other media types. (Unfortunately, there is not much overlap between the file types supported for print output and the file types supported by HTML5.)

For media9, a multimedia object is inserted for each addresource=, as well as each flashvars source= and src=. This may result in duplicate objects.

Undesired objects may be nullified by placing them inside \warpprintonly or the warpprint environment.

Each HTML multimedia object includes the poster text, except for <embed> objects. For movie15, the text option is supported to specify the poster text.

The width, height, and totalheight options are supported. The HTML object is scaled according to the display width, correctly compensating for either tall or wide viewports.

Other options are ignored.

media9 \addmediapath is supported. It is assumed that the same path structure will exist for the HTML document.

HTML5 media controls are always specified for each <audio> and <video> object.

media9 slideshows are not supported.

\hyperlinkmovie, \movieref, and \mediabutton are not supported.

3D objects are not supported.

If using a YouTube™ video, use an "embedded" URL with .../embed/... instead of .../v/...

for HTML output:

```
\LWR@ProvidesPackageDrop{multimedia}[2012/05/02]

\LWR@origRequirePackage{lwarp-common-multimedia}

\LetLtxMacro\movie\LWR@multimedia
\LetLtxMacro\sound\LWR@multimedia

\newcommand{\hyperlinkmovie}[3][]{}
```

File 246 **lwarp-multirow.sty**

§ 346 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.

vposn

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

multirow cells

- For multirow, insert \mrowcell into any empty multi-row cells. This will be a null function for the print output, and is a placeholder for parsing the table for HTML output. An error is generated if this is missed.

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

colored cells

- The multirow documentation regarding colored cells recommends using a negative number of rows. This will not work with lwarp, so \warpprintonly and \warpHTMLonly must be used to make versions for print and HTML.

with \multicolumn

- See section 346.2 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

⚠ empty cells

As per \multirow, skipped cells to the right of the \multicolumnrow statement are not included in the source code on the same line. On the following lines, \mcolrowcell must be used for each cell of each column and each row to be skipped. An error is generated if this is missed.

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

⚠ MathJax

- MathJax does not support multirow, so it is emulated to only print its text on the first row. \multirow works as expected in text tabulars or SVG math.

In a lateximage, \LWR@restoreorigformatting restores the original print-mode versions.

See section 72.25 for the print-mode versions.

for HTML output: Remove the placeholder macro which was used if multirow was not loaded:

```
1 \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}{}
```

§ 346.1 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}%

\booltrue{LWR@usedmultirow}%

\LWR@maybenewtablerow%
\LWR@tabularleftedge%
```

Print the start of a new table data cell:

```
\LWR@htmltag{td rowspan="#2" %
```

The vertical alignment, if given:

```
\ifstrequal{#1}{c}{style="\LWR@print@mbox{vertical-align:middle}" }{}%
\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%
\boolfalse{LWR@intabularmetadata}%
\renewcommand{\LWR@multirowborder}{}%
\LWR@traceinfo{*** LWR@HTML@multirow done}%
}%

\LWR@formatted{multirow}
```

§ 346.2 Combined multicolumn and multirow

\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{

\NewExpandableDocumentCommand{\LWR@HTML@multicolumnrow}{m m O{} m O{} m O{} +m}{%

\booltrue{LWR@usedmultirow}%
```

Figure out how many extra HTML columns to add for @ and ! columns:

```
\LWR@tabularhtmlcolumns{\arabic{LWR@tableLaTeXcolindex}}{#1}
```

Create the multicolumn/multirow tag, temporarily redefining the end of line. (Using a group caused problems with a nested tabular.

```
\LetLtxMacro{\\}{\LWR@endofline}%
\LWR@domulticolumn[#3][#4]{#1}{\arabic{LWR@tabhtmlcoltotal}}{#2}{#8}%
\LetLtxMacro{\\}{\LWR@tabularendofline}%
```

Move to the next LaTeX column:

```
\defaddtocounter{LWR@tableLaTeXcolindex}{#1}%
\defaddtocounter{LWR@tableLaTeXcolindex}{-1}%
```

Skip any trailing `@` or `!` columns for this cell:

```
\booltrue{LWR@skipatbang}%
}

\LWR@expandableformatted{multicolumnrow}

}% \AtBeginDocument
```

For MathJax. Only the text is used. All other parameters are ignored.

```
\begin{warpMathJax}
% \multirow[vpos]{num}[bigstruts]{width}[vmove]{text}
\CustomizeMathJax{\newcommand{\LWRsubmultirow}[2][]{#2}}
\CustomizeMathJax{\newcommand{\LWRmultirow}[2][]{\LWRsubmultirow}}
\CustomizeMathJax{\newcommand{\multirow}[2][]{\LWRmultirow}}
%
\CustomizeMathJax{\newcommand{\mrowcell}{}}
\CustomizeMathJax{\newcommand{\mcolrowcell}{}}
\CustomizeMathJax{\newcommand{\STneed}[1]{}}
\end{warpMathJax}
```

File 247 **lwarp-multitoc.sty**

§ 347 Package multitoc

Pkg multitoc multitoc is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{multitoc}[1999/06/08]

\newcommand{\multicolumntoc}{2}
\newcommand{\multicolumnlot}{2}
\newcommand{\multicolumnlof}{2}
\newcommand*{\immediateaddtocontents}[2]{}
```

File 248 **lwarp-musicography.sty**

§ 348 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 and fingered bass, since the HTML fonts tend not to be the correct size and HTML cannot stack items. The HTML `alt` tag copies `C` and `3/2`, etc. Hashes are used for the meter images, which are then reused as necessary.

⚠ Note that browser support for musical symbols may be buggy. ALT text and copy/paste into a text editor work well.

for HTML output:

```
\LWR@ProvidesPackagePass{musicography}[2019/05/28]

\NewDocumentCommand{\LWR@HTML@musSymbol}{ O{\musFont} m m m m }{%
\begin{lateximage}%
{#1\kern#2\raisebox{#3}{#5}\kern#4}%
\end{lateximage}%
}

\LWR@formatted{musSymbol}

\NewDocumentCommand{\LWR@HTML@musStemmedNote}{ m }{%
\begin{lateximage}%
\musSymbol{0.05em}{0.5ex}{0.2em}{#1\musStem}%
\end{lateximage}%
}

\LWR@formatted{musStemmedNote}

\NewDocumentCommand{\LWR@HTML@musFlaggedNote}{ m m }{%
\begin{lateximage}%
\musSymbol{0.05em}{0.5ex}{0pt}{#1\musStem}%
\musSymbol{0pt}{0pt}{0.9em}{#2}%
\end{lateximage}%
}

\LWR@formatted{musFlaggedNote}

\NewDocumentCommand{\LWR@HTML@musDottedNote}{ m }{%
\begin{lateximage}%
#1\musDot%
\end{lateximage}%
}

\LWR@formatted{musDottedNote}

\NewDocumentCommand{\LWR@HTML@musMeter}{ m m }{%
\begin{lateximage}*[#1/#2][#1#2]*%
\musStack{#1 #2}\kern0.05em%
\end{lateximage}%
}

\LWR@formatted{musMeter}

\NewDocumentCommand{\LWR@HTML@meterCplus}{ m }{%
\begin{lateximage}*[C#1]*%
    \meterC{}\kern-0.7pt#1%
\end{lateximage}%
}

\LWR@formatted{meterCplus}

\NewDocumentCommand{\LWR@HTML@meterC}{}{%
```

```
\begin{lateximage}*[C]*%
\musSymbolMeter{\symbol{83}}%
\end{lateximage}%
}

\LWR@formatted{meterC}

\NewDocumentCommand{\LWR@HTML@meterCutC}{}{%
\begin{lateximage}*[C|]*%
\musSymbolMeter{\symbol{82}}%
\end{lateximage}%
}

\LWR@formatted{meterCutC}

\NewDocumentCommand{\LWR@HTML@meterCThreeTwo}{}{%
\begin{lateximage}*[C3/2]*%
\meterCplus{\musStack{3 2}}%
\end{lateximage}%
}

\LWR@formatted{meterCThreeTwo}

\NewDocumentCommand{\LWR@HTML@meterO}{}{\HTMLunicode{25EF}}

\LWR@formatted{meterO}

\newcommand{\LWR@null@noFig}[1][]{}%

\NewDocumentCommand{\LWR@HTML@musFig}{ m }{%
\begin{lateximage}*[%
    {% ALT text for copy/paste
        \LetLtxMacro\noFig\LWR@null@noFig%
        \LetLtxMacro\musSharp\LWR@HTML@musSharp%
        \LetLtxMacro\musDoubleSharp\LWR@HTML@musDoubleSharp%
        \LetLtxMacro\musFlat\LWR@HTML@musFlat%
        \LetLtxMacro\musDoubleFlat\LWR@HTML@musDoubleFlat%
        \LetLtxMacro\musNatural\LWR@HTML@musNatural%
        {#1}% braces here because \noFig uses []
    }%
]*%
    \musStack[\musFigFont]{#1}%
\end{lateximage}%
}

\LWR@formatted{musFig}

\NewDocumentCommand{\LWR@HTML@musFlat}       {}{\HTMLunicode{266D}}
\NewDocumentCommand{\LWR@HTML@musDoubleFlat} {}{\HTMLunicode{1D12B}}
\NewDocumentCommand{\LWR@HTML@musSharp}      {}{\HTMLunicode{266F}}
\NewDocumentCommand{\LWR@HTML@musDoubleSharp}{}{\HTMLunicode{1D12A}}
\NewDocumentCommand{\LWR@HTML@musNatural}    {}{\HTMLunicode{266E}}

\LWR@formatted{musFlat}
\LWR@formatted{musDoubleFlat}
```

```
\LWR@formatted{musSharp}
\LWR@formatted{musDoubleSharp}
\LWR@formatted{musNatural}

\NewDocumentCommand{\LWR@HTML@musWhole}         {}{\HTMLunicode{1D15D}}
\NewDocumentCommand{\LWR@HTML@musHalf}          {}{\HTMLunicode{1D15E}}
\NewDocumentCommand{\LWR@HTML@musQuarter}       {}{\HTMLunicode{1D15F}}
\NewDocumentCommand{\LWR@HTML@musEighth}        {}{\HTMLunicode{1D160}}
\NewDocumentCommand{\LWR@HTML@musSixteenth}     {}{\HTMLunicode{1D161}}
\NewDocumentCommand{\LWR@HTML@musThirtySecond}  {}{\HTMLunicode{1D162}}
\NewDocumentCommand{\LWR@HTML@musSixtyFourth}   {}{\HTMLunicode{1D163}}

\LWR@formatted{musWhole}
\LWR@formatted{musHalf}
\LWR@formatted{musQuarter}
\LWR@formatted{musEighth}
\LWR@formatted{musSixteenth}
\LWR@formatted{musThirtySecond}
\LWR@formatted{musSixtyFourth}

\NewDocumentCommand{\LWR@HTML@musWholeDotted}{}
    {\HTMLunicode{1D15D}\HTMLunicode{1D16D}}
\NewDocumentCommand{\LWR@HTML@musHalfDotted}{}
    {\HTMLunicode{1D15E}\HTMLunicode{1D16D}}
\NewDocumentCommand{\LWR@HTML@musQuarterDotted}{}
    {\HTMLunicode{1D15F}\HTMLunicode{1D16D}}
\NewDocumentCommand{\LWR@HTML@musEighthDotted}{}
    {\HTMLunicode{1D160}\HTMLunicode{1D16D}}
\NewDocumentCommand{\LWR@HTML@musSixteenthDotted}{}
    {\HTMLunicode{1D161}\HTMLunicode{1D16D}}
\NewDocumentCommand{\LWR@HTML@musThirtySecondDotted}{}
    {\HTMLunicode{1D162}\HTMLunicode{1D16D}}
\NewDocumentCommand{\LWR@HTML@musSixtyFourthDotted}{}
    {\HTMLunicode{1D163}\HTMLunicode{1D16D}}

\LWR@formatted{musWholeDotted}
\LWR@formatted{musHalfDotted}
\LWR@formatted{musQuarterDotted}
\LWR@formatted{musEighthDotted}
\LWR@formatted{musSixteenthDotted}
\LWR@formatted{musThirtySecondDotted}
\LWR@formatted{musSixtyFourthDotted}
```

File 249 **lwarp-nameauth.sty**

§ 349 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 250 **lwarp-nameref.sty**

§ 350 Package nameref

Pkg nameref nameref is emulated by lwarp.

for HTML output: Discard all options for lwarp-nameref:

```
\PackageInfo{lwarp}{%
Using the lwarp HTML version of package ‘nameref’,\MessageBreak
and discarding options.\MessageBreak
(Not using \protect\ProvidesPackage, so that other packages\MessageBreak
do not attempt to patch lwarp’s version of ‘nameref’.)\MessageBreak
}
\DeclareOption*{}
\ProcessOptions\relax
```

File 251 **lwarp-natbib.sty**

§ 351 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 252 **lwarp-nccfancyhdr.sty**

§ 352 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 253 **lwarp-needspace.sty**

§ 353 Package needspace

(*Emulates or patches code by* Peter Wilson.)

Pkg needspace needspace is ignored.

for HTML output: Discard all options for lwarp-needspace:

```
\LWR@ProvidesPackageDrop{needspace}[2010/09/12]

\DeclareDocumentCommand{\needspace}{m}{}
\DeclareDocumentCommand{\Needspace}{s m}{}
```

File 254 **lwarp-nextpage.sty**

§ 354 Package nextpage

(*Emulates or patches code by* Peter Wilson.)

Pkg nextpage nextpage is ignored.

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 255 **lwarp-nfssext-cfr.sty**

§ 355 Package nfssext-cfr

(*Emulates or patches code by* Clea F. Rees.)

Pkg nfssext-cfr nfssext-cfr is emulated in HTML, and used as-is in print output.

Results depend on the browser's font.

for HTML output:

```
\LWR@ProvidesPackagePass{nfssext-cfr}[2017/03/28]
```

Macros which are present in the lwarp core are commented out here.

```
\newrobustcmd{\LWR@HTML@lnstyle}{}
\newrobustcmd{\LWR@HTML@osstyle}{}
\newrobustcmd{\LWR@HTML@instyle}{}
\newrobustcmd{\LWR@HTML@sustyle}{}
\newrobustcmd{\LWR@HTML@swstyle}{}
\newrobustcmd{\LWR@HTML@pstyle}{}
\newrobustcmd{\LWR@HTML@tistyle}{}
\newrobustcmd{\LWR@HTML@ostyle}{\LWR@HTML@scshape}
\newrobustcmd{\LWR@HTML@postyle}{\LWR@HTML@scshape}
\newrobustcmd{\LWR@HTML@ltstyle}{}
\newrobustcmd{\LWR@HTML@ofstyle}{}
\newrobustcmd{\LWR@HTML@altstyle}{}
\newrobustcmd{\LWR@HTML@regstyle}{}
\newrobustcmd{\LWR@HTML@embossstyle}{}
\newrobustcmd{\LWR@HTML@ornamentalstyle}{}
\newrobustcmd{\LWR@HTML@qtstyle}{}
\newrobustcmd{\LWR@HTML@shstyle}{}
\newrobustcmd{\LWR@HTML@swashstyle}{}
\newrobustcmd{\LWR@HTML@tmstyle}{}
\newrobustcmd{\LWR@HTML@tvstyle}{\renewcommand*{\LWR@f@family}{tt}}
\newrobustcmd{\LWR@HTML@tstyle}{}
\newrobustcmd{\LWR@HTML@lstyle}{}
\newrobustcmd{\LWR@HTML@tlstyle}{}
\newrobustcmd{\LWR@HTML@plstyle}{}
\newrobustcmd{\LWR@HTML@tostyle}{\LWR@HTML@scshape}
% \newrobustcmd{\LWR@HTML@sishape}{}
\newrobustcmd{\LWR@HTML@olshape}{}
\newrobustcmd{\LWR@HTML@scolshape}{}
\newrobustcmd{\LWR@HTML@ushape}{}
\newrobustcmd{\LWR@HTML@scushape}{}
\newrobustcmd{\LWR@HTML@uishape}{\LWR@HTML@itshape}
\newrobustcmd{\LWR@HTML@rishape}{}
\newrobustcmd{\LWR@HTML@regwidth}{}
\newrobustcmd{\LWR@HTML@nwwidth}{}
\newrobustcmd{\LWR@HTML@cdwidth}{}
\newrobustcmd{\LWR@HTML@ecwidth}{}
\newrobustcmd{\LWR@HTML@ucwidth}{}
\newrobustcmd{\LWR@HTML@etwidth}{}
\newrobustcmd{\LWR@HTML@epwidth}{}
\newrobustcmd{\LWR@HTML@exwidth}{}
\newrobustcmd{\LWR@HTML@uxwidth}{}
\newrobustcmd{\LWR@HTML@mbweight}{\renewcommand*{\LWR@f@series}{md}}
\newrobustcmd{\LWR@HTML@dbweight}{\renewcommand*{\LWR@f@series}{bf}}
\newrobustcmd{\LWR@HTML@sbweight}{\renewcommand*{\LWR@f@series}{bf}}
% \newrobustcmd{\LWR@HTML@ebweight}{\renewcommand*{\LWR@f@series}{eb}}
\newrobustcmd{\LWR@HTML@ubweight}{\renewcommand*{\LWR@f@series}{eb}}
% \newrobustcmd{\LWR@HTML@lgweight}{\renewcommand*{\LWR@f@series}{lg}}
\newrobustcmd{\LWR@HTML@elweight}{\renewcommand*{\LWR@f@series}{lg}}
\newrobustcmd{\LWR@HTML@ulweight}{\renewcommand*{\LWR@f@series}{lg}}
```

```
% \newrobustcmd{\LWR@HTML@itshape}{}
% \newrobustcmd{\LWR@HTML@scshape}{}
% \newrobustcmd{\LWR@HTML@upshape}{}
\newrobustcmd{\LWR@HTML@dfshape}{}

\ifdef{\LWR@HTML@swshape}{}{% duplicated by fontaxes
    \newrobustcmd{\LWR@HTML@swshape}{}
}

\newrobustcmd{\LWR@HTML@ornament}[1]{}

\LWR@formatted{lnstyle}
\LWR@formatted{osstyle}
\LWR@formatted{instyle}
\LWR@formatted{sustyle}
\LWR@formatted{swstyle}
\LWR@formatted{pstyle}
\LWR@formatted{tistyle}
\LWR@formatted{ostyle}
\LWR@formatted{postyle}
\LWR@formatted{ltstyle}
\LWR@formatted{ofstyle}
\LWR@formatted{altstyle}
\LWR@formatted{regstyle}
\LWR@formatted{embossstyle}
\LWR@formatted{ornamentalstyle}
\LWR@formatted{qtstyle}
\LWR@formatted{shstyle}
\LWR@formatted{swashstyle}
\LWR@formatted{tmstyle}
\LWR@formatted{tvstyle}
\LWR@formatted{tstyle}
\LWR@formatted{lstyle}
\LWR@formatted{tlstyle}
\LWR@formatted{plstyle}
\LWR@formatted{tostyle}
% \LWR@formatted{sishape}
\LWR@formatted{olshape}
\LWR@formatted{scolshape}
\LWR@formatted{ushape}
\LWR@formatted{scushape}
\LWR@formatted{uishape}
\LWR@formatted{rishape}
\LWR@formatted{regwidth}
\LWR@formatted{nwwidth}
\LWR@formatted{cdwidth}
\LWR@formatted{ecwidth}
\LWR@formatted{ucwidth}
\LWR@formatted{etwidth}
\LWR@formatted{epwidth}
\LWR@formatted{exwidth}
\LWR@formatted{uxwidth}
\LWR@formatted{mbweight}
\LWR@formatted{dbweight}
\LWR@formatted{sbweight}
```

```
% \LWR@formatted{ebweight}
\LWR@formatted{ubweight}
% \LWR@formatted{lgweight}
\LWR@formatted{elweight}
\LWR@formatted{ulweight}
% \LWR@formatted{itshape}
% \LWR@formatted{scshape}
% \LWR@formatted{upshape}
\LWR@formatted{dfshape}

\ifdef{\LWR@HTML@swshape}{}{% duplicated by fontaxes
    \LWR@formatted{swshape}
}

\LWR@formatted{ornament}


\newcommand{\LWR@null@lnstyle}{}
\newcommand{\LWR@null@osstyle}{}
\newcommand{\LWR@null@instyle}{}
\newcommand{\LWR@null@sustyle}{}
\newcommand{\LWR@null@swstyle}{}
\newcommand{\LWR@null@pstyle}{}
\newcommand{\LWR@null@tistyle}{}
\newcommand{\LWR@null@ostyle}{}
\newcommand{\LWR@null@postyle}{}
\newcommand{\LWR@null@ltstyle}{}
\newcommand{\LWR@null@ofstyle}{}
\newcommand{\LWR@null@altstyle}{}
\newcommand{\LWR@null@regstyle}{}
\newcommand{\LWR@null@embossstyle}{}
\newcommand{\LWR@null@ornamentalstyle}{}
\newcommand{\LWR@null@qtstyle}{}
\newcommand{\LWR@null@shstyle}{}
\newcommand{\LWR@null@swashstyle}{}
\newcommand{\LWR@null@tmstyle}{}
\newcommand{\LWR@null@tvstyle}{}
\newcommand{\LWR@null@tstyle}{}
\newcommand{\LWR@null@lstyle}{}
\newcommand{\LWR@null@tlstyle}{}
\newcommand{\LWR@null@plstyle}{}
\newcommand{\LWR@null@tostyle}{}
% \newcommand{\LWR@null@sishape}{}
\newcommand{\LWR@null@olshape}{}
\newcommand{\LWR@null@scolshape}{}
\newcommand{\LWR@null@ushape}{}
\newcommand{\LWR@null@scushape}{}
\newcommand{\LWR@null@uishape}{}
\newcommand{\LWR@null@rishape}{}
\newcommand{\LWR@null@regwidth}{}
\newcommand{\LWR@null@nwwidth}{}
\newcommand{\LWR@null@cdwidth}{}
\newcommand{\LWR@null@ecwidth}{}
\newcommand{\LWR@null@ucwidth}{}
\newcommand{\LWR@null@etwidth}{}
```

```
\newcommand{\LWR@null@epwidth}{}
\newcommand{\LWR@null@exwidth}{}
\newcommand{\LWR@null@uxwidth}{}
\newcommand{\LWR@null@mbweight}{}
\newcommand{\LWR@null@dbweight}{}
\newcommand{\LWR@null@sbweight}{}
% \newcommand{\LWR@null@ebweight}{}
\newcommand{\LWR@null@ubweight}{}
% \newcommand{\LWR@null@lgweight}{}
\newcommand{\LWR@null@elweight}{}
\newcommand{\LWR@null@ulweight}{}
% \newcommand{\LWR@null@itshape}{}
% \newcommand{\LWR@null@scshape}{}
% \newcommand{\LWR@null@upshape}{}
\newcommand{\LWR@null@dfshape}{}

\ifdef{\LWR@HTML@swshape}{}{% duplicated by fontaxes
    \newcommand{\LWR@null@swshape}{}
}

\newcommand{\LWR@null@ornament}[1]{}

\appto{\LWR@nullfonts}{%
\LetLtxMacro\lnstyle\LWR@null@lnstyle%
\LetLtxMacro\osstyle\LWR@null@osstyle%
\LetLtxMacro\instyle\LWR@null@instyle%
\LetLtxMacro\sustyle\LWR@null@sustyle%
\LetLtxMacro\swstyle\LWR@null@swstyle%
\LetLtxMacro\pstyle\LWR@null@pstyle%
\LetLtxMacro\tistyle\LWR@null@tistyle%
\LetLtxMacro\ostyle\LWR@null@ostyle%
\LetLtxMacro\postyle\LWR@null@postyle%
\LetLtxMacro\ltstyle\LWR@null@ltstyle%
\LetLtxMacro\ofstyle\LWR@null@ofstyle%
\LetLtxMacro\altstyle\LWR@null@altstyle%
\LetLtxMacro\regstyle\LWR@null@regstyle%
\LetLtxMacro\embossstyle\LWR@null@embossstyle%
\LetLtxMacro\ornamentalstyle\LWR@null@ornamentalstyle%
\LetLtxMacro\qtstyle\LWR@null@qtstyle%
\LetLtxMacro\shstyle\LWR@null@shstyle%
\LetLtxMacro\swashstyle\LWR@null@swashstyle%
\LetLtxMacro\tmstyle\LWR@null@tmstyle%
\LetLtxMacro\tvstyle\LWR@null@tvstyle%
\LetLtxMacro\tstyle\LWR@null@tstyle%
\LetLtxMacro\lstyle\LWR@null@lstyle%
\LetLtxMacro\tlstyle\LWR@null@tlstyle%
\LetLtxMacro\plstyle\LWR@null@plstyle%
\LetLtxMacro\tostyle\LWR@null@tostyle%
% \LetLtxMacro\sishape\LWR@null@sishape
\LetLtxMacro\olshape\LWR@null@olshape%
\LetLtxMacro\scolshape\LWR@null@scolshape%
\LetLtxMacro\ushape\LWR@null@ushape%
\LetLtxMacro\scushape\LWR@null@scushape%
\LetLtxMacro\uishape\LWR@null@uishape%
\LetLtxMacro\rishape\LWR@null@rishape%
```

```
\LetLtxMacro\regwidth\LWR@null@regwidth%
\LetLtxMacro\nwwidth\LWR@null@nwwidth%
\LetLtxMacro\cdwidth\LWR@null@cdwidth%
\LetLtxMacro\ecwidth\LWR@null@ecwidth%
\LetLtxMacro\ucwidth\LWR@null@ucwidth%
\LetLtxMacro\etwidth\LWR@null@etwidth%
\LetLtxMacro\epwidth\LWR@null@epwidth%
\LetLtxMacro\exwidth\LWR@null@exwidth%
\LetLtxMacro\uxwidth\LWR@null@uxwidth%
\LetLtxMacro\mbweight\LWR@null@mbweight%
\LetLtxMacro\dbweight\LWR@null@dbweight%
\LetLtxMacro\sbweight\LWR@null@sbweight%
% \LetLtxMacro\ebweight\LWR@null@ebweight%
\LetLtxMacro\ubweight\LWR@null@ubweight%
% \LetLtxMacro\lgweight\LWR@null@lgweight%
\LetLtxMacro\elweight\LWR@null@elweight%
\LetLtxMacro\ulweight\LWR@null@ulweight%
% \LetLtxMacro\itshape\LWR@null@itshape%
% \LetLtxMacro\scshape\LWR@null@scshape%
% \LetLtxMacro\upshape\LWR@null@upshape%
\LetLtxMacro\dfshape\LWR@null@dfshape%
\LetLtxMacro\swshape\LWR@null@swshape%
\LetLtxMacro\ornament\LWR@null@ornament%
}

\newrobustcmd{\LWR@HTML@textln}[1]{#1}
\newrobustcmd{\LWR@HTML@textos}[1]{\textsc{#1}}
\newrobustcmd{\LWR@HTML@textin}[1]{#1}
\newrobustcmd{\LWR@HTML@textsu}[1]{#1}
% \newrobustcmd{\LWR@HTML@textsi}[1]{#1}
\newrobustcmd{\LWR@HTML@textdf}[1]{#1}

\ifdef{\LWR@HTML@swshape}{}{% duplicated by fontaxes
    \newrobustcmd{\LWR@HTML@textsw}[1]{#1}
}

\newrobustcmd{\LWR@HTML@textti}[1]{#1}
\newrobustcmd{\LWR@HTML@textlt}[1]{#1}
\newrobustcmd{\LWR@HTML@textof}[1]{#1}
\newrobustcmd{\LWR@HTML@textalt}[1]{#1}
\newrobustcmd{\LWR@HTML@textreg}[1]{#1}
\newrobustcmd{\LWR@HTML@emboss}[1]{#1}
\newrobustcmd{\LWR@HTML@textorn}[1]{#1}
\newrobustcmd{\LWR@HTML@textqt}[1]{#1}
\newrobustcmd{\LWR@HTML@textsh}[1]{#1}
\newrobustcmd{\LWR@HTML@texttm}[1]{#1}
\newrobustcmd{\LWR@HTML@texttv}[1]{\texttt{#1}}
\newrobustcmd{\LWR@HTML@textl}[1]{#1}
\newrobustcmd{\LWR@HTML@texto}[1]{#1}
\newrobustcmd{\LWR@HTML@textp}[1]{#1}
\newrobustcmd{\LWR@HTML@textt}[1]{#1}
\newrobustcmd{\LWR@HTML@textpl}[1]{#1}
\newrobustcmd{\LWR@HTML@textpo}[1]{\textsc{#1}}
\newrobustcmd{\LWR@HTML@texttl}[1]{#1}
\newrobustcmd{\LWR@HTML@textto}[1]{\textsc{#1}}
```

```
\newrobustcmd{\LWR@HTML@textol}[1]{#1}
\newrobustcmd{\LWR@HTML@textswash}[1]{#1}
\newrobustcmd{\LWR@HTML@textu}[1]{#1}
\newrobustcmd{\LWR@HTML@textscu}[1]{#1}
\newrobustcmd{\LWR@HTML@textui}[1]{\LWR@HTML@textit{#1}}
\newrobustcmd{\LWR@HTML@textri}[1]{#1}
\newrobustcmd{\LWR@HTML@textnw}[1]{#1}
\newrobustcmd{\LWR@HTML@textcd}[1]{#1}
\newrobustcmd{\LWR@HTML@textec}[1]{#1}
\newrobustcmd{\LWR@HTML@textuc}[1]{#1}
\newrobustcmd{\LWR@HTML@textet}[1]{#1}
\newrobustcmd{\LWR@HTML@textep}[1]{#1}
\newrobustcmd{\LWR@HTML@textex}[1]{#1}
\newrobustcmd{\LWR@HTML@textux}[1]{#1}
\newrobustcmd{\LWR@HTML@textrw}[1]{#1}
\newrobustcmd{\LWR@HTML@textmb}[1]{\textmd{#1}}
\newrobustcmd{\LWR@HTML@textdb}[1]{\textbf{#1}}
\newrobustcmd{\LWR@HTML@textsb}[1]{\textbf{#1}}
% \newrobustcmd{\LWR@HTML@texteb}[1]{#1}
\newrobustcmd{\LWR@HTML@textub}[1]{\texteb{#1}}
% \newrobustcmd{\LWR@HTML@textlg}[1]{#1}
\newrobustcmd{\LWR@HTML@textel}[1]{\textlg{#1}}
\newrobustcmd{\LWR@HTML@textul}[1]{\textlg{#1}}

\LWR@formatted{textln}
\LWR@formatted{textos}
\LWR@formatted{textin}
\LWR@formatted{textsu}
% \LWR@formatted{textsi}
\LWR@formatted{textdf}
\LWR@formatted{textsw}
\LWR@formatted{textti}
\LWR@formatted{textlt}
\LWR@formatted{textof}
\LWR@formatted{textalt}
\LWR@formatted{textreg}
\LWR@formatted{emboss}
\LWR@formatted{textorn}
\LWR@formatted{textqt}
\LWR@formatted{textsh}
\LWR@formatted{texttm}
\LWR@formatted{texttv}
\LWR@formatted{textl}
\LWR@formatted{texto}
\LWR@formatted{textp}
\LWR@formatted{textt}
\LWR@formatted{textpl}
\LWR@formatted{textpo}
\LWR@formatted{texttl}
\LWR@formatted{textto}
\LWR@formatted{textol}
\LWR@formatted{textswash}
\LWR@formatted{textu}
\LWR@formatted{textscu}
\LWR@formatted{textui}
```

```
\LWR@formatted{textri}
\LWR@formatted{textnw}
\LWR@formatted{textcd}
\LWR@formatted{textec}
\LWR@formatted{textuc}
\LWR@formatted{textet}
\LWR@formatted{textep}
\LWR@formatted{textex}
\LWR@formatted{textux}
\LWR@formatted{textrw}
\LWR@formatted{textmb}
\LWR@formatted{textdb}
\LWR@formatted{textsb}
% \LWR@formatted{texteb}
\LWR@formatted{textub}
% \LWR@formatted{textlg}
\LWR@formatted{textel}
\LWR@formatted{textul}

\newrobustcmd{\LWR@null@textln}[1]{#1}
\newrobustcmd{\LWR@null@textos}[1]{#1}
\newrobustcmd{\LWR@null@textin}[1]{#1}
\newrobustcmd{\LWR@null@textsu}[1]{#1}
% \newrobustcmd{\LWR@null@textsi}[1]{#1}
\newrobustcmd{\LWR@null@textdf}[1]{#1}

\ifdef{\LWR@HTML@swshape}{}{% duplicated by fontaxes
    \newrobustcmd{\LWR@null@textsw}[1]{#1}
}

\newrobustcmd{\LWR@null@textti}[1]{#1}
\newrobustcmd{\LWR@null@textlt}[1]{#1}
\newrobustcmd{\LWR@null@textof}[1]{#1}
\newrobustcmd{\LWR@null@textalt}[1]{#1}
\newrobustcmd{\LWR@null@textreg}[1]{#1}
\newrobustcmd{\LWR@null@emboss}[1]{#1}
\newrobustcmd{\LWR@null@textorn}[1]{#1}
\newrobustcmd{\LWR@null@textqt}[1]{#1}
\newrobustcmd{\LWR@null@textsh}[1]{#1}
\newrobustcmd{\LWR@null@texttm}[1]{#1}
\newrobustcmd{\LWR@null@texttv}[1]{#1}
\newrobustcmd{\LWR@null@textl}[1]{#1}
\newrobustcmd{\LWR@null@texto}[1]{#1}
\newrobustcmd{\LWR@null@textp}[1]{#1}
\newrobustcmd{\LWR@null@textt}[1]{#1}
\newrobustcmd{\LWR@null@textpl}[1]{#1}
\newrobustcmd{\LWR@null@textpo}[1]{#1}
\newrobustcmd{\LWR@null@texttl}[1]{#1}
\newrobustcmd{\LWR@null@textto}[1]{#1}
\newrobustcmd{\LWR@null@textol}[1]{#1}
\newrobustcmd{\LWR@null@textswash}[1]{#1}
\newrobustcmd{\LWR@null@textu}[1]{#1}
\newrobustcmd{\LWR@null@textscu}[1]{#1}
\newrobustcmd{\LWR@null@textui}[1]{#1}
\newrobustcmd{\LWR@null@textri}[1]{#1}
```

```
\newrobustcmd{\LWR@null@textnw}[1]{#1}
\newrobustcmd{\LWR@null@textcd}[1]{#1}
\newrobustcmd{\LWR@null@textec}[1]{#1}
\newrobustcmd{\LWR@null@textuc}[1]{#1}
\newrobustcmd{\LWR@null@textet}[1]{#1}
\newrobustcmd{\LWR@null@textep}[1]{#1}
\newrobustcmd{\LWR@null@textex}[1]{#1}
\newrobustcmd{\LWR@null@textux}[1]{#1}
\newrobustcmd{\LWR@null@textrw}[1]{#1}
\newrobustcmd{\LWR@null@textmb}[1]{#1}
\newrobustcmd{\LWR@null@textdb}[1]{#1}
\newrobustcmd{\LWR@null@textsb}[1]{#1}
% \newrobustcmd{\LWR@null@texteb}[1]{#1}
\newrobustcmd{\LWR@null@textub}[1]{#1}
% \newrobustcmd{\LWR@null@textlg}[1]{#1}
\newrobustcmd{\LWR@null@textel}[1]{#1}
\newrobustcmd{\LWR@null@textul}[1]{#1}

\appto{\LWR@nullfonts}{%
\LetLtxMacro\textln\LWR@null@textln%
\LetLtxMacro\textos\LWR@null@textos%
\LetLtxMacro\textin\LWR@null@textin%
\LetLtxMacro\textsu\LWR@null@textsu%
% \LetLtxMacro\textsi\LWR@null@textsi%
\LetLtxMacro\textdf\LWR@null@textdf%
\LetLtxMacro\textsw\LWR@null@textsw%
\LetLtxMacro\textti\LWR@null@textti%
\LetLtxMacro\textlt\LWR@null@textlt%
\LetLtxMacro\textof\LWR@null@textof%
\LetLtxMacro\textalt\LWR@null@textalt%
\LetLtxMacro\textreg\LWR@null@textreg%
\LetLtxMacro\emboss\LWR@null@emboss%
\LetLtxMacro\textorn\LWR@null@textorn%
\LetLtxMacro\textqt\LWR@null@textqt%
\LetLtxMacro\textsh\LWR@null@textsh%
\LetLtxMacro\texttm\LWR@null@texttm%
\LetLtxMacro\texttv\LWR@null@texttv%
\LetLtxMacro\textl\LWR@null@textl%
\LetLtxMacro\texto\LWR@null@texto%
\LetLtxMacro\textp\LWR@null@textp%
\LetLtxMacro\textt\LWR@null@textt%
\LetLtxMacro\textpl\LWR@null@textpl%
\LetLtxMacro\textpo\LWR@null@textpo%
\LetLtxMacro\texttl\LWR@null@texttl%
\LetLtxMacro\textto\LWR@null@textto%
\LetLtxMacro\textol\LWR@null@textol%
\LetLtxMacro\textswash\LWR@null@textswash%
\LetLtxMacro\textu\LWR@null@textu%
\LetLtxMacro\textscu\LWR@null@textscu%
\LetLtxMacro\textui\LWR@null@textui%
\LetLtxMacro\textri\LWR@null@textri%
\LetLtxMacro\textnw\LWR@null@textnw%
\LetLtxMacro\textcd\LWR@null@textcd%
\LetLtxMacro\textec\LWR@null@textec%
\LetLtxMacro\textuc\LWR@null@textuc%
```

```
\LetLtxMacro\textet\LWR@null@textet%
\LetLtxMacro\textep\LWR@null@textep%
\LetLtxMacro\textex\LWR@null@textex%
\LetLtxMacro\textux\LWR@null@textux%
\LetLtxMacro\textrw\LWR@null@textrw%
\LetLtxMacro\textmb\LWR@null@textmb%
\LetLtxMacro\textdb\LWR@null@textdb%
\LetLtxMacro\textsb\LWR@null@textsb%
% \LetLtxMacro\texteb\LWR@null@texteb%
\LetLtxMacro\textub\LWR@null@textub%
% \LetLtxMacro\textlg\LWR@null@textlg%
\LetLtxMacro\textel\LWR@null@textel%
\LetLtxMacro\textul\LWR@null@textul%
}

\providecommand*{\zeroslash}{0}
\newrobustcmd*{\LWR@HTML@zeroslash}{0}
\LWR@formatted{zeroslash}
```

File 256 **lwarp-nicefrac.sty**

§ 356 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:

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\nicefrac}[3][]{#2/#3}}
\end{warpMathJax}
```

File 257 **lwarp-niceframe.sty**

§ 357 Package niceframe

Pkg niceframe niceframe is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{niceframe}% the original date is in yyyy/dd/mm format


\newcommand{\LWR@niceframe}[3]{%
    \begin{LWR@setvirtualpage}*%
    \setlength{\LWR@templengthone}{#1}%
    \begin{BlockClass}[max-width:\LWR@printlength{\LWR@templengthone}]{#3}%
    #2
    \end{BlockClass}%
    \end{LWR@setvirtualpage}%
}

\newcommand{\niceframe}[2][\textwidth]{\LWR@niceframe{#1}{#2}{niceframe}}
\newcommand{\curlyframe}[2][\textwidth]{\LWR@niceframe{#1}{#2}{curlyframe}}
\newcommand{\artdecoframe}[2][\textwidth]{\LWR@niceframe{#1}{#2}{artdecoframe}}

\newcommand{\generalframe}[9]{\LWR@niceframe{\textwidth}{#9}{generalframe}}
```

File 258 **lwarp-nomencl.sty**

§ 358 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]
```

\BaseJobname is added to the label in case xr or xr-hyper are used.

```
\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{\BaseJobname-autopage-#1}}%
```

File 259 **lwarp-nonfloat.sty**

§ 359 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 260 **lwarp-nonumonpart.sty**

§ 360 Package nonumonpart

Pkg nonumonpart nonumonpart is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{nonumonpart}[2011/04/15]
```

File 261 **lwarp-nopageno.sty**

§ 361 Package nopageno

Pkg nopageno nopageno is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{nopageno}[1989/01/01]
```

File 262 **lwarp-notes.sty**

§ 362 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 263 **lwarp-notespages.sty**

§ 363 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 264 **lwarp-nowidow.sty**

§ 364 Package nowidow

(*Emulates or patches code by* Raphaël Pinson.)

Pkg nowidow nowidow is ignored.

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 265 **lwarp-ntheorem.sty**

§ 365 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.

§ 365.1 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.

§ 365.2 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]
```

§ 365.3 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}
}
```

§ 365.4 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%
}
```

§ 365.5 \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}
}
```

§ 365.6 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}]}

\renewtheoremstyle{nonumberbreak}%
  {\item[
    \InlineClass{theoremheaderbreak}{##1\theorem@separator}\newline
    ]}%
  {\item[
    \InlineClass{theoremheaderbreak}{##1\ (##3)\theorem@separator}\newline
    ]}

\renewtheoremstyle{empty}%
  {\item[]}%
  {\item[
    \InlineClass{theoremheaderplain}{##3}]}

\renewtheoremstyle{emptybreak}%
  {\item[]}%
  {\item[
    \InlineClass{theoremheaderplain}{##3}] \ \newline}
```

§ 365.7 Additional objects

The following manually adjust the css for the standard configuration objects which are not a purely plain style:

```
\ifbool{LWR@ntheoremamsthm}{}{%
```

Upright text via CSS:

```
    \newtheoremstyle{plainupright}%
    {\item[
        \InlineClass{theoremheaderplain}{##1\ ##2\theorem@separator}]}%
    {\item[
        \InlineClass{theoremheaderplain}{##1\ ##2\ (##3)\theorem@separator}]}
```

Upright text and small caps header via CSS:

```
    \newtheoremstyle{nonumberplainuprightsc}%
    {\item[
        \InlineClass{theoremheadersc}{##1\theorem@separator}]}%
    {\item[
        \InlineClass{theoremheadersc}{##1\ (##3)\theorem@separator}]}
}% not amsthm
```

§ 365.8 Renewed standard configuration

The following standard configuration is renewed using the new CSS:

```
\ifbool{LWR@ntheoremamsthm}{}{%

\ifx\thm@usestd\@undefined
\else
    \theoremnumbering{arabic}
    \theoremstyle{plain}
    \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
```

§ 365.9 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
```

§ 365.10 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%
}
```

§ 365.11 `\NoEndMark`

```
\gdef\NoEndMark{\global\setendmarkfalse}
```

§ 365.12 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}%
}
```

§ 365.13 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
```

§ 365.14 Cross-referencing

`\thref {`⟨*label*⟩`}`

```
\newcommand*{\thref}[1]{\cref{#1}}%
```

File 266 **lwarp-octave.sty**

§ 366 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 267 **lwarp-overpic.sty**

§ 367 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 generation. These are sent to \fontsize to adjust the font size for scaling differences between the print and HTML versions of the document. Renew these macros before using the overpic and Overpic environments.

See section 84.2 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 268 **lwarp-pagegrid.sty**

§ 368 Package pagegrid

Pkg pagegrid pagegrid is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pagegrid}[2016/05/16]

\newcommand*{\pagegridsetup}[1]{}
```

File 269 **lwarp-pagenote.sty**

§ 369 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 270 **lwarp-pagesel.sty**

§ 370 Package pagesel

Pkg pagesel pagesel is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pagesel}[2016/05/16]
```

File 271 **lwarp-paralist.sty**

§ 371 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 272 **lwarp-parallel.sty**

§ 372 Package

parallel

(*Emulates or patches code by* Matthias Eckermann.)

Pkg parallel parallel is emulated.

Package options are ignored. Footnotes are treated as normal lwarp footnotes.

Environment option c gives side-by-side s of class minipage, each of whose width is a percent depending on the given left and right widths, proportional to \linewidth.

Inside each environment, \linewidth and \textwidth are set for the print-output sizes.

for HTML output: Discard all options for lwarp-parallel:

```
\LWR@ProvidesPackageDrop{parallel}[2003/04/13]

\newcounter{LWR@parallel@Lwidth}
\newcounter{LWR@parallel@Rwidth}
\newcommand*{\LWR@parallel@border}

\newenvironment*{Parallel}[3][]%
    {%
        \LWR@printpendingfootnotes%
        \setlength{\linewidth}{\LWR@userstextwidth}%
        \setlength{\textwidth}{\LWR@userstextwidth}%
        \renewcommand*{\LWR@parallel@border}{}%
        \ifstrequal{#1}{v}%
            {%
           \renewcommand*{\LWR@parallel@border}{ ; border-left: 2px solid black}%
            }%
            {}%
        \ifblank{#2}{%
            \ifblank{#3}{% {}{}
                \setcounter{LWR@parallel@Lwidth}{45}%
                \setcounter{LWR@parallel@Rwidth}{45}%
            }% {}{}
            {% {}{x}
                \setlength{\LWR@templengthone}{\linewidth-#3}%
                \setcounter{LWR@parallel@Lwidth}{%
                    90*\ratio{\LWR@templengthone}{\linewidth}%
                }%
                \setcounter{LWR@parallel@Rwidth}{%
                    90*\ratio{#3}{\linewidth}%
                }%
            }% {}{x}
        }% #2 blank
        {% #2 non-blank
            \ifblank{#3}{% {x}{}
```

```
                \setcounter{LWR@parallel@Lwidth}{%
                    90*\ratio{#2}{\linewidth}%
                }%
                \setlength{\LWR@templengthone}{\linewidth-#2}%
                \setcounter{LWR@parallel@Rwidth}{%
                    90*\ratio{\LWR@templengthone}{\linewidth}%
                }%
            }% {x}{}
            {% {x}{x}
                \setcounter{LWR@parallel@Lwidth}{%
                    90*\ratio{#2}{\linewidth}%
                }%
                \setcounter{LWR@parallel@Rwidth}{%
                    90*\ratio{#3}{\linewidth}%
                }%
            }% {x}{x}
        }% #2 non-blank
    }
    {%
        \ParallelAtEnd%
        \renewcommand*{\ParallelAtEnd}{}%
        \LWR@printpendingfootnotes%
    }

\newcommand*{\ParallelLText}[1]{%
    \begin{BlockClass}[%
        width:\arabic{LWR@parallel@Lwidth}\% ; % space
        padding: .5ex 1\% ; % space
    ]{minipage}%
    #1%
    \end{BlockClass}%
}

\newcommand*{\ParallelRText}[1]{%
    \begin{BlockClass}[%
        width:\arabic{LWR@parallel@Rwidth}\% ; % space
        padding: .5ex 1\% ; % space
        \LWR@parallel@border%
    ]{minipage}%
    #1%
    \end{BlockClass}%
}

\newcommand*{\ParallelPar}{\LWR@printpendingfootnotes}

\newcommand*{\ParallelAtEnd}{}
```

File 273 **lwarp-parcolumns.sty**

§ 373 Package

parcolumns

(*Emulates or patches code by* Jonathan Sauer.)

Pkg parcolumns parcolumns is emulated.

rulebetween is honored. The other keys are ignored, including colwidths.

Each column is placed inside a of class minipage, each of whose width is fixed at 85% divided by the number of columns. In most cases, this results in side-by-side minipages adapting to the browser width. Inside each minipage, \linewidth, \textwidth, and \textheight are set for a virtual 6×9 inch page, with \linewidth divided by the number of columns.

for HTML output: Discard all options for lwarp-parcolumns:

```
\RequirePackage{keyval}%

\LWR@ProvidesPackageDrop{parcolumns}[2004/11/25]


\newcounter{LWR@parcolumns@numcols}
\newcounter{LWR@parcolumns@thiscol}
\newcounter{LWR@parcolumns@width}
\newbool{LWR@parcolumns@started}
\newbool{LWR@parcolumns@rule}

\define@key{LWRparcols}{colwidths}{}
\define@key{LWRparcols}{distance}{}
\define@key{LWRparcols}{rulebetween}[true]{%
    \setbool{LWR@parcolumns@rule}{#1}%
}
\define@key{LWRparcols}{nofirstindent}{}
\define@key{LWRparcols}{sloppy}{}
\define@key{LWRparcols}{sloppyspaces}{}

\newenvironment*{parcolumns}[2][]
    {%
        \begin{LWR@setvirtualpage}*[#2]%
        \setcounter{LWR@parcolumns@numcols}{#2}%
        \setcounter{LWR@parcolumns@thiscol}{1}%
        \boolfalse{LWR@parcolumns@started}%
        \boolfalse{LWR@parcolumns@rule}%
        \setcounter{LWR@parcolumns@width}{%
            85/#2
        }%
        \setkeys{LWRparcols}{#1}%
    }
    {\end{LWR@setvirtualpage}}

\newcommand{\LWR@parcolumns@onecol}[1]{%
    \ifbool{LWR@parcolumns@started}%
        {}%
        {%
            \LWR@htmldivclass{parcolumns}%
            \booltrue{LWR@parcolumns@started}%
        }%
    \ifboolexpr{%
        bool {LWR@parcolumns@rule} and
        test {%
```

```
            \ifnumgreater
                {\value{LWR@parcolumns@thiscol}}
                {1}
        }%
    }%
        {\renewcommand{\LWR@tempone}{ ; border-left: 2px solid black}}%
        {\renewcommand{\LWR@tempone}{}}%
    \begin{BlockClass}[%
        width:\arabic{LWR@parcolumns@width}\% ; % space
        padding: .5ex 1\% ; % space
        \LWR@tempone%
    ]{minipage}%
    #1%
    \end{BlockClass}%
    \addtocounter{LWR@parcolumns@thiscol}{1}%
}

\newcommand{\colchunk}[2][\value{LWR@parcolumns@thiscol}]{%
    \whileboolexpr{%
        test {%
            \ifnumcomp%
                {\value{LWR@parcolumns@thiscol}}
                {<}
                {#1}%
        }%
    }{%
        \LWR@parcolumns@onecol{}%
    }%
    \LWR@parcolumns@onecol{#2}%
}

\newcommand*{\colplacechunks}{%
    \ifbool{LWR@parcolumns@started}%
        {%
            \LWR@htmldivclassend{div}%
            \boolfalse{LWR@parcolumns@started}%
        }%
        {}%
    \setcounter{LWR@parcolumns@thiscol}{1}%
}
```

File 274 **lwarp-parnotes.sty**

§ 374 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 275 **lwarp-parskip.sty**

§ 375 Package

parskip

Pkg parskip parskip is ignored.

for HTML output: Discard all options for lwarp-parskip.

```
\LWR@ProvidesPackageDrop{parskip}[2001/04/09]
```

File 276 **lwarp-pbox.sty**

§ 376 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 277 **lwarp-pdfcol.sty**

§ 377 Package pdfcol

Pkg pdfcol pdfcol is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pdfcol}[2018/11/01]

\ltx@newif\ifpdfcolAvailable
\pdfcolAvailablefalse

\def\pdfcolErrorNoStacks{
    \PackageInfo{lwarp-pdfcol}{Ignoring pdfcol for HTML output.}
}

\def\pdfcolInitStack#1{}%

\long\def\pdfcolIfStackExists#1#2#3{#3}%

\def\pdfcolSwitchStack#1{}%

\def\pdfcolSetCurrentColor{}%

\def\pdfcolSetCurrent#1{}%
```

File 278 **lwarp-pdfcolfoot.sty**

§ 378 Package pdfcolfoot

Pkg pdfcolfoot pdfcolfoot is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pdfcolfoot}[2016/05/16]

\newcommand*{\pdfcolfoot@switch}{}

\newcommand*{\pdfcolfoot@current}{}
```

File 279 **lwarp-pdfcolmk.sty**

§ 379 Package pdfcolmk

Pkg pdfcolmk pdfcolmk is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pdfcolmk}[2016/05/16]
```

File 280 **lwarp-pdfcolparallel.sty**

§ 380 Package pdfcolparallel

Pkg pdfcolparallel pdfcolparallel is ignored.

for HTML output:

```
\RequirePackage{keyval}%

\LWR@ProvidesPackageDropA{pdfcolparallel}{2016/05/16}
```

Pass options to parallel:

```
\DeclareOption*{%
    \PassoptionsToPackage{\CurrentOption}{parallel}%
}
```

Process the options:

```
\LWR@ProvidesPackageDropB
```

Require parallel with the given options:

```
\RequirePackage{parallel}[2003/04/13]
```

Ignore the new key:

```
\define@key{parallel}{rulebetweencolor}{}
```

File 281 **lwarp-pdfcolparcolumns.sty**

§ 381 Package pdfcolparcolumns

Pkg pdfcolparcolumns pdfcolparcolumns is ignored.

for HTML output:

```
\LWR@ProvidesPackageDropA{pdfcolparcolumns}{2016/05/16}
```

Pass options to parcolumns:

```
\DeclareOption*{%
    \PassoptionsToPackage{\CurrentOption}{parcolumns}%
}
```

Process the options:

```
\LWR@ProvidesPackageDropB
```

Require parcolumns with the given options:

```
\RequirePackage{parcolumns}[2004/11/25]
```

Ignore the new key:

```
\define@key{LWRparcols}{rulebetweencolor}{}
```

File 282 **lwarp-pdfcomment.sty**

§ 382 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 283 **lwarp-pdfcrypt.sty**

§ 383 Package pdfcrypt

Pkg pdfcrypt pdfcrypt is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pdfcrypt}[2016/05/16]
```

```
\newcommand*{\pdfcryptsetup}[1]{}
```

File 284 **lwarp-pdflscape.sty**

§ 384 Package pdflscape

Pkg pdflscape pdflscape is ignored.

for HTML output: Discard all options for lwarp-pdflscape:

```
\LWR@ProvidesPackageDrop{pdflscape}[2016/05/14]
```

File 285 **lwarp-pdfmarginpar.sty**

§ 385 Package pdfmarginpar

Pkg pdfmarginpar pdfmarginpar is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pdfmarginpar}[2011/08/05]
```

```
\newcommand{\pdfmarginpar}[2][]{}
\newcommand{\pdfmarginparset}[1]{}
```

File 286 **lwarp-pdfpages.sty**

§ 386 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 target:

```
    \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 the user's paper size.

```
\xpretocmd{\includepdf}{%
    \begingroup%
    \setlength{\paperwidth}{\LWR@userspaperwidth}%
    \setlength{\paperheight}{\LWR@userspaperheight}%
}{}{}

\xapptocmd{\includepdf}{%
    \endgroup%
}{}{}
```

`\includepdfmerge` Patched to set the user's paper size.

```
\xpretocmd{\includepdfmerge}{%
    \begingroup%
```

```
    \setlength{\paperwidth}{\LWR@userspaperwidth}%
    \setlength{\paperheight}{\LWR@userspaperheight}%
}{}{}

\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 287 **lwarp-pdfprivacy.sty**

§ 387 Package pdfprivacy

Pkg pdfprivacy pdfprivacy is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pdfprivacy}[2017/12/03]
```

File 288 **lwarp-pdfrender.sty**

§ 388 Package pdfrender

Pkg pdfrender pdfrender is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pdfrender}[2016/05/17]

\newcommand*{\pdfrender}[1]{}
\newcommand{\textpdfrender}[2]{#2}
```

File 289 **lwarp-pdfsync.sty**

§ 389 Package pdfsync

(*Emulates or patches code by* J. Laurens.)

Pkg pdfsync pdfsync is ignored.

for HTML output: Discard all options for lwarp-pdfsync:

```
\LWR@ProvidesPackageDrop{pdfsync}[2008/01/26]

\newcommand*{\pdfsync}{}
\newcommand*{\pdfsyncstart}{}
\newcommand*{\pdfsyncstop}{}
```

File 290 **lwarp-pdftricks.sty**

§ 390 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:

```
1 \LWR@ProvidesPackagePass{pdftricks}[2003/08/10]
```

Reuse the print-mode images:

```
2 \def\PDFTfigname{\BaseJobname-fig\thepsfig}
```

If the `.pdf` images have not yet been converted to `.svg` then an error about a missing file will occur. Warn the user to convert the images.

```
3 \PackageWarning{lwarp-pdftricks}{
4 When the pdftricks images change,
5 remember to convert PDF images to SVG using 'lwarpmk pdftosvg *-fig.pdf',
6 }
7
8 \AfterEndDocument{\typeout{***}}
9 \AfterEndDocument{\typeout{*** Note: If pdftricks images are not found, new, or updated,}}
10 \AfterEndDocument{\typeout{*** \space use 'lwarpmk pdftosvg \BaseJobname-fig*.pdf'}}
11 \AfterEndDocument{\typeout{***}}
```

File 291 **lwarp-pdfx.sty**

§ 391 Package

pdfx

Pkg pdfx pdfx is ignored.

for HTML output:

```
1 \LWR@ProvidesPackageDrop{pdfx}[2017/05/18]
```

File 292 **lwarp-perpage.sty**

§ 392 Package

perpage

(*Emulates or patches code by* David Kastrup.)

Pkg perpage

perpage is mostly ignored, but support is added for footnote counters.

There is no page number in HTML, so most counters are not reset. If the document redefines \the<countername> to include \theperpage, it is necessary to place that redefininition inside a warpprint environment to avoid modifying the HTML defintions.

\AddAbsoluteCounter must not be inside warpprint, as the counter must be added for HTML also, although it is not incremented.

footnote numbering

To have footnote numbers reset each time footnotes are printed:

```
\setcounter{footnoteReset}{1}
```

For bigfoot, manyfoot, or perpage:

```
\MakePerPage{footnoteX}
— or —
\MakeSortedPerPage{footnoteX}
```

The footnotes are reset when they are printed, according to section level as set by FootnoteDepth, which is not necessarily by HTML page. This is recommended for \alph, \Alph, or \fnsymbol footnotes, due to the limited number of symbols which are available.

for HTML output:

```
\LWR@ProvidesPackageDrop{perpage}[2014/10/25]

\newcommand\AddAbsoluteCounter[1]
{
    \@ifundefined{c@abs#1}{%
        \expandafter\newcount\csname c@abs#1\endcsname
        \global\value{abs#1}\@ne
%         \global\expandafter\let\csname cl@abs#1\endcsname\@empty
        \expandafter\xdef\csname theabs#1\endcsname{%
%             \noexpand\number \csname c@abs#1\endcsname
        }%
%         \global\@namedef{c@pabs@#1}{\pp@cl@begin
%         \stepcounter{abs#1}%
%         \pp@cl@end}%
%         \@addtoreset{pabs@#1}{#1}
    }
    {}
}

\AddAbsoluteCounter{page}
\def\theabspage{1}

\newcommand*\MakePerPage[2][1]{%
    \ifltxcounter{#2Reset}{%
        \setcounter{#2Reset}{#1}%
    }{

}%
}

\newcommand*\MakeSorted[1]{}

```

```
\newcommand*\MakeSortedPerPage[2][1]{%
    \ifltxcounter{#2Reset}{%
        \setcounter{#2Reset}{#1}%
    }{
}%
}

\newcommand*{\theperpage}{1}
```

File 293 **lwarp-pfnote.sty**

§ 393 Package

pfnote

Pkg pfnote pfnote is ignored.

pfnote

⚠ pfnote numbers While emulating pfnote, lwarp is not able to reset HTML footnote numbers per page number to match the printed version, as HTML has no concept of page numbers. lwarp therefore uses continuous footnote numbering even for pfnote.

for HTML output:

```
\LWR@ProvidesPackageDrop{pfnote}[1999/07/14]
```

File 294 **lwarp-phfqit.sty**

§ 394 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 295 **lwarp-physics.sty**

§ 395 Package

physics

(*Emulates or patches code by* Sergio C. de la Barrera.)

Pkg physics physics works as-is for HTML with SVG math.

For MathJax, emulation is provided via MathJax macros. These are not the same as the third-party MathJax extension.

- The `notrig` option is honored.
- ⚠ Starred macros are not yet detected.
- ⚠ Most macros don't work with `\big`, etc.
- ⚠ Macros do not auto-detect variable numbers of mandatory arguments. Provide empty `{}` arguments for those which are not used.
- ⚠ Many of the macros do not work with auto-detected delimiters. Use the delimiter-specific versions instead. Some macros do not even consider the following arguments, so they may work as expected.
- ⚠ For `\Re` and `\Im`, the arguments must be in braces.
- For `\functionalderivative`, for the example in the manual with $(E - TS)$, enclose the parens in braces.
- ⚠ `\expectationvalue` requires and uses two mandatory arguments, unlike the third-party MathJax physics extension.
- Each of `\matrixquantity`, `\smallmatrixquantity`, and `\matrixdeterminant` work, while `\identitymatrix` and the following simply print a place-holder, and must be replaced by hand.

for HTML output:

```
\LWR@ProvidesPackagePass{physics}% No date is provided by the package.

% doesn't work with \big, etc:
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\quantity}{}}
\CustomizeMathJax{\let\qty\quantity}
\CustomizeMathJax{\newcommand{\pqty}[1]{\left(#1\right)}}
\CustomizeMathJax{\newcommand{\bqty}[1]{\left\lbrack#1\right\rbrack}}
\CustomizeMathJax{\newcommand{\vqty}[1]{\left\vert#1\right\rvert}}
\CustomizeMathJax{\newcommand{\Bqty}[1]{\left\lbrace#1\right\rbrace}}

% doesn't work with \big, etc., no star
\CustomizeMathJax{\newcommand{\absolutevalue}[1]{\left\lvert#1\right\rvert}}
\CustomizeMathJax{\let\abs\absolutevalue}

% doesn't work with \big, etc., no star
\CustomizeMathJax{\newcommand{\norm}[1]{%
    \left\lvert\left\lvert#1\right\rvert\right\rvert}%
}

% doesn't work with \big, etc., no star, no paren or bracket
\CustomizeMathJax{\newcommand{\evaluated}[1]{#1\vert}}
\CustomizeMathJax{\let\eval\evaluated}

% no \Big, star
```

```
\CustomizeMathJax{\newcommand{\order}[1]{\mathcal{O}\left(#1\right)}}

% no \Big, star
\CustomizeMathJax{\newcommand{\commutator}[2]{\left\lbrack#1,#2\right\rbrack}}
\CustomizeMathJax{\let\comm\commutator}

% no \Big, star
\CustomizeMathJax{\newcommand{\anticommutator}[2]{%
    \left\lbrace#1,#2\right\rbrace}%
}
\CustomizeMathJax{\let\acomm\anticommutator}

% no \Big, star
\CustomizeMathJax{\let\poissonbracket\anticommutator}
\CustomizeMathJax{\let\pb\anticommutator}

% no star
\CustomizeMathJax{\newcommand{\vectorbold}[1]{\mathbf{#1}}}
\CustomizeMathJax{\let\vb\vectorbold}

% no star
\CustomizeMathJax{\newcommand{\vectorarrow}[1]{\vec{\mathbf{#1}}}}
\CustomizeMathJax{\let\va\vectorarrow}

% no star
\CustomizeMathJax{\newcommand{\vectorunit}[1]{\mathbf{\hat{#1}}}}
\CustomizeMathJax{\let\va\vectorunit}

\CustomizeMathJax{\newcommand{\dotproduct}{\boldsymbol\cdot}}
\CustomizeMathJax{\let\vdot\dotproduct}

\CustomizeMathJax{\newcommand{\crossproduct}{\boldsymbol\times}}
\CustomizeMathJax{\let\cross\crossproduct}
\CustomizeMathJax{\let\cp\crossproduct}

\CustomizeMathJax{\newcommand{\gradient}{\mathbf{\nabla}}}
\CustomizeMathJax{\let\grad\gradient}

\CustomizeMathJax{\let\divisionsymbol\div}

\CustomizeMathJax{\newcommand{\divergence}{\nabla\vdot}}
\CustomizeMathJax{\let\div\divergence}

\CustomizeMathJax{\newcommand{\curl}{\nabla\cross}}

\CustomizeMathJax{\newcommand{\laplacian}{\nabla^2}}

% responds to notrig option
\ifx\trigopt 1
    \CustomizeMathJax{\DeclareMathOperator{\sine}{sin}}
    \CustomizeMathJax{\DeclareMathOperator{\cosine}{cos}}
    \CustomizeMathJax{\DeclareMathOperator{\tangent}{tan}}
    \CustomizeMathJax{\DeclareMathOperator{\cosecant}{csc}}
    \CustomizeMathJax{\DeclareMathOperator{\secant}{sec}}
    \CustomizeMathJax{\DeclareMathOperator{\cotangent}{cot}}
```

```
    \CustomizeMathJax{\DeclareMathOperator{\arcsine}{arcsin}}
    \CustomizeMathJax{\DeclareMathOperator{\arccosine}{arccos}}
    \CustomizeMathJax{\DeclareMathOperator{\arctangent}{arctan}}
    \CustomizeMathJax{\DeclareMathOperator{\hypsine}{sinh}}
    \CustomizeMathJax{\DeclareMathOperator{\hypcosine}{cosh}}
    \CustomizeMathJax{\DeclareMathOperator{\hyptangent}{tanh}}
    \CustomizeMathJax{\DeclareMathOperator{\hypcotangent}{coth}}
    \CustomizeMathJax{\DeclareMathOperator{\exponential}{exp}}
    \CustomizeMathJax{\DeclareMathOperator{\logarithm}{log}}
    \CustomizeMathJax{\DeclareMathOperator{\naturallogarithm}{\ln}}
    \CustomizeMathJax{\DeclareMathOperator{\determinant}{det}}
    \CustomizeMathJax{\DeclareMathOperator{\Probability}{Pr}}
    \CustomizeMathJax{\newcommand{\sin}[1][{}]{\mathrm{sin}^#1}}
    \CustomizeMathJax{\newcommand{\sinh}[1][{}]{\mathrm{sinh}^#1}}
    \CustomizeMathJax{\newcommand{\arcsin}[1][{}]{\mathrm{arcsin}^#1}}
    \CustomizeMathJax{\newcommand{\asin}[1][{}]{\mathrm{asin}^#1}}
    \CustomizeMathJax{\newcommand{\cos}[1][{}]{\mathrm{cos}^#1}}
    \CustomizeMathJax{\newcommand{\cosh}[1][{}]{\mathrm{cosh}^#1}}
    \CustomizeMathJax{\newcommand{\arccos}[1][{}]{\mathrm{arccos}^#1}}
    \CustomizeMathJax{\newcommand{\acos}[1][{}]{\mathrm{acos}^#1}}
    \CustomizeMathJax{\newcommand{\tan}[1][{}]{\mathrm{tan}^#1}}
    \CustomizeMathJax{\newcommand{\tanh}[1][{}]{\mathrm{tanh}^#1}}
    \CustomizeMathJax{\newcommand{\arctan}[1][{}]{\mathrm{arctan}^#1}}
    \CustomizeMathJax{\newcommand{\atan}[1][{}]{\mathrm{atan}^#1}}
    \CustomizeMathJax{\newcommand{\csc}[1][{}]{\mathrm{csc}^#1}}
    \CustomizeMathJax{\newcommand{\csch}[1][{}]{\mathrm{csch}^#1}}
    \CustomizeMathJax{\newcommand{\arccsc}[1][{}]{\mathrm{arccsc}^#1}}
    \CustomizeMathJax{\newcommand{\acsc}[1][{}]{\mathrm{acsc}^#1}}
    \CustomizeMathJax{\newcommand{\sec}[1][{}]{\mathrm{sec}^#1}}
    \CustomizeMathJax{\newcommand{\sech}[1][{}]{\mathrm{sech}^#1}}
    \CustomizeMathJax{\newcommand{\arcsec}[1][{}]{\mathrm{arcsec}^#1}}
    \CustomizeMathJax{\newcommand{\asec}[1][{}]{\mathrm{asec}^#1}}
    \CustomizeMathJax{\newcommand{\cot}[1][{}]{\mathrm{cot}^#1}}
    \CustomizeMathJax{\newcommand{\coth}[1][{}]{\mathrm{coth}^#1}}
    \CustomizeMathJax{\newcommand{\arccot}[1][{}]{\mathrm{arccot}^#1}}
    \CustomizeMathJax{\newcommand{\acot}[1][{}]{\mathrm{acot}^#1}}
\else
    \CustomizeMathJax{\DeclareMathOperator{\arccsc}{arccsc}}
    \CustomizeMathJax{\DeclareMathOperator{\arcsec}{arcsec}}
    \CustomizeMathJax{\DeclareMathOperator{\arccot}{arccot}}
    \CustomizeMathJax{\DeclareMathOperator{\asin}{asin}}
    \CustomizeMathJax{\DeclareMathOperator{\acos}{acos}}
    \CustomizeMathJax{\DeclareMathOperator{\atan}{atan}}
    \CustomizeMathJax{\DeclareMathOperator{\acsc}{acsc}}
    \CustomizeMathJax{\DeclareMathOperator{\asec}{asec}}
    \CustomizeMathJax{\DeclareMathOperator{\acot}{acot}}
    \CustomizeMathJax{\DeclareMathOperator{\csch}{csch}}
    \CustomizeMathJax{\DeclareMathOperator{\sech}{sech}}
    \CustomizeMathJax{\newcommand{\tr}{\trace}}
    \CustomizeMathJax{\newcommand{\Tr}{\Trace}}
    \CustomizeMathJax{\newcommand{\Res}{\Residue}}
\fi

\CustomizeMathJax{\DeclareMathOperator{\trace}{tr}}
\CustomizeMathJax{\let\tr\trace}
```

```
\CustomizeMathJax{\DeclareMathOperator{\Trace}{Tr}}
\CustomizeMathJax{\let\Tr\Trace}
\CustomizeMathJax{\DeclareMathOperator{\rank}{rank}}
\CustomizeMathJax{\DeclareMathOperator{\erf}{erf}}
\CustomizeMathJax{\DeclareMathOperator{\Residue}{Res}}
\CustomizeMathJax{\newcommand{\principalvalue}{\mathcal{P}}}
\CustomizeMathJax{\let\pv\principalvalue}
\CustomizeMathJax{\newcommand{\PV}{\text{P.V.}}}
\CustomizeMathJax{\newcommand{\real}{\mathcal{R}}}
\CustomizeMathJax{\newcommand{\imaginary}{\mathcal{I}}}

% must be brace arguments
\CustomizeMathJax{\newcommand{\Re}[1]{\mathrm{Re}\left\lbrace#1\right\rbrace}}
\CustomizeMathJax{\newcommand{\Im}[1]{\mathrm{Im}\left\lbrace#1\right\rbrace}}

\CustomizeMathJax{\newcommand{\qqtext}[1]{\quad\mathrm{#1}\quad}}
\CustomizeMathJax{\let\qq\qqtext}
\CustomizeMathJax{\newcommand{\qqcomma}{\text{,}\quad}}
\CustomizeMathJax{\let\qc\qqcomma}
\CustomizeMathJax{\newcommand{\qcc}{\quad\text{c.c.}\quad}}
\CustomizeMathJax{\let\qcc}
\CustomizeMathJax{\newcommand{\qif}{\quad\text{if}\quad}}
\CustomizeMathJax{\newcommand{\qthen}{\quad\text{then}\quad}}
\CustomizeMathJax{\newcommand{\qelse}{\quad\text{else}\quad}}
\CustomizeMathJax{\newcommand{\qotherwise}{\quad\text{otherwise}\quad}}
\CustomizeMathJax{\newcommand{\qunless}{\quad\text{unless}\quad}}
\CustomizeMathJax{\newcommand{\qgiven}{\quad\text{given}\quad}}
\CustomizeMathJax{\newcommand{\qusing}{\quad\text{using}\quad}}
\CustomizeMathJax{\newcommand{\qassume}{\quad\text{assume}\quad}}
\CustomizeMathJax{\newcommand{\qsince}{\quad\text{since}\quad}}
\CustomizeMathJax{\newcommand{\qlet}{\quad\text{let}\quad}}
\CustomizeMathJax{\newcommand{\qfor}{\quad\text{for}\quad}}
\CustomizeMathJax{\newcommand{\qall}{\quad\text{all}\quad}}
\CustomizeMathJax{\newcommand{\qeven}{\quad\text{even}\quad}}
\CustomizeMathJax{\newcommand{\qodd}{\quad\text{odd}\quad}}
\CustomizeMathJax{\newcommand{\qinteger}{\quad\text{integer}\quad}}
\CustomizeMathJax{\newcommand{\qand}{\quad\text{and}\quad}}
\CustomizeMathJax{\newcommand{\qor}{\quad\text{or}\quad}}
\CustomizeMathJax{\newcommand{\qas}{\quad\text{as}\quad}}
\CustomizeMathJax{\newcommand{\qin}{\quad\text{in}\quad}}

\CustomizeMathJax{\newcommand{\differential}[1][]{\text{d}^{#1}}}
\CustomizeMathJax{\let\dd\differential}

\CustomizeMathJax{\newcommand{\derivative}[3][]{%
    \frac{\text{d}^{#1}#2}{\text{d}#3^{#1}}}%
}
\CustomizeMathJax{\let\dv\derivative}

\CustomizeMathJax{\newcommand{\partialderivative}[3][]{%
    \frac{\partial^{#1}#2}{\partial#3^{#1}}}%
}
\CustomizeMathJax{\let\pderivative\partialderivative}
\CustomizeMathJax{\let\pdv\partialderivative}

```

```
\CustomizeMathJax{\newcommand{\variation}{\delta}}
\CustomizeMathJax{\let\var\variation}

% Must provide two mandatory args.
% For the example in the manual with (E-TS), enclose the parens in braces
\CustomizeMathJax{\newcommand{\functionalderivative}[3][]{%
    \frac{\delta^{#1}#2}{\delta#3^{#1}}}%
}
\CustomizeMathJax{\let\fdv\functionalderivative}


% use \braket to contract
\CustomizeMathJax{\newcommand{\bra}[1]{\langle{#1}\rvert}}
\CustomizeMathJax{\newcommand{\ket}[1]{\lvert{#1}\rangle}}

% must have two args
\CustomizeMathJax{\newcommand{\innerproduct}[2]{%
    \left\langle{#1}\middle\vert{#2}\right\rangle}%
}
\CustomizeMathJax{\let\braket\innerproduct}
\CustomizeMathJax{\let\ip\innerproduct}

% must have two args
\CustomizeMathJax{\newcommand{\outerproduct}[2]{%
    \left\lvert{#1}\middle\rangle\!\middle\langle#2\right\rvert}%
}
\CustomizeMathJax{\let\dyad\outerproduct}
\CustomizeMathJax{\let\op\outerproduct}

% must have two args, unlike the MathJax version
\CustomizeMathJax{\newcommand{\expectationvalue}[2]{%
    \left\langle{#2}\middle\vert{#1}\middle\vert{#2}\right\rangle}%
}
\CustomizeMathJax{\let\expval\expectationvalue}
\CustomizeMathJax{\let\ev\expectationvalue}

\CustomizeMathJax{\newcommand{\matrixelement}[3]{%
    \left\langle{#1}\middle\vert{#2}\middle\vert{#3}\right\rangle}%
}
\CustomizeMathJax{\let\mel\matrixelement}

\CustomizeMathJax{\newcommand{\matrixquantity}[1]{\begin{matrix}#1\end{matrix}}}
\CustomizeMathJax{\let\mqty\matrixquantity}
\CustomizeMathJax{\newcommand{\pmqty}[1]{\begin{pmatrix}#1\end{pmatrix}}}
\CustomizeMathJax{\newcommand{\Pmqty}[1]{%
    \left\lgroup\begin{matrix}#1\end{matrix}\right\rgroup}%
}
\CustomizeMathJax{\newcommand{\bmqty}[1]{\begin{bmatrix}#1\end{bmatrix}}}
\CustomizeMathJax{\newcommand{\vmqty}[1]{\begin{vmatrix}#1\end{vmatrix}}}

\CustomizeMathJax{\newcommand{\smallmatrixquantity}[1]{%
    \begin{smallmatrix}#1\end{smallmatrix}}%
}
\CustomizeMathJax{\let\smqty\smallmatrixquantity}
\CustomizeMathJax{\newcommand{\spmqty}[1]{%
```

```
    \pqty{\begin{smallmatrix}#1\end{smallmatrix}}}%
}
\CustomizeMathJax{\newcommand{\sPmqty}[1]{%
    \left\lgroup\begin{smallmatrix}#1\end{smallmatrix}\right\rgroup}%
}
\CustomizeMathJax{\newcommand{\sbmqty}[1]{%
    \bqty{\begin{smallmatrix}#1\end{smallmatrix}}}%
}
\CustomizeMathJax{\newcommand{\svmqty}[1]{%
    \vqty{\begin{smallmatrix}#1\end{smallmatrix}}}%
}

\CustomizeMathJax{\let\matrixdeterminant\vmqty}
\CustomizeMathJax{\let\mdet\vmqty}
\CustomizeMathJax{\let\smdet\svmqty}

\CustomizeMathJax{\newcommand{\identitymatrix}[1]{(\text{imat}\{#1\})}}
\CustomizeMathJax{\let\imat\identitymatrix}

\CustomizeMathJax{\newcommand{\xmatrix}[3]{(\text{xmat}\{\}\{#2\}\{#3\})}}
\CustomizeMathJax{\let\xmat\xmatrix}

\CustomizeMathJax{\newcommand{\zeromatrix}[2]{(\text{zmat}\{#1\}\{#2\})}}
\CustomizeMathJax{\let\zmat\zeromatrix}

\CustomizeMathJax{\newcommand{\paulimatrix}[1]{(\text{pmat}\{#1\})}}
\CustomizeMathJax{\let\pmat\paulimatrix}

\CustomizeMathJax{\newcommand{\diagonalmatrix}[2][]{%
    \left(\text{dmat}\right)}%
}
\CustomizeMathJax{\let\dmat\diagonalmatrix}

\CustomizeMathJax{\newcommand{\antidiagonalmatrix}[2][]{%
    \left(\text{admat}\right)}%
}
\CustomizeMathJax{\let\admat\antidiagonalmatrix}
\end{warpMathJax}
```

File 296 **lwarp-pifont.sty**

§ 396 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 297 **lwarp-placeins.sty**

§ 397 Package placeins

(*Emulates or patches code by* Donald Arseneau.)

Pkg placeins placeins is ignored.

Discard all options for lwarp-placeins:

for HTML output:

```
\LWR@ProvidesPackageDrop{placeins}[2005/04/18]

\newcommand*{\FloatBarrier}{}
```

File 298 **lwarp-plarydshln.sty**

§ 398 Package plarydshln

Pkg plarydshln plarydshln is emulated by lwarp-arydshln.

for HTML output:

```
\LWR@ProvidesPackageDrop{plarydshln}[2018/10/20]

\LWR@origRequirePackage{lwarp-arydshln}
```

File 299 **lwarp-plext.sty**

§ 399 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 300 **lwarp-plextarydshln.sty**

§ 400 Package

plextarydshln

Pkg plextarydshln

plextarydshln is emulated by lwarp-arydshln.

for HTML output:

```
\LWR@ProvidesPackageDrop{plextarydshln}[2018/10/20]

\LWR@origRequirePackage{lwarp-arydshln}
```

File 301 **lwarp-plextcolortbl.sty**

§ 401 Package plextcolortbl

Pkg plextcolortbl plextcolortbl is emulated by lwarp-colortbl.

for HTML output:

```
\LWR@ProvidesPackageDrop{plextcolortbl}[2018/09/19]
\LWR@origRequirePackage{lwarp-colortbl}
```

File 302 **lwarp-prelim2e.sty**

§ 402 Package prelim2e

(*Emulates or patches code by* Martin Schröder.)

Pkg prelim2e prelim2e is ignored.

for HTML output: Discard all options for lwarp-prelim2e:

```
\LWR@ProvidesPackageDrop{prelim2e}[2009/05/29]
\newcommand{\PrelimText}{}
\newcommand{\PrelimTextStyle}{}
\newcommand{\PrelimWords}{}
```

File 303 **lwarp-prettyref.sty**

§ 403 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 304 **lwarp-preview.sty**

§ 404 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 305 **lwarp-psfrag.sty**

§ 405 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-~\PackageDiagramAltText]%
}

\AfterEndEnvironment{psfrags}{\end{lateximage}}
```

File 306 **lwarp-psfragx.sty**

§ 406 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-~\PackageDiagramAltText]%
    \mbox{\pfx@overpix{#1}{#2}\endpfx@overpix}%
    \end{lateximage}%
}

\def\@@@overpix[#1]<#2>[#3]#4{%
    \begin{lateximage}[-psfragx-~\PackageDiagramAltText]%
    \pfx@overpix{#1,ovpfgd={#2},ovpbgd={#3}}{#4}%
}

\def\endoverpix{%
    \endpfx@overpix%
    \end{lateximage}%
}
```

File 307 **lwarp-pst-eps.sty**

§ 407 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 308 **lwarp-pstool.sty**

§ 408 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-~\PackageDiagramAltText]%
    \includegraphics{#2.pdf}%
    \end{lateximage}%
}
\LetLtxMacro\pstool@neverprocess\pstool@alwaysprocess
\LetLtxMacro\pstool@maybeprocess\pstool@alwaysprocess

\renewcommand\pstool@@psfragfig[4]{%
    \begin{lateximage}[-pstool-~\PackageDiagramAltText]%
    \includegraphics{#2.pdf}%
    \end{lateximage}%
}
```

File 309 **lwarp-pstricks.sty**

§ 409 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 310 **lwarp-pxatbegshi.sty**

§ 410 Package pxatbegshi

Pkg pxatbegshi pxatbegshi is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pxatbegshi}[2017/11/04]

\LWR@origRequirePackage{lwarp-atbegshi}
```

File 311 **lwarp-pxeveryshi.sty**

§ 411 Package pxeveryshi

Pkg pxeveryshi pxeveryshi is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pxeveryshi}[2012/05/19]
\LWR@origRequirePackage{lwarp-everyshi}
```

File 312 **lwarp-pxftnright.sty**

§ 412 Package pxftnright

Pkg pxftnright pxftnright is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pxftnright}[2017/02/28]
\LWR@origRequirePackage{lwarp-ftnright}
```

File 313 **lwarp-pxjahyper.sty**

§ 413 Package pxjahyper

Pkg pxjahyper pxjahyper is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{pxjahyper}[2018/07/15]
```

File 314 **lwarp-quotchap.sty**

§ 414 Package quotchap

(*Emulates or patches code by* Karsten Tinnefeld, Jan Klever.)

Pkg quotchap quotchap is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{quotchap}[2019/07/09]

\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]{%
    \LWR@stoppars%
    \begin{BlockClass}{qauthor}%
    {#1}%
    \end{BlockClass}%
    \LWR@startpars%
}
```

Fonts are ignored. Use CSS.

```
\newcommand{\qsetcnfont}[1]{}
\providecommand*{\quotefont}{}
\providecommand*{\qauthorfont}{}
```

File 315 **lwarp-quoting.sty**

§ 415 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@stoppars}
    {}
    {\LWR@patcherror{imakeidx}{endquoting}}
```

File 316 **lwarp-ragged2e.sty**

§ 416 Package ragged2e

(*Emulates or patches code by* Martin Schröder.)

Pkg ragged2e ragged2e is emulated.

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 317 **lwarp-realscripts.sty**

§ 417 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 318 **lwarp-refcheck.sty**

§ 418 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 319 **lwarp-register.sty**

§ 419 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-~\PackageDiagramAltText]%
    }
    {}
    {\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-~\PackageDiagramAltText]%
    }
    {}
    {\LWR@patcherror{register}{register*}}

\expandafter\xpatchcmd\csname endregister*\endcsname
    {\leftskip}
    {%
        \end{lateximage}\end{center}%
        \leftskip%
    }%
    {}
    {\LWR@patcherror{register}{endregister*}}

\setlength{\regWidth}{5in}
```

File 320 **lwarp-relsize.sty**

§ 420 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 321 **lwarp-repeatindex.sty**

§ 421 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 322 **lwarp-resizegather.sty**

§ 422 Package resizegather

Pkg resizegather resizegather is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{resizegather}[2016/05/16]

\newcommand*{\resizegathersetup}[1]{}
```

File 323 **lwarp-rmpage.sty**

§ 423 Package rmpage

Pkg rmpage rmpage is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{rmpage}[1997/09/29]
```

File 324 **lwarp-romanbar.sty**

§ 424 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 325 **lwarp-romanbarpagenumber.sty**

§ 425 Package romanbarpagenumber

Pkg romanbarpagenumber romanbarpagenumber is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{romanbarpagenumber}[2015/02/06]
```

File 326 **lwarp-rotating.sty**

§ 426 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]
\RequirePackage{graphicx}

\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 327 **lwarp-rotfloat.sty**

§ 427 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 328 **lwarp-rviewport.sty**

§ 428 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 329 **lwarp-savetrees.sty**

§ 429 Package savetrees

Pkg savetrees

savetrees is ignored.

for HTML output:

Discard all options for lwarp-savetrees:

```
\LWR@ProvidesPackageDrop{savetrees}[2016/04/13]
```

File 330 **lwarp-scalefnt.sty**

§ 430 Package scalefnt

(*Emulates or patches code by* D. Carlisle.)

Pkg scalefnt scalefnt is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{scalefnt}

\DeclareRobustCommand\scalefont[1]{}
```

File 331 **lwarp-schemata.sty**

§ 431 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 332 **lwarp-scrextend.sty**

§ 432 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@koma@orig@maketitle\maketitle
    \DeclareDocumentCommand{\maketitle}{o}{\LWR@koma@orig@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}
    {
        \LWR@stoppars%
        \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}
```

```
{
\LWR@stoppars%
\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}\LWR@startpars}
```

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 333 **lwarp-scrhack.sty**

§ 433 Package scrhack

Pkg scrhack scrhack is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{scrhack}[2018/03/30]
```

File 334 **lwarp-scrlayer.sty**

§ 434 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 335 **lwarp-scrlayer-notecolumn.sty**

§ 435 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 336 **lwarp-scrlayer-scrpage.sty**

§ 436 Package scrlayer-scrpage

(*Emulates or patches code by* Markus Kohm.)

Pkg scrlayer-scrpage scrlayer-scrpage is ignored.

⚠ 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 337 **lwarp-scrpage2.sty**

§ 437 Package

scrpage2

(*Emulates or patches code by* Markus Kohm.)

Pkg scrpage2 scrpage2 is ignored.

⚠ 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 338 **lwarp-section.sty**

§ 438 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 339 **lwarp-sectionbreak.sty**

§ 439 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 340 **lwarp-sectsty.sty**

§ 440 Package sectsty

(*Emulates or patches code by* Rowland McDonnell.)

Pkg sectsty sectsty is ignored.

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 341 **lwarp-semantic-markup.sty**

§ 441 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 342 **lwarp-setspace.sty**

§ 442 Package

setspace

(*Emulates or patches code by* Robin Fairbairns.)

Pkg setspace setspace is emulated.

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}{}

```

```
\newenvironment*{singlespace}
{
\LWR@forcenewpage
\BlockClass{singlespace}
}
{\endBlockClass}

\newenvironment*{singlespace*}
{
\LWR@forcenewpage
\BlockClass{singlespace}
}
{\endBlockClass}

\newenvironment*{spacing}[1]{

}{

}

\newenvironment*{onehalfspace}
{
\LWR@forcenewpage
\BlockClass{onehalfspace}
}
{\endBlockClass}

\newenvironment*{doublespace}
{
\LWR@forcenewpage
\BlockClass{doublespace}
}
{\endBlockClass}
```

File 343 **lwarp-shadow.sty**

§ 443 Package

shadow

(*Emulates or patches code by* Mauro Orlandini.)

Pkg shadow shadow is emulated.

for HTML output: Discard all options for lwarp-shadow:

```
\LWR@ProvidesPackageDrop{shadow}[2003/02/19]

\newdimen\sboxsep
\newdimen\sboxrule
\newdimen\sdim

\newcommand{\shabox}[1]{%
\InlineClass{shabox}{#1}%
}
```

File 344 **lwarp-shapepar.sty**

§ 444 Package shapepar

(*Emulates or patches code by* Donald Arseneau.)

Pkg shapepar shapepar is patched for use by lwarp. Shapes appear in print mode, as well as inside a `lateximage`, but are ignored for HTML.

for HTML output:

```
\LWR@ProvidesPackagePass{shapepar}[2013/03/26]

\newcommand*{\LWR@HTML@shapepar}[2][]{}
\LWR@formatted{shapepar}

\NewDocumentCommand{\LWR@HTML@cutout}{m d()}{}
\LWR@formatted{cutout}
```

File 345 **lwarp-showidx.sty**

§ 445 Package showidx

Pkg showidx showidx is ignored.

for HTML output: Discard all options for lwarp-showidx:

```
\LWR@ProvidesPackageDrop{showidx}[2014/09/29]
```

`\@wrindex` is redefined `\AtBeginDocument` by the lwarp core.

File 346 **lwarp-showkeys.sty**

§ 446 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:

```
\LWR@ProvidesPackageDrop{showkeys}[2014/10/28]

\NewDocumentCommand{\showkeys}{s}{}
```

File 347 **lwarp-showtags.sty**

§ 447 Package showtags

Pkg showtags showtags is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{showtags}% no version is given
\newcommand{\thecitetag}[1]{}
```

File 348 **lwarp-sidecap.sty**

§ 448 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.

```
\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.

```
\NewDocumentEnvironment{SCtable}{soo}
{\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 349 **lwarp-sidenotes.sty**

§ 449 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 350 **lwarp-SIunits.sty**

§ 450 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 according to `\MathImageAltText`, 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 351 **lwarp-siunitx.sty**

§ 451 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.

⚠ 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`.

⚠ MathJax

For math mode with SVG display, the original siunitx code is used while generating the SVG image. For text mode, lwarp uses an emulation which provides a very effective HTML interpretation of siunitx. For math expressions while using MathJax, a limited emulation is used. Most functions work reasonably well, but many options cannot be emulated. siunitx macros with more than one optional value cannot absorb the second optional value, and complicated parsing such as for `\ang` is not supported. The result usually looks fine, and otherwise is enough to get the meaning across.

lwarp's MathJax emulation for siunitx is meant to be a stop-gap measure until an extension is included in MathJax. As of this writing, the third-party siunitx extension for MathJax is not currently hosted at any public CDN, thus siunitx is not usable with this extension unless a local copy of this extension is created first. See `\MathJaxFilename` to select a custom MathJax script, but lwarp's emulation would have to be diabled as well.

Document modifications required for MathJax:

custom units

- Custom units may be added with `\CustomizeMathJax`. See the lwarp-siunitx code for examples.

⚠ unit spacing

- Units work better using ~ between units instead of using periods.

⚠ `\square, \cubic`

- To square or cube compound units, enclose the following compound units in braces:

  ```
  \cubic{\centi\meter}
  ```

  Single units do not require braces.

Also see MathJax option, section 8.7.4.

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}
    }
    {
```

For parse-numbers=false:

```
        \__siunitx_unit_output_pre_print:
        \begingroup%    lwarp
            \boolfalse{mathjax}%    lwarp
%       \__siunitx_print:nn { number } { \ensuremath {#1} }
            \LWR@subsingledollar{% lwarp
                \textbackslash( \LWR@HTMLsanitize{#1} \textbackslash)% lwarp
            }{siunitx}{%
                \__siunitx_print:nn { number } {%
                    \LWR@origensuredmath{#1}%
                }%
            }% lwarp
        \endgroup%    lwarp
        \__siunitx_unit_output_print:
    }
    \group_end:% lwarp
}
```

For parse-numbers=false:

```
\cs_set_protected:Npn \__siunitx_range_numbers_aux:n #1
  {
    \bool_if:NTF \l__siunitx_number_parse_bool
      {
```

```
      \tl_clear:N \l__siunitx_number_out_tl
      \tl_clear:N \l__siunitx_number_out_saved_tl
      \bool_set_false:N \l__siunitx_number_compound_bool
      \__siunitx_number_output_parse:n {#1}
      \bool_if:NT \l__siunitx_number_compound_bool
        { \msg_error:nnx { siunitx } { multi-part-range } {#1} }
    }
    {
      \__siunitx_unit_output_pre_print:
      \begingroup%    lwarp
          \boolfalse{mathjax}%    lwarp
%         \__siunitx_print:nn { number } {#1}
              \LWR@subsingledollar{% lwarp
                \textbackslash( \LWR@HTMLsanitize{#1} \textbackslash)% lwarp
              }{siunitx}{%
                  \__siunitx_print:nn { number } {
                      \LWR@origensuredmath{#1}%
                  } % lwarp
              }% lwarp
      \endgroup%    lwarp
      \__siunitx_unit_output_print:
    }
}
```

For `parse-numbers=false`:

```
\cs_set_protected:Npn \__siunitx_angle_print_direct_aux:nn #1#2 {
  \tl_if_empty:nF {#1}
    {
      \tl_set:Nn \l__siunitx_unit_tl {#2}
      \begingroup%    lwarp
          \boolfalse{mathjax}%    lwarp
%         \__siunitx_print:nn { number } {#1}
              \LWR@subsingledollar{% lwarp
                \textbackslash( \LWR@HTMLsanitize{#1} \textbackslash)% lwarp
              }{siunitx}{%
                  \__siunitx_print:nn { number } {
                      \LWR@origensuredmath{#1}%
                  } % lwarp
              }% lwarp
      \endgroup%    lwarp
      \__siunitx_unit_output_print:
    }
}
%
```

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 }
}


\cs_undefine:N \__siunitx_textsuperscript:n
\cs_new_protected:Npn \__siunitx_textsuperscript:n #1 {\textsuperscript{#1}}

\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
```

For MathJax. (The following runs much faster as separate `\CusomizeMathJax` calls instead of one single call.)

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\ang}[2][]{(\mathrm{#2})\degree}}
\CustomizeMathJax{\newcommand{\num}[2][]{\mathrm{#2}}}
\CustomizeMathJax{\newcommand{\si}[2][]{\mathrm{#2}}}
%
% \SI[opt]{num}[preunit]{unit}
\CustomizeMathJax{\newcommand{\LWRSI}[2][]{\mathrm{#1\LWRSInumber\,#2}}}
\CustomizeMathJax{\newcommand{\SI}[2][]{\def\LWRSInumber{#2}\LWRSI}}
%
\CustomizeMathJax{\newcommand{\numlist}[2][]{\mathrm{#2}}}
\CustomizeMathJax{\newcommand{\numrange}[3][]{\mathrm{#2~-- #3}}}
\CustomizeMathJax{\newcommand{\SIlist}[3][]{\mathrm{#2\,#3}}}
\CustomizeMathJax{\newcommand{\SIrange}[4][]{\mathrm{#2\,#4~-- #3\,#4}}}
\CustomizeMathJax{\newcommand{\tablenum}[2][]{\mathrm{#2}}}
%
\CustomizeMathJax{\newcommand{\ampere}{\mathrm{A}}}
\CustomizeMathJax{\newcommand{\candela}{\mathrm{cd}}}
\CustomizeMathJax{\newcommand{\kelvin}{\mathrm{K}}}
\CustomizeMathJax{\newcommand{\kilogram}{\mathrm{kg}}}
\CustomizeMathJax{\newcommand{\metre}{\mathrm{m}}}
\CustomizeMathJax{\newcommand{\mole}{\mathrm{mol}}}
\CustomizeMathJax{\newcommand{\second}{\mathrm{s}}}
%
\CustomizeMathJax{\newcommand{\becquerel}{\mathrm{Bq}}}
\CustomizeMathJax{\newcommand{\degreeCelsius}{\unicode{x2103}}}
\CustomizeMathJax{\newcommand{\coulomb}{\mathrm{C}}}
\CustomizeMathJax{\newcommand{\farad}{\mathrm{F}}}
\CustomizeMathJax{\newcommand{\gray}{\mathrm{Gy}}}
\CustomizeMathJax{\newcommand{\hertz}{\mathrm{Hz}}}
\CustomizeMathJax{\newcommand{\henry}{\mathrm{H}}}
\CustomizeMathJax{\newcommand{\joule}{\mathrm{J}}}
```

```
\CustomizeMathJax{\newcommand{\katal}{\mathrm{kat}}}
\CustomizeMathJax{\newcommand{\lumen}{\mathrm{lm}}}
\CustomizeMathJax{\newcommand{\lux}{\mathrm{lx}}}
\CustomizeMathJax{\newcommand{\newton}{\mathrm{N}}}
\CustomizeMathJax{\newcommand{\ohm}{\mathrm{\Omega}}}
\CustomizeMathJax{\newcommand{\pascal}{\mathrm{Pa}}}
\CustomizeMathJax{\newcommand{\radian}{\mathrm{rad}}}
\CustomizeMathJax{\newcommand{\siemens}{\mathrm{S}}}
\CustomizeMathJax{\newcommand{\sievert}{\mathrm{Sv}}}
\CustomizeMathJax{\newcommand{\steradian}{\mathrm{sr}}}
\CustomizeMathJax{\newcommand{\tesla}{\mathrm{T}}}
\CustomizeMathJax{\newcommand{\volt}{\mathrm{V}}}
\CustomizeMathJax{\newcommand{\watt}{\mathrm{W}}}
\CustomizeMathJax{\newcommand{\weber}{\mathrm{Wb}}}
\CustomizeMathJax{\newcommand{\day}{\mathrm{d}}}
\CustomizeMathJax{\newcommand{\degree}{\mathrm{^\circ}}}
\CustomizeMathJax{\newcommand{\hectare}{\mathrm{ha}}}
\CustomizeMathJax{\newcommand{\hour}{\mathrm{h}}}
\CustomizeMathJax{\newcommand{\litre}{\mathrm{l}}}
\CustomizeMathJax{\newcommand{\liter}{\mathrm{L}}}
\CustomizeMathJax{\newcommand{\arcminute}{^\prime}}
\CustomizeMathJax{\newcommand{\minute}{\mathrm{min}}}
\CustomizeMathJax{\newcommand{\arcsecond}{^{\prime\prime}}}
\CustomizeMathJax{\newcommand{\tonne}{\mathrm{t}}}
\CustomizeMathJax{\newcommand{\astronomicalunit}{au}}
\CustomizeMathJax{\newcommand{\atomicmassunit}{u}}
\CustomizeMathJax{\newcommand{\bohr}{\mathit{a}_0}}
\CustomizeMathJax{\newcommand{\clight}{\mathit{c}_0}}
\CustomizeMathJax{\newcommand{\dalton}{\mathrm{D}_\mathrm{a}}}
\CustomizeMathJax{\newcommand{\electronmass}{\mathit{m}_{\mathrm{e}}}}
\CustomizeMathJax{\newcommand{\electronvolt}{\mathrm{eV}}}
\CustomizeMathJax{\newcommand{\elementarycharge}{\mathit{e}}}
\CustomizeMathJax{\newcommand{\hartree}{\mathit{E}_{\mathrm{h}}}}
\CustomizeMathJax{\newcommand{\planckbar}{\mathit{\unicode{x0127}}}}
\CustomizeMathJax{\newcommand{\angstrom}{\mathrm{\unicode{x00C5}}}}
\CustomizeMathJax{\let\LWRorigbar\bar}
\CustomizeMathJax{\newcommand{\bar}{\mathrm{bar}}}
\CustomizeMathJax{\newcommand{\barn}{\mathrm{b}}}
\CustomizeMathJax{\newcommand{\bel}{\mathrm{B}}}
\CustomizeMathJax{\newcommand{\decibel}{\mathrm{dB}}}
\CustomizeMathJax{\newcommand{\knot}{\mathrm{kn}}}
\CustomizeMathJax{\newcommand{\mmHg}{\mathrm{mmHg}}}
\CustomizeMathJax{\newcommand{\nauticalmile}{\mathrm{M}}}
\CustomizeMathJax{\newcommand{\neper}{\mathrm{Np}}}
%
\CustomizeMathJax{\newcommand{\yocto}{\mathrm{y}}}
\CustomizeMathJax{\newcommand{\zepto}{\mathrm{z}}}
\CustomizeMathJax{\newcommand{\atto}{\mathrm{a}}}
\CustomizeMathJax{\newcommand{\femto}{\mathrm{f}}}
\CustomizeMathJax{\newcommand{\pico}{\mathrm{p}}}
\CustomizeMathJax{\newcommand{\nano}{\mathrm{n}}}
\CustomizeMathJax{\newcommand{\micro}{\mathrm{\unicode{x00B5}}}}
\CustomizeMathJax{\newcommand{\milli}{\mathrm{m}}}
\CustomizeMathJax{\newcommand{\centi}{\mathrm{c}}}
\CustomizeMathJax{\newcommand{\deci}{\mathrm{d}}}
```

```
\CustomizeMathJax{\newcommand{\deca}{\mathrm{da}}}
\CustomizeMathJax{\newcommand{\hecto}{\mathrm{h}}}
\CustomizeMathJax{\newcommand{\kilo}{\mathrm{k}}}
\CustomizeMathJax{\newcommand{\mega}{\mathrm{M}}}
\CustomizeMathJax{\newcommand{\giga}{\mathrm{G}}}
\CustomizeMathJax{\newcommand{\tera}{\mathrm{T}}}
\CustomizeMathJax{\newcommand{\peta}{\mathrm{P}}}
\CustomizeMathJax{\newcommand{\exa}{\mathrm{E}}}
\CustomizeMathJax{\newcommand{\zetta}{\mathrm{Z}}}
\CustomizeMathJax{\newcommand{\yotta}{\mathrm{Y}}}
%
\CustomizeMathJax{\newcommand{\percent}{\mathrm{\%}}}
%
\CustomizeMathJax{\newcommand{\meter}{\mathrm{m}}}
\CustomizeMathJax{\newcommand{\metre}{\mathrm{m}}}
%
\CustomizeMathJax{\newcommand{\gram}{\mathrm{g}}}
\CustomizeMathJax{\newcommand{\kg}{\kilo\gram}}
\CustomizeMathJax{\newcommand{\of}[1]{_{\mathrm{#1}}}}
\CustomizeMathJax{\newcommand{\squared}{^2}}
\CustomizeMathJax{\newcommand{\square}[1]{\mathrm{#1}^2}}
\CustomizeMathJax{\newcommand{\cubed}{^3}}
\CustomizeMathJax{\newcommand{\cubic}[1]{\mathrm{#1}^3}}
\CustomizeMathJax{\newcommand{\per}{/}}
\CustomizeMathJax{\newcommand{\celsius}{\unicode{x2103}}}
%
\CustomizeMathJax{\newcommand{\fg}{\femto\gram}}
\CustomizeMathJax{\newcommand{\pg}{\pico\gram}}
\CustomizeMathJax{\newcommand{\ng}{\nano\gram}}
\CustomizeMathJax{\newcommand{\ug}{\micro\gram}}
\CustomizeMathJax{\newcommand{\mg}{\milli\gram}}
\CustomizeMathJax{\newcommand{\g}{\gram}}
\CustomizeMathJax{\newcommand{\kg}{\kilo\gram}}
%
\CustomizeMathJax{\newcommand{\amu}{\mathrm{u}}}
%
\CustomizeMathJax{\newcommand{\pm}{\pico\metre}}
\CustomizeMathJax{\newcommand{\nm}{\nano\metre}}
\CustomizeMathJax{\newcommand{\um}{\micro\metre}}
\CustomizeMathJax{\newcommand{\mm}{\milli\metre}}
\CustomizeMathJax{\newcommand{\cm}{\centi\metre}}
\CustomizeMathJax{\newcommand{\dm}{\deci\metre}}
\CustomizeMathJax{\newcommand{\m}{\metre}}
\CustomizeMathJax{\newcommand{\km}{\kilo\metre}}
%
\CustomizeMathJax{\newcommand{\as}{\atto\second}}
\CustomizeMathJax{\newcommand{\fs}{\femto\second}}
\CustomizeMathJax{\newcommand{\ps}{\pico\second}}
\CustomizeMathJax{\newcommand{\ns}{\nano\second}}
\CustomizeMathJax{\newcommand{\us}{\micro\second}}
\CustomizeMathJax{\newcommand{\ms}{\milli\second}}
\CustomizeMathJax{\newcommand{\s}{\second}}
%
\CustomizeMathJax{\newcommand{\fmol}{\femto\mol}}
\CustomizeMathJax{\newcommand{\pmol}{\pico\mol}}
```

```
\CustomizeMathJax{\newcommand{\nmol}{\nano\mol}}
\CustomizeMathJax{\newcommand{\umol}{\micro\mol}}
\CustomizeMathJax{\newcommand{\mmol}{\milli\mol}}
\CustomizeMathJax{\newcommand{\mol}{\mol}}
\CustomizeMathJax{\newcommand{\kmol}{\kilo\mol}}
%
\CustomizeMathJax{\newcommand{\pA}{\pico\ampere}}
\CustomizeMathJax{\newcommand{\nA}{\nano\ampere}}
\CustomizeMathJax{\newcommand{\uA}{\micro\ampere}}
\CustomizeMathJax{\newcommand{\mA}{\milli\ampere}}
\CustomizeMathJax{\newcommand{\A}{\ampere}}
\CustomizeMathJax{\newcommand{\kA}{\kilo\ampere}}
%
\CustomizeMathJax{\newcommand{\ul}{\micro\litre}}
\CustomizeMathJax{\newcommand{\ml}{\milli\litre}}
\CustomizeMathJax{\newcommand{\l}{\litre}}
\CustomizeMathJax{\newcommand{\hl}{\hecto\litre}}
\CustomizeMathJax{\newcommand{\uL}{\micro\liter}}
\CustomizeMathJax{\newcommand{\mL}{\milli\liter}}
\CustomizeMathJax{\newcommand{\L}{\liter}}
\CustomizeMathJax{\newcommand{\hL}{\hecto\liter}}
%
\CustomizeMathJax{\newcommand{\mHz}{\milli\hertz}}
\CustomizeMathJax{\newcommand{\Hz}{\hertz}}
\CustomizeMathJax{\newcommand{\kHz}{\kilo\hertz}}
\CustomizeMathJax{\newcommand{\MHz}{\mega\hertz}}
\CustomizeMathJax{\newcommand{\GHz}{\giga\hertz}}
\CustomizeMathJax{\newcommand{\THz}{\tera\hertz}}
%
\CustomizeMathJax{\newcommand{\mN}{\milli\newton}}
\CustomizeMathJax{\newcommand{\N}{\newton}}
\CustomizeMathJax{\newcommand{\kN}{\kilo\newton}}
\CustomizeMathJax{\newcommand{\MN}{\mega\newton}}
%
\CustomizeMathJax{\newcommand{\Pa}{\pascal}}
\CustomizeMathJax{\newcommand{\kPa}{\kilo\pascal}}
\CustomizeMathJax{\newcommand{\MPa}{\mega\pascal}}
\CustomizeMathJax{\newcommand{\GPa}{\giga\pascal}}
%
\CustomizeMathJax{\newcommand{\mohm}{\milli\ohm}}
\CustomizeMathJax{\newcommand{\kohm}{\kilo\ohm}}
\CustomizeMathJax{\newcommand{\Mohm}{\mega\ohm}}
%
\CustomizeMathJax{\newcommand{\pV}{\pico\volt}}
\CustomizeMathJax{\newcommand{\nV}{\nano\volt}}
\CustomizeMathJax{\newcommand{\uV}{\micro\volt}}
\CustomizeMathJax{\newcommand{\mV}{\milli\volt}}
\CustomizeMathJax{\newcommand{\V}{\volt}}
\CustomizeMathJax{\newcommand{\kV}{\kilo\volt}}
%
\CustomizeMathJax{\newcommand{\W}{\watt}}
\CustomizeMathJax{\newcommand{\uW}{\micro\watt}}
\CustomizeMathJax{\newcommand{\mW}{\milli\watt}}
\CustomizeMathJax{\newcommand{\kW}{\kilo\watt}}
\CustomizeMathJax{\newcommand{\MW}{\mega\watt}}
```

```
\CustomizeMathJax{\newcommand{\GW}{\giga\watt}}
%
\CustomizeMathJax{\newcommand{\J}{\joule}}
\CustomizeMathJax{\newcommand{\uJ}{\micro\joule}}
\CustomizeMathJax{\newcommand{\mJ}{\milli\joule}}
\CustomizeMathJax{\newcommand{\kJ}{\kilo\joule}}
%
\CustomizeMathJax{\newcommand{\eV}{\electronvolt}}
\CustomizeMathJax{\newcommand{\meV}{\milli\electronvolt}}
\CustomizeMathJax{\newcommand{\keV}{\kilo\electronvolt}}
\CustomizeMathJax{\newcommand{\MeV}{\mega\electronvolt}}
\CustomizeMathJax{\newcommand{\GeV}{\giga\electronvolt}}
\CustomizeMathJax{\newcommand{\TeV}{\tera\electronvolt}}
%
\CustomizeMathJax{\newcommand{\kWh}{\kilo\watt\hour}}
%
\CustomizeMathJax{\newcommand{\F}{\farad}}
\CustomizeMathJax{\newcommand{\fF}{\femto\farad}}
\CustomizeMathJax{\newcommand{\pF}{\pico\farad}}
%
\CustomizeMathJax{\newcommand{\K}{\mathrm{K}}}
%
\CustomizeMathJax{\newcommand{\dB}{\mathrm{dB}}}
%
\CustomizeMathJax{\newcommand{\kibi}{\mathrm{Ki}}}
\CustomizeMathJax{\newcommand{\mebi}{\mathrm{Mi}}}
\CustomizeMathJax{\newcommand{\gibi}{\mathrm{Gi}}}
\CustomizeMathJax{\newcommand{\tebi}{\mathrm{Ti}}}
\CustomizeMathJax{\newcommand{\pebi}{\mathrm{Pi}}}
\CustomizeMathJax{\newcommand{\exbi}{\mathrm{Ei}}}
\CustomizeMathJax{\newcommand{\zebi}{\mathrm{Zi}}}
\CustomizeMathJax{\newcommand{\yobi}{\mathrm{Yi}}}
\end{warpMathJax}
```

File 352 **lwarp-slantsc.sty**

§ 452 Package slantsc

(*Emulates or patches code by* Harald Harders.)

Pkg slantsc slantsc is emulated for HTML, and used as-is for print output.

for HTML output:

```
\LWR@ProvidesPackagePass{slantsc}[2012/01/01]

\newcommand*{\LWR@HTML@noscshape}{}
\LWR@formatted{noscshape}

\newcommand*{\LWR@null@noscshape}{}

\appto{\LWR@nullfonts}{%
\LetLtxMacro\noscshape\LWR@null@noscshape%
}
```

File 353 **lwarp-slashed.sty**

§ 453 Package slashed

(*Emulates or patches code by* David Carlisle.)

Pkg slashed slashed works as-s for HTML SVG math. For MathJax, emulation is provided.

for HTML output:

```
\LWR@ProvidesPackagePass{slashed}[1997/01/16]

\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\slashed}[1]{\cancel{#1}}}
\end{warpMathJax}
```

File 354 **lwarp-soul.sty**

§ 454 Package soul

(*Emulates or patches code by* Melchior FRANZ.)

Pkg soul soul is 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 355 **lwarp-soulpos.sty**

§ 455 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 356 **lwarp-soulutf8.sty**

§ 456 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]
\RequirePackage{soul}
```

File 357 **lwarp-splitidx.sty**

§ 457 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 555.17 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 358 **lwarp-srcltx.sty**

§ 458 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 359 **lwarp-srctex.sty**

§ 459 Package srctex

Pkg srctex srctex is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{srctex}[2006/11/12]
\LWR@origRequirePackage{lwarp-srcltx}
```

File 360 **lwarp-stabular.sty**

§ 460 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 361 **lwarp-stackengine.sty**

§ 461 Package stackengine

(*Emulates or patches code by* Steven B. Segletes.)

Pkg stackengine stackengine is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{stackengine}[2017/02/13]
```

The original version is neccessary for the patched \@stack and \stackanchor, where nesting lateximages does not work:

```
\LetLtxMacro\LWR@orig@stackengine\stackengine

\renewcommand*{\stackengine}[8]{%
    \ifstrequal{#4}{O}%
        {\begin{lateximage}[\ImageAltText]}%
        {\begin{lateximage}[\ImageAltText][][vertical-align:top]}%
    \LWR@orig@stackengine{#1}{#2}{#3}{#4}{#5}{#6}{#7}{#8}%
    \end{lateximage}%
}
```

\@stack uses a lateximage with a vertical alignment:

```
\LetLtxMacro\LWR@orig@@stack\@stack

\xpatchcmd{\LWR@orig@@stack}{\stackengine}{\LWR@orig@stackengine}
    {}
    {\LWR@patcherror{stackengine}{LWR@orig@@stack}}

\renewcommand*{\@stack}[4]{%
    \ifstrequal{#3}{O}%
        {\begin{lateximage}[\ImageAltText]}%
        {\begin{lateximage}[\ImageAltText][][vertical-align:top]}%
    \LWR@orig@@stack{#1}{#2}{#3}{#4}%
    \end{lateximage}%
}
```

The lapping macros are disabled for HTML:

```
\newcommand*\LWR@HTML@@stacklap[4]{#3}
\LWR@formatted{@stacklap}
```

\stackanchor is patched for two instances of \stackengine. A lateximage with vertical alignment is used.

```
\xpatchcmd{\stackanchor}{\stackengine}{\LWR@orig@stackengine}
    {}
    {\LWR@patcherror{stackengine}{stackanchor patch 1}}

\xpatchcmd{\stackanchor}{\stackengine}{\LWR@orig@stackengine}
    {}
    {\LWR@patcherror{stackengine}{stackanchor patch 2}}

\xpretocmd{\stackanchor}
    {\begin{lateximage}[\ImageAltText][][vertical-align:middle]}
    {}
    {\LWR@patcherror{stackengine}{stackanchor pre}}

\xapptocmd{\stackanchor}{\end{lateximage}}
    {}
    {\LWR@patcherror{stackengine}{stackanchor app}}
```

\Centerstack is simply placed inside a lateximage with a vertical alignment:

```
\xpretocmd{\Centerstack}
    {\begin{lateximage}[\ImageAltText][][vertical-align:middle]}
    {}
    {\LWR@patcherror{stackengine}{Centerstack pre}}

\xapptocmd{\Centerstack}{\end{lateximage}}
    {}
    {\LWR@patcherror{stackengine}{Centerstack app}}
```

\savestack reverts to print mode while saving the box, then places it inside a lateximage when used:

```
\renewcommand*\savestack[2]{%
  \xdef\sv@name{\stack@macro@name{#1}}%
  \@ifundefined{\sv@name content}{%
    \expandafter\newsavebox\expandafter{\csname\sv@name content\endcsname}%
   }{}%
    \begingroup%    lwarp
    \LWR@restoreorigformatting%     lwarp
    \RenewDocumentEnvironment{lateximage}{s o s o o}{}{}% lwarp: inside group
  \expandafter\LWR@gsavebox\csname\sv@name content\endcsname{#2}%
  \expandafter\gdef\expandafter#1\expandafter{%
        \expandafter\begin\expandafter{lateximage\expandafter}%     lwarp
        \expandafter\usebox\expandafter%
        {\csname\sv@name content\endcsname}%
        \expandafter\end\expandafter{lateximage\expandafter}%       lwarp
    }%
    \endgroup%      lwarp
}
```

File 362 **lwarp-steinmetz.sty**

§ 462 Package

steinmetz

(*Emulates or patches code by* Enrico Gregorio.)

Pkg steinmetz steinmetz is patched for use by lwarp. Emulation is provided for MathJax

for HTML output:

```
\LWR@ProvidesPackagePass{steinmetz}[2009/06/14]

\begin{warpHTML}
\renewcommand{\phase}[2][]{%
    \begin{lateximage}*[steinmetz\{\detokenize{#2}\}]
    \ensuremath{\underline{/#2}}
    \end{lateximage}
}
\end{warpHTML}

\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\phase}[2][]{\underline{/#2}}}
\end{warpMathJax}
```

File 363 **lwarp-stfloats.sty**

§ 463 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 364 **lwarp-struktex.sty**

§ 464 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}[-struktex-~\PackageDiagramAltText]%
}
\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}
```

```
16
17 \newcommand{\LWR@HTML@centernssfile}{\LWR@HTML@CenterNssFile}
18 \LWR@formatted{centernssfile}
```

File 365 **lwarp-subcaption.sty**

§ 465 Package

subcaption

(*Emulates or patches code by* Axel Sommerfeldt.)

Pkg subcaption subcaption is patched for use by lwarp.

for HTML output:

```
1 \LWR@ProvidesPackagePass{subcaption}[2018/05/01]
```

Tells lwarp to ignore minipage widths inside a subfigure or subtable. In print mode the minipages are used to place the items next to each other. In HTML they are placed side-by-side automatically.

```
2 \ifdef{\subcaption@minipage}{
3     \xpatchcmd{\subcaption@minipage}
4         {\minipage}
5         {\minipagefullwidth\minipage}
6         {}
7         {\LWR@patcherror{subcaption}{subcaption@minipage}}
8 }{}
```

Likewise for a \subcaptionbox:

```
9 \ifdef{\subcaptionbox}{
10     \xpretocmd{\subcaptionbox}
11         {\minipagefullwidth}
12         {}
13         {\LWR@patcherror{subcaption}{subcaptionbox}}
14 }{}
```

File 366 **lwarp-subfig.sty**

§ 466 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 367 **lwarp-subfigure.sty**

§ 467 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 368 **lwarp-supertabular.sty**

§ 468 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 8.10.1.

⚠ 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}%

\gdef\LWRST@caption{}%

\LWR@traceinfo{supertabular done}%
}

\NewDocumentEnvironment{mpsupertabular}{s o m}
{\minipage{\linewidth}\supertabular{#3}}
{\endsupertabular\endminipage}
```

File 369 **lwarp-svg.sty**

§ 469 Package svg

(*Emulates or patches code by* Philip Ilten, Falk Hanisch.)

Pkg svg svg is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{svg}[2019/10/22]

\xpretocmd{\includesvg}%
    {\begin{lateximage}}%
    {}%
    {\LWR@patcherror{svg}{includesvg}}

\xapptocmd{\includesvg}%
    {\end{lateximage}}%
    {}%
    {\LWR@patcherror{svg}{includesvg}}

\xpretocmd{\includeinkscape}%
    {\begin{lateximage}}%
    {}%
    {\LWR@patcherror{svg}{includeinkscape}}

\xapptocmd{\includeinkscape}%
    {\end{lateximage}}%
    {}%
    {\LWR@patcherror{svg}{includeinkscape}}
```

File 370 **lwarp-syntonly.sty**

§ 470 Package syntonly

(*Emulates or patches code by* Frank Mittelbach, Rainer Schöpf.)

Pkg syntonly syntonly is ignored.

for HTML output: Discard all options for lwarp-syntonly:

```
\LWR@ProvidesPackageDrop{syntonly}[2017/06/30]

\newif\ifsyntax@
\syntax@false

\newcommand*{\syntaxonly}{}

\@onlypreamble\syntaxonly
```

File 371 **lwarp-tabfigures.sty**

§ 471 Package tabfigures

Pkg tabfigures tabfigures is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{tabfigures}[2012/01/24]
```

File 372 **lwarp-tablefootnote.sty**

§ 472 Package tablefootnote

Pkg tablefootnote tablefootnote is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{tablefootnote}[2014/01/26]
```

This works because in HTML tables are no longer floats.

```
\LetLtxMacro\tablefootnote\footnote
```

File 373 **lwarp-tabls.sty**

§ 473 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 374 **lwarp-tabularx.sty**

§ 474 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]
\RequirePackage{array}

\DeclareDocumentEnvironment{tabularx}{m o m}
{\tabular{#3}}
{\endtabular}

\DeclareDocumentEnvironment{tabularx*}{m o m}
{\tabular{#3}}
{\endtabular}
```

File 375 **lwarp-tabulary.sty**

§ 475 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]
\RequirePackage{array}

\NewDocumentEnvironment{tabulary}{m o m}
{\tabular{#3}}
{\endtabular}

\NewDocumentEnvironment{tabulary*}{m o m}
```

```
{\tabular{#3}}
{\endtabular}

\newdimen\tymin
\newdimen\tymax
\def\tyformat{}
```

File 376 **lwarp-tascmac.sty**

§ 476 Package

tascmac

Pkg tascmac tascmac is emulated.

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}}

\def\ascii{ASCII Corporation}

\def\Ascii{ASCII Corporation}

\def\ASCII{ASCII Corporation}
```

File 377 **lwarp-textarea.sty**

§ 477 Package

textarea

(*Emulates or patches code by* Alexander I. Rozhenko.)

Pkg textarea textarea is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{textarea}[2005/12/26]

\newcommand\StartFromTextArea{}
\newcommand\StartFromHeaderArea{}
\newcommand*\RestoreTextArea{}
\newcommand*\ExpandTextArea[1][*]{}
\let\NCC@restoretextarea\@empty
```

File 378 **lwarp-textcomp.sty**

§ 478 Package textcomp

(*Emulates or patches code by* Frank Mittelbach, Robin Fairbairns, Werner Lemberg.)

Pkg textcomp textcomp is patched for use by lwarp.

§ 478.1 Limitations

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

⚠ missing symbols Many textcomp symbols are not supported by many system / browser fonts. In the css try referencing fonts which are more complete, but expect to see gaps in coverage.

§ 478.2 Package loading

for HTML output:

```
\LWR@ProvidesPackagePass{textcomp}[2017/04/05]
```

§ 478.3 HTML symbols

For HTML, use HTML entities or direct Unicode, depending on the engine.

`\AtBeginDocument` improves support for LuaLaTeX and XeLaTeX.

§ 478.3.1 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@textinterrobang}{\HTMLunicode{203D}}
\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
```

§ 478.3.2 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@textinterrobang}{}
\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{textinterrobang}
\LWR@formatted{textinterrobangdown}
\LWR@formatted{textperthousand}
\LWR@formatted{textpertenthousand}
\LWR@formatted{textbaht}
\LWR@formatted{textdiscount}
\LWR@formatted{textservicemark}
```

§ 478.4 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*{\textinterrobang}{}%
    \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 379 **lwarp-textfit.sty**

§ 479 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@templengthtwo}{\usebox{\LWR@textfitbox}}%
\LWR@textfitscale{#1}{#2}%
}
```

File 380 **lwarp-textpos.sty**

§ 480 Package textpos

(*Emulates or patches code by* Norman Gray.)

Pkg textpos textpos is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{textpos}[2019/04/15]
```

```
\NewDocumentEnvironment{textblock}{m o 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]{}
\newcommand*{\TPoptions}[1]{}
\newcommand*{\TPReferencePosition}[1]{}
```

File 381 **lwarp-theorem.sty**

§ 481 Package theorem

(*Emulates or patches code by* Frank Mittelbach.)

Pkg theorem theorem is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{theorem}[2014/10/28]
```

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.

§ 481.1 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}
```

§ 481.2 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 382 **lwarp-thinsp.sty**

§ 482 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 383 **lwarp-threadcol.sty**

§ 483 Package threadcol

Pkg threadcol threadcol is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{threadcol}[2013/01/06]
```

```
\newcommand{\setthreadname}[1]{}
```

File 384 **lwarp-threeparttable.sty**

§ 484 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*⟩]

```
\newenvironment*{threeparttable}[1][b]
    {\def\@captype{table}}
    {}
```

Env tablenotes [⟨*options*⟩]

```
\newenvironment*{tablenotes}[1][]
{%
\LWR@forcenewpage
\BlockClass{tnotes}%

\description%
}
{%
\enddescription%
\endBlockClass%
}
```

\tnote {⟨*text*⟩}

```
\newcommand{\tnote}[1]{\LWR@htmlspan{sup}{#1}}
```

Env measuredfigure [⟨*alignment*⟩]

```
\newenvironment*{measuredfigure}[1][t]
    {\def\@captype{figure}}
    {}
```

File 385 **lwarp-threeparttablex.sty**

§ 485 Package

threeparttablex

Pkg threeparttablex

threeparttablex is patched for use by lwarp.

threeparttablex is used with longtable and booktabs as follows:

```
\begin{longtable}{ [column specifiers] }
[ . . . ] \endfirsthead   % or \endhead, for print and HTML
\warpprintonly{          % not used in HTML
  [ . . . ] \endhead     % or \endfirsthead
  [ . . . ] \endfoot
  \bottomrule \insertTableNotes \endlastfoot
}
. . .  table contents . . .
\warpHTMLonly{   % HTML last footer
  \bottomrule
  \UseMinipageWidths     % optional
  \insertTableNotes
  \endlastfoot
}
\end{longtable}
```

table width

The table notes are created using a \multicolumn. By default the width is not specified to the browser, so long table notes can cause the table to be spread out horizontally. For HTML output, lwarp guesses the width of the table depending on the number of columns, then restricts its guess to a min/max range. To use this guess for the width of the table notes, use \UseMinipageWidths before \insertTableNotes. The width is then specified, and in many cases the result is an improvement in overall table layout.

for HTML output:

```
\LWR@ProvidesPackagePass{threeparttablex}[2013/07/23]
```

The width is guessed depending on the number of columns, then limited to a min/max.

```
\renewcommand\insertTableNotes{%
   \setlength{\LWR@templengthone}{.375in*\value{LWR@tabletotalLaTeXcols}}%
   \setlength{\LWR@templengthone}{\minof{\textwidth}{\LWR@templengthone}}%
   \setlength{\LWR@templengthone}{\maxof{2.5in}{\LWR@templengthone}}%
  \multicolumn{\value{LWR@tabletotalLaTeXcols}}{c}{%
     \parbox{\LWR@templengthone}{%
       \begin{tablenotes}[\TPTL@optarg]%
         \TPTL@font%
         \TPTL@body%
       \end{tablenotes}%
     }%
  }%
}

\renewcommand\TPTL@tnotex[2]{\tnote{\nameref{#2}}}
```

File 386 **lwarp-thumb.sty**

§ 486 Package thumb

Pkg thumb thumb is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{thumb}[1997/12/24]

\newcommand*{\Overviewpage}{}
\newlength{\thumbheight}
\newlength{\thumbwidth}
```

File 387 **lwarp-thumbs.sty**

§ 487 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 388 **lwarp-tikz.sty**

§ 488 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}[-tikz-~\PackageDiagramAltText]%
    \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}
```

```
33
34 \def\pgfutil@font@normalfont{\LWR@orignormalfont}
```

File 389 **lwarp-titleps.sty**

§ 489 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:

```
1 \LWR@ProvidesPackageDrop{titleps}[2016/03/15]
```

\pagestyle and \thispagestyle are already disabled in the lwarp code.

\newpagestyle {⟨*name*⟩} [⟨*style*⟩] {⟨*commands*⟩}

```
2 \NewDocumentCommand{\newpagestyle}{m o m}{}
```

\renewpagestyle {⟨*name*⟩} [⟨*style*⟩] {⟨*commands*⟩}

```
3 \NewDocumentCommand{\renewpagestyle}{m o m}{}
```

\sethead [⟨*el*⟩] [⟨*ec*⟩] [⟨*er*⟩] {⟨*ol*⟩} {⟨*oc*⟩} {⟨*or*⟩}

```
4 \NewDocumentCommand{\sethead}{o o o m m m}{}
```

\setfoot [⟨*el*⟩] [⟨*ec*⟩] [⟨*er*⟩] {⟨*ol*⟩} {⟨*oc*⟩} {⟨*or*⟩}

```
5 \NewDocumentCommand{\setfoot}{o o o m m m}{}
```

\settitlemarks * {⟨*names*⟩}

```
6 \NewDocumentCommand{\settitlemarks}{s m}{}
```

\headrule

```
7 \newcommand*{\headrule}{}
```

\footrule

```
8 \newcommand*{\footrule}{}
```

\setheadrule {⟨*length*⟩}

```
9 \newcommand*{\setheadrule}[1]{}
```

\setfootrule {⟨*length*⟩}

```
\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 390 **lwarp-titleref.sty**

§ 490 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 391 **lwarp-titlesec.sty**

§ 491 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 392 **lwarp-titletoc.sty**

§ 492 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

```
\newcommand*{\thecontentslabel}{thecontentslabel}
```

\thecontentspage

```
\newcommand*{\thecontentspage}{thecontentspage}
```

\contentslabel [⟨*format*⟩] {⟨*space*⟩}

```
\newcommand{\contentslabel}[2][]{\thecontentslabel}
```

\contentspage [⟨*format*⟩]

```
\newcommand{\contentspage}[1][]{\thecontentspage}
```

\contentspush {⟨*text*⟩}

```
\newcommand{\contentspush}[1]{}
```

\contentsuse {⟨*name*⟩} {⟨*text*⟩}

```
\newcommand{\contentsuse}[2]{}
```

\startcontents [⟨*name*⟩]

```
\newcommand*{\startcontents}[1][]{}
```

\stopcontents [⟨*name*⟩]

```
\newcommand*{\stopcontents}[1][]{}
```

\resumecontents [⟨*name*⟩]

```
\newcommand*{\resumecontents}[1][]{}
```

\printcontents [⟨*name*⟩] {⟨*prefix*⟩} {⟨*start*⟩} {⟨*code*⟩}

```
\newcommand{\printcontents}[4][]{}
```

\startlist [⟨*name*⟩] {⟨*list*⟩}

```
\newcommand{\startlist}[2][]{}
```

\stoplist [⟨*name*⟩] {⟨*list*⟩}

```
\newcommand{\stoplist}[2][]{}
```

\resumelist [⟨*name*⟩] {⟨*list*⟩}

```
\newcommand{\resumelist}[2][]{}
```

\printlist [⟨*name*⟩] {⟨*list*⟩} {⟨*prefix*⟩} {⟨*code*⟩}

```
\newcommand{\printlist}[4][]{}
```

File 393 **lwarp-titling.sty**

§ 493 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 66.8.

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{}
```

```
27
28 \preauthor{}
29 \postauthor{}
30
31 \predate{}
32 \postdate{}
```

`\LWR@maketitlesetup` Patches `\thanks` macros.

```
33 \renewcommand*{\LWR@maketitlesetup}{%
```

Redefine the footnote mark:

```
34 \def\@makefnmark{\textsuperscript{\@thefnmark}}
```

`\thefootnote` ⇒ `\nameuse`{arabic}{footnote}, or
`\thefootnote` ⇒ `\nameuse`{fnsymbol}{footnote}

Redefine the footnote text:

```
35 \long\def\@makefntext##1{%
```

Make the footnote mark and some extra horizontal space for the tags:

```
36 \makethanksmark~%
```

`\makethanksmark` ⇒ `\thanksfootmark` ⇒ `\tamark` ⇒
`\@thefnmark` ⇒ `\itshape` a (or similar)

Print the text:

```
37 ##1%
38 }% \@makefntext
39 }
```

`\thanksfootmark`

```
40 \renewcommand{\thanksfootmark}{%
41 %   \hb@xt@\thanksmarkwidth{\hfil\normalfont%
42         \thanksscript{%
43             \thanksfootpre \tamark \thanksfootpost%
44         }%
45 %   }%
46 }
```

`\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.

```
47 \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:

```
\LWR@stoppars\@thanks\LWR@startpars
```

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:

```
\LWR@stoppars\@thanks\LWR@startpars
```

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:

```
\LWR@stoppars\@thanks\LWR@startpars
```

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 66.7.

```
\@ifclassloaded{memoir}{
   \thanksmarkseries{arabic}
}{% not memoir
\if@titlepage
   \thanksmarkseries{arabic}
\else
```

```
    \thanksmarkseries{fnsymbol}
\fi
}% not memoir
```

File 394 **lwarp-tocbasic.sty**

§ 494 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 395 **lwarp-tocbibind.sty**

§ 495 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 396 **lwarp-tocdata.sty**

§ 496 Package

tocdata

(*Emulates or patches code by* Brian Dunn.)

Pkg tocdata tocdata is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{tocdata}[2019/07/06]


\renewcommand*{\LWR@maybetocdata}{%
    \ifdefempty{\TD@thistocdata}{}{%
        \qquad \InlineClass{authorartist}{\tocdataformat{\TD@thistocdata}}%
        \def\TD@thistocdata{}
    }
}


\renewrobustcmd{\tocdatapartprint}[4]
{%
    \InlineClass{authorartist}{%
        \qquad --- %
        \TDoptionalnameprint{#1}\TDoptionalnameprint{#2}#3#4%
    }%
}

\@ifundefined{chapter}{}{
```

```
    \let\tocdatachapterprint\tocdatapartprint
}
\let\tocdatasectionprint\tocdatapartprint
\let\tocdatasubsectionprint\tocdatapartprint

\newcommand*{\LWR@TD@settextalign}[1]{%
    \def\LWR@TD@textalign{justify}%
    \ifcsstring{TD@#1align}{\centering}%
        {\def\LWR@TD@textalign{center}}%
        {}%
    \ifcsstring{TD@#1align}{\raggedleft}%
        {\def\LWR@TD@textalign{right}}%
        {}%
    \ifcsstring{TD@#1align}{\raggedright}%
        {\def\LWR@TD@textalign{left}}%
        {}%
}

\renewcommand{\TDartistauthorprint}[5]{%
    \LWR@TD@settextalign{#1}%
    \begin{BlockClass}[text-align:\LWR@TD@textalign]{floatnotes}%
   \InlineClass{authorartist}{\TDoptionalnameprint{#2}\TDoptionalnameprint{#3}#4#5}%
    \end{BlockClass}%
}

\newcommand*{\LWR@TD@setnamealign}[1]{%
    \def\LWR@TD@textalign{justify}%
    \ifcsstring{TD@#1textalign}{\centering}%
        {\def\LWR@TD@textalign{center}}%
        {}%
    \ifcsstring{TD@#1textalign}{\raggedleft}%
        {\def\LWR@TD@textalign{right}}%
        {}%
    \ifcsstring{TD@#1textalign}{\raggedright}%
        {\def\LWR@TD@textalign{left}}%
        {}%
}

\renewcommand{\TDartistauthortextprint}[2]{%
    \LWR@TD@setnamealign{#1}%
    \begin{BlockClass}[text-align:\LWR@TD@textalign]{floatnotes}%
    #2%
    \end{BlockClass}%
}
```

File 397 **lwarp-tocenter.sty**

§ 497 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 398 **lwarp-tocloft.sty**

§ 498 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{tocloft}[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 399 **lwarp-tocstyle.sty**

§ 499 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 400 **lwarp-todo.sty**

§ 500 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 401 **lwarp-todonotes.sty**

§ 501 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 402 **lwarp-topcapt.sty**

§ 502 Package topcapt

Pkg topcapt topcapt is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{topcapt}[2004/12/11]

\LetLtxMacro\topcaption\caption
```

File 403 **lwarp-tram.sty**

§ 503 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 404 **lwarp-transparent.sty**

§ 504 Package transparent

(*Emulates or patches code by* Heiko Oberdiek.)

Pkg transparent transparent is 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@ProvidesPackagePass{transparent}[2019/11/29]

\newcommand*{\LWR@HTML@transparent}[1]{\edef\LWR@opacity{#1}}

\LWR@formatted{transparent}


\newcommand*{\LWR@HTML@texttransparent}[2]{%
\begingroup%
\transparent{#1}%
\InlineClass[opacity: #1]{transparent}{#2}%
\endgroup%
}

\LWR@formatted{texttransparent}
```

File 405 **lwarp-trimclip.sty**

§ 505 Package trimclip

Pkg trimclip trimclip is ignored.

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 406 **lwarp-trivfloat.sty**

§ 506 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}
```

§ 506.1 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 407 **lwarp-truncate.sty**

§ 507 Package truncate

Pkg truncate truncate is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{truncate}[2001/08/20]

\providecommand{\TruncateMarker}{}
\newcommand{\truncate}[3][\TruncateMarker]{#3}
```

File 408 **lwarp-turnthepage.sty**

§ 508 Package turnthepage

Pkg turnthepage turnthepage is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{turnthepage}[2011/03/24]

\newcommand{\turnthepage}{}
```

File 409 **lwarp-twoup.sty**

§ 509 Package twoup

Pkg twoup twoup is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{twoup}[2007/02/26]

\newcommand{\cleartolastpage}{}
```

File 410 **lwarp-typearea.sty**

§ 510 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 411 **lwarp-typicons.sty**

§ 511 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 412 **lwarp-ulem.sty**

§ 512 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 413 **lwarp-umoline.sty**

§ 513 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 414 **lwarp-underscore.sty**

§ 514 Package underscore

Pkg underscore underscore is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{underscore}[2006/09/13]
```

File 415 **lwarp-unicode-math.sty**

§ 515 Package unicode-math

(*Emulates or patches code by* Will Robertson.)

Pkg unicode-math unicode-math is supported as-is for HTML with SVGmath. For MathJax, many characters do not seem to be encoded correctly, and so are not read correctly by *pdftotext* for HTML. Symbol font commands are emulated, but not all combinations are supported by MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{unicode-math}[2019/09/26]

\begin{warpHTML}
% Not all are possible in MathJax.
\CustomizeMathJax{\let\symnormal\mathrm}
\CustomizeMathJax{\let\symliteral\mathrm}
\CustomizeMathJax{\let\symbb\mathbb}
\CustomizeMathJax{\let\symbbit\mathbb}% not italic
\CustomizeMathJax{\let\symcal\mathcal}
\CustomizeMathJax{\let\symscr\mathscr}
\CustomizeMathJax{\let\symfrak\mathfrak}
\CustomizeMathJax{\let\symsfup\mathsf}% not up
\CustomizeMathJax{\let\symsfit\mathit}% not sans
\CustomizeMathJax{\let\symbfsf\mathbf}% not sans
\CustomizeMathJax{\let\symbfup\mathbf}% not up
\CustomizeMathJax{\let\symbfit\mathit}% not bold
```

```
\CustomizeMathJax{\let\symbfcal\mathcal}% not bold
\CustomizeMathJax{\let\symbfscr\mathscr}% not bold
\CustomizeMathJax{\let\symbffrak\mathfrak}% not bold
\CustomizeMathJax{\let\symbfsfup\mathbf}% not sans
\CustomizeMathJax{\let\symbfsfit\mathit}% not bold nor sans
% Duplicates below are commented out.
\CustomizeMathJax{\let\symup\mathrm}
\CustomizeMathJax{\let\symbf\mathbf}% \symbfup defined above
\CustomizeMathJax{\let\symit\mathit}
% \CustomizeMathJax{\let\symbfit\mathit}% not bold
\CustomizeMathJax{\let\symsf\mathsf}
% \CustomizeMathJax{\let\symbfsfup\mathbf}% not sans
% \CustomizeMathJax{\let\symsfit\mathit}% not sans
% \CustomizeMathJax{\let\symbfsfit\mathit}% not bold nor sans
\CustomizeMathJax{\let\symtt\mathtt}
% \CustomizeMathJax{\let\symbb\mathbb}
% \CustomizeMathJax{\let\symbbit\mathbb}% not italic
% \CustomizeMathJax{\let\symscr\mathscr}
% \CustomizeMathJax{\let\symbfscr\mathscr}% not bold
% \CustomizeMathJax{\let\symfrak\mathfrak}
\CustomizeMathJax{\let\symbffrac\mathbffrac}
\end{warpHTML}
```

File 416 **lwarp-units.sty**

§ 516 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:

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\unit}[2][]{#1 #2}}
\CustomizeMathJax{\newcommand{\unitfrac}[3][]{#1 #2/#3}}
\end{warpMathJax}
```

File 417 **lwarp-unitsdef.sty**

§ 517 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 418 **lwarp-upref.sty**

§ 518 Package upref

Pkg upref upref is ignored.

for HTML output: Discard all options for lwarp-upref:

```
\LWR@ProvidesPackageDrop{upref}[2007/03/14]
```

File 419 **lwarp-url.sty**

§ 519 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 420 **lwarp-uspace.sty**

§ 520 Package uspace

Pkg uspace uspace is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{uspace}[2016/11/06]
```

File 421 **lwarp-verse.sty**

§ 521 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 422 **lwarp-versonotes.sty**

§ 522 Package

versonotes

(*Emulates or patches code by* Norman Gray.)

Pkg versonotes versonotes is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{versonotes}[2019/07/06]
```

```
\newcommand{\versonote}[1]{\marginpar{#1}}
\newdimen\versotextwidth
\newdimen\versoleftmargin
\newcommand*{\versolayout}{}
```

In case the user changed the page number before loading versonotes:

```
\setcounter{page}{1}
```

File 423 **lwarp-vertbars.sty**

§ 523 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 424 **lwarp-vmargin.sty**

§ 524 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 425 **lwarp-vowel.sty**

§ 525 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-~\PackageDiagramAltText]%
        \@vowel[#1]%
    }
    {%
        \@@vowel%
        \end{lateximage}%
    }
```

File 426 **lwarp-vpe.sty**

§ 526 Package vpe

Pkg vpe vpe is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{vpe}[2012/04/18]
```

File 427 **lwarp-vwcol.sty**

§ 527 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:

```
\LWR@stoppars%
\vwcolsetup{#1}%
```

Begin with no style:

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

`presep` and `postsep` are created with HTML `margin`s:

```
\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
\LWR@startpars
}
```

File 428 **lwarp-wallpaper.sty**

§ 528 Package

wallpaper

(*Emulates or patches code by* Michael H.F. Wilkinson.)

Pkg wallpaper wallpaper is ignored.

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 429 **lwarp-watermark.sty**

§ 529 Package watermark

(*Emulates or patches code by* Alexander I. Rozhenko.)

Pkg watermark watermark is ignored.

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 430 **lwarp-widetable.sty**

§ 530 Package widetable

(*Emulates or patches code by* Claudio Beccari.)

Pkg widetable widetable is emulated.

for HTML output:

```
\LWR@ProvidesPackageDrop{widetable}[2019-06-25]

\newenvironment{widetable}{\begin{tabular*}}{\end{tabular*}}
```

File 431 **lwarp-widows-and-orphans.sty**

§ 531 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 432 **lwarp-witharrows.sty**

§ 532 Package witharrows

(*Emulates or patches code by* F. Pantigny.)

Pkg witharrows witharrows is patched for use by lwarp. Emulation is provided for MathJax.

for HTML output:

```
\LWR@ProvidesPackagePass{witharrows}[2019/12/27]

\begin{warpHTML}
\makeatletter
\ifbool{mathjax}{
    % For the hidden print version in the HTML:
    \newcommand{\Arrow}[2][]{}
    \newcommand{\unicode}[1]{}
    \NewDocumentEnvironment { DispWithArrows } { ! d < > ! O { } +b}
        {
            \IfValueTF{#1}{
                \begin{displaymath}
                #1 \left\lbrace
                \begin{align}
                #3
                \end{align}
                \right .
                \end{displaymath}
            }{
                \begin{displaymath}
                \begin{align}
                #3
                \end{align}
                \end{displaymath}
            }
        }
        {}
    \NewDocumentEnvironment { DispWithArrows* } { ! d < > ! O { } +b}
```

```
        {
            \IfValueTF{#1}{
                \begin{displaymath}
                #1 \left\lbrace
                \begin{align*}
                #3
                \end{align*}
                \right .
                \end{displaymath}
            }{
                \begin{displaymath}
                \begin{align*}
                #3
                \end{align*}
                \end{displaymath}
            }
        }
        {}
}{
    % If not MathJax, use SVG images.
    \BeforeBeginEnvironment{WithArrows}{\global\booltrue{LWR@unknownmathsize}}
    \BeforeBeginEnvironment{DispWithArrows}{%
        \begin{BlockClass}{displaymathnumbered}%
        \begin{lateximage}%
    }
    \AfterEndEnvironment{DispWithArrows}{\end{lateximage}\end{BlockClass}}
    \BeforeBeginEnvironment{DispWithArrows*}{%
        \begin{BlockClass}{displaymath}%
        \begin{lateximage}%
    }
    \AfterEndEnvironment{DispWithArrows*}{\end{lateximage}\end{BlockClass}}
}
\makeatother
\end{warpHTML}

\begin{warpMathJax}
\CustomizeMathJax{\newenvironment{WithArrows}[1][]{\begin{aligned}}{\end{aligned}}}
% Unable to make a sized box.
\CustomizeMathJax{\newcommand{\Arrow}[2][]{&{\Large\unicode{x2938}}~\textit{#2}}}
\end{warpMathJax}
```

File 433 **lwarp-wrapfig.sty**

§ 533 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}%


\setlength{\linewidth}{\LWR@templengthone}%
}


\NewDocumentEnvironment{wrapfigure}{o m o m}
{%
\begin{LWR@setvirtualpage}*%
\LWR@subwrapfigure{#2}{#4}%
\captionsetup{type=figure}%
}
{%
\endLWR@BlockClassWP%
\end{LWR@setvirtualpage}%
}


\NewDocumentEnvironment{wraptable}{o m o m}
{%
\begin{LWR@setvirtualpage}*%
\LWR@subwrapfigure{#2}{#4}%
\captionsetup{type=table}%
}
{%
\endLWR@BlockClassWP%
\end{LWR@setvirtualpage}%
}


\NewDocumentEnvironment{wrapfloat}{m o m o m}
{%
\begin{LWR@setvirtualpage}*%
\LWR@subwrapfigure{#3}{#5}%
\captionsetup{type=#1}%
}
{%
\endLWR@BlockClassWP%
\end{LWR@setvirtualpage}%
```

```
}

\newlength{\wrapoverhang}
```

File 434 **lwarp-xbmks.sty**

§ 534 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 435 **lwarp-xcolor.sty**

§ 535 Package xcolor

(*Emulates or patches code by* Dr. Uwe Kern.)

Pkg xcolor xcolor is supported by lwarp.

§ 535.1 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.

§ 535.2 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 85. 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 503.

`\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 1036.

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 1036.

HTML lateximage: Remembers and reuses the print version.

\colorboxBlock:

Print: Becomes `\colorbox`.

HTML: Newly defined by lwarp-xcolor to use a `<div>`, page 1037.

HTML lateximage: Remembers and reuses the print version `\colorbox`.

\fcolorbox:

Print: Modified to allow a background of `none`.
`\LWR@print@fcolorbox` at section 85

HTML: Redefined by lwarp-xcolor, page 1037.

HTML lateximage: Remembers and reuses the print version.

\fcolorboxBlock:

Print: Becomes `\fcolorbox`. Section 85

HTML: Newly defined by lwarp-xcolor to use a `<div>`, page 1038.

HTML lateximage: Remembers and reuses the print version `\fcolorbox`.

fcolorminipage:

Print: Newly defined in the lwarp core.
`LWR@print@fcolorminipage` at section 85

HTML: Newly defined by lwarp-xcolor, page 1039.

HTML lateximage: Uses the print version.

\boxframe:

Print: Used as-is.

HTML: Redefined by lwarp-xcolor, page 1040.

HTML lateximage: Remembers and reuses the print version.

§ 535.3 Package loading

for HTML output:

```
\LWR@ProvidesPackagePass{xcolor}[2016/05/11]
\begin{warpHTML}
```

§ 535.4 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%
}
```

§ 535.5 `\normalcolor`

`\normalcolor`

```
\DeclareRobustCommand{\LWR@HTML@normalcolor}{\color{black}}%

\LWR@formatted{normalcolor}
```

§ 535.6 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%
}
```

§ 535.7 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}%
}
```

§ 535.8 High-level macros

\color [⟨*model*⟩] {⟨*color*⟩}

⚠ The current \color is used by HTML rules and frames, but does not affect the current HTML text output, due to the lack of HTML states and scoping limitations. Use \textcolor if possible.

```
\NewDocumentCommand{\LWR@HTML@color}{o m}{%
\IfValueTF{#1}{%
    \LWR@print@color[#1]{#2}%
    \convertcolorspec{#1}{#2}{HTML}\LWR@tempcolor%
}{%
    \LWR@print@color{#2}%
    \convertcolorspec{named}{#2}{HTML}\LWR@tempcolor%
}%
\edef\LWR@currenttextcolor{\LWR@origpound\LWR@tempcolor}%
}

\LWR@formatted{color}
```

\textcolor [⟨*model*⟩] {⟨*color*⟩} {⟨*text*⟩}

Converted into an HTML hex color span.

```
\NewDocumentCommand{\LWR@HTML@textcolor}{o m m}{%
\begingroup%
\IfValueTF{#1}{%
    \color[#1]{#2}%
}{%
    \color{#2}%
}
\InlineClass[color:\LWR@currenttextcolor]{textcolor}{#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 85

`\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}
```

§ 535.9 Row colors

\rowc@l@rs [⟨*cmds*⟩] {⟨*startrow*⟩} {⟨*odd color*⟩} {⟨*even color*⟩}

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

\def\rowc@l@rs[#1]#2#3#4%
{
\rownum=1%
   \@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{%
        \advance\rownum\@ne%
%   }%
    \@rowcolors%
}

\end{warpHTML}
```

File 436 **lwarp-xechangebar.sty**

§ 536 Package

xechangebar

Pkg xechangebar xechangebar is ignored

for HTML output:

```
\LWR@ProvidesPackageDrop{xechangebar}[2017/08/03]
\LWR@origRequirePackage{lwarp-changebar}
```

File 437 **lwarp-xellipsis.sty**

§ 537 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 438 **lwarp-xetexko-vertical.sty**

§ 538 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 439 **lwarp-xfakebold.sty**

§ 539 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}%
}
```

For MathJax, xfakebold is ignored.

```
\begin{warpMathJax}
\CustomizeMathJax{\newcommand{\setBold}[1][]{}}
\CustomizeMathJax{\newcommand{\unsetBold}{}}
\end{warpMathJax}
```

File 440 **lwarp-xfrac.sty**

§ 540 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 441 **lwarp-xltabular.sty**

§ 541 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}
\RequirePackage{ltablex}

\LWR@ProvidesPackageDrop{xltabular}[2018/05/23]

\DeclareDocumentEnvironment{xltabular}{o m m}
{\longtable{#3}}
{\endlongtable}
```

File 442 **lwarp-xltxtra.sty**

§ 542 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 443 **lwarp-xmpincl.sty**

§ 543 Package xmpincl

(*Emulates or patches code by* Maarten Sneep.)

Pkg xmpincl xmpincl is ignored.

for HTML output: Discard all options for lwarp-xmpincl:

```
\LWR@ProvidesPackageDrop{xmpincl}[2008/05/10]

\newcommand*{\includexmp}[1]{}
```

File 444 **lwarp-xpiano.sty**

§ 544 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-~\PackageDiagramAltText{}: \detokenize\expandafter{#2}%
    ]
    [\detokenize\expandafter{#1}]
\xpiano_keyboard:nn { #1 } { #2 }
\end{lateximage}
}
\ExplSyntaxOff

\LWR@formatted{keyboard}
```

File 445 **lwarp-xpinyin.sty**

§ 545 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 446 **lwarp-xr.sty**

§ 546 Package xr

(*Emulates or patches code by* Jean-Pierre Drucbert, David Carlisle.)

Pkg xr xr is patched for use by lwarp. The `*_html.aux` file is used. `\externaldocument` is modified to also accept the optional arguments for xr-hyper, which currently uses xr for HTML output.

See section 5.17.

for HTML output:

```
\LWR@ProvidesPackagePass{xr}[2019/07/22]%

\LetLtxMacro\LWR@orig@externaldocument\externaldocument

\RenewDocumentCommand{\externaldocument}{O{} O{} m O{}}{%
```

```
    \ifblank{#1}{%
        \LWR@orig@externaldocument{#3_html}%
    }{%
        \LWR@orig@externaldocument[#1]{#3_html}%
    }%
}
```

File 447 **lwarp-xr-hyper.sty**

§ 547 Package xr-hyper

(*Emulates or patches code by* David Carlisle.)

Pkg xr-hyper xr-hyper is replaced by xr, which is modified to accept the optional arguments for `\externaldocument`. So far, no hyperlinks are provided for citations.

See section 5.17.

for HTML output:

```
\LWR@ProvidesPackageDrop{xr-hyper}[2019/10/03]%

\LWR@origRequirePackage{lwarp-xr}
```

File 448 **lwarp-xtab.sty**

§ 548 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 8.10.1.

⚠ 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%


\gdef\LWRXT@caption{}%

\LWR@traceinfo{xtabular done}%
}
```

```
59
60 \NewDocumentEnvironment{mpxtabular}{s o m}
61 {\minipage{\linewidth}\xtabular{#3}}
62 {\endxtabular\endminipage}
```

File 449 **lwarp-xunicode.sty**

§ 549 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:

```
1 \LWR@loadbefore{xunicode}%
2
3 \LWR@ProvidesPackagePass{xunicode}[2011/09/09]
```

\textcircled becomes a span with a rounded border. \providecommand is used to avoid conflict with textcomp.

```
4 \providecommand*{\LWR@HTML@textcircled}[1]{%
5     \InlineClass[border: 1px solid \LWR@currenttextcolor]{textcircled}{#1}%
6 }
7
8 \LWR@formatted{textcircled}
```

Nullify xunicode macros when generating filenames:

```
9 \FilenameNullify{%
10     \renewcommand*{\textdegree}{}%
11     \renewcommand*{\textcelsius}{}%
12     \renewcommand*{\textohm}{}%
13     \renewcommand*{\textmu}{}%
14     \renewcommand*{\textlquill}{}%
15     \renewcommand*{\textrquill}{}%
16     \renewcommand*{\textcircledP}{}%
17     \renewcommand*{\texttwelveudash}{}%
18     \renewcommand*{\textthreequartersemdash}{}%
19     \renewcommand*{\textmho}{}%
20     \renewcommand*{\textnaira}{}%
21     \renewcommand*{\textpeso}{}%
22     \renewcommand*{\textrecipe}{}%
23     \renewcommand*{\textinterrobang}{}%
24     \renewcommand*{\textinterrobangdown}{}%
25     \renewcommand*{\textperthousand}{}%
26     \renewcommand*{\textpertenthousand}{}%
27     \renewcommand*{\textbaht}{}%
28     \renewcommand*{\textdiscount}{}%
29     \renewcommand*{\textservicemark}{}%
30     \renewcommand*{\textcircled}[1]{#1}%
31     \renewcommand*{\capitalcedilla}[1]{#1}%
32     \renewcommand*{\capitalogonek}[1]{#1}%
```

```
    \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 450 **lwarp-xurl.sty**

§ 550 Package xurl

Pkg xurl xurl is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{xurl}[2020/01/14]

\def\useOriginalUrlSetting{}
```

File 451 **lwarp-xy.sty**

§ 551 Package xy

(*Emulates or patches code by* Kristoffer H. Rose, Ross Moore.)

Pkg xy xy is patched for use by lwarp.

for HTML output:

```
\LWR@ProvidesPackagePass{xy}[2013/10/06]
```

After xy modules have been loaded:

```
\AtBeginDocument{
```

The original definitions without a `lateximage`:

```
\LetLtxMacro\LWR@orig@xy\xy
\LetLtxMacro\LWR@orig@endxy\endxy
```

The outer-most xy environment is placed in a `lateximage`, but not more than one level deep, which would conflict with xy:

```
\renewcommand*{\xy}{%
    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{0}%
        {\addtocounter{LWR@lateximagedepth}{1}}%
        {\begin{lateximage}[-xy-~\PackageDiagramAltText]}%
```

```
    \LWR@orig@xy%
}

\renewcommand*{\endxy}{%
    \LWR@orig@endxy%
    \ifnumcomp{\value{LWR@lateximagedepth}}{>}{1}%
        {\addtocounter{LWR@lateximagedepth}{-1}}%
        {\end{lateximage}}%
}
```

The \xybox must use the original definitions of \xy, \endxy:

```
\def\xybox#1{%
    \LWR@orig@xy#1\LWR@orig@endxy%
    \Edge@c={\rectangleEdge}\computeLeftUpness@%
}
```

If \xygraph is used, it is placed inside a lateximage:

```
\@ifundefined{xygraph}{}{

\LetLtxMacro\LWR@origxygraph\xygraph

\renewcommand{\xygraph}[1]{%
    \begin{lateximage}[-xy- xygraph \PackageDiagramAltText]
    \LWR@origxygraph{#1}
    \end{lateximage}
}

}% xygraph defined

}% AtBeginDocument
```

File 452 **lwarp-zhlineskip.sty**

§ 552 Package zhlineskip

Pkg zhlineskip zhlineskip is ignored.

for HTML output:

```
\LWR@ProvidesPackageDrop{zhlineskip}[2019/05/15]

\newcommand*\SetTextEnvironmentSinglespace[1]{}
\newcommand*\RestoreTextEnvironmentLeading[1]{}
\newcommand*\SetMathEnvironmentSinglespace[1]{}
\newcommand*\RestoreMathEnvironmentLeading[1]{}
```

File 453 **lwarp-zwpagelayout.sty**

§ 553 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 454 **lwarp-patch-komascript.sty**

§ 554 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.

```
\AtBeginDocument{
    \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}%
```

```
59          {\captionof{#1}[#2]{#3}}%
60          {\captionof{#1}{#3}}%
61 }
62
63 \RenewDocumentCommand{\setcapindent}{s m}{}
64 \renewcommand*{\setcaphanging}{}
65 \renewcommand*{\setcapwidth}[2][]{}
66 \renewcommand*{\setcapdynwidth}[2][]{}
67 \RenewDocumentCommand{\setcapmargin}{s o m}{}
```

File 455 **lwarp-patch-memoir.sty**

§ 555 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 555.1). 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}
```

§ 555.1 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}
```

Avoids a conflict with memoir:

```
\AtBeginDocument{%
    \@ifpackageloaded{subcaption}{}{
            \RequirePackage{lwarp-subfigure}
            \let\subfigure\subbottom
            \let\subtable\subtop
    }%
}%

\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
```

§ 555.2 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}{*}
```

§ 555.3 Page layout

```
\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]{}

\renewcommand*{\settypeblocksize}[3]{}
\renewcommand*{\setlrmargins}[3]{}
\renewcommand*{\setlrmarginsandblock}[3]{}
\renewcommand*{\setbinding}[1]{}
\renewcommand*{\setulmargins}[3]{}
\renewcommand*{\setulmarginsandblock}[3]{}
\renewcommand*{\setcolsepandrule}[2]{}

\renewcommand*{\setheadfoot}[2]{}
\renewcommand*{\setheaderspaces}[3]{}
\renewcommand*{\setmarginnotes}[3]{}
\renewcommand*{\setfootins}[2]{}
\renewcommand*{\checkandfixthelayout}[1][]{}
\renewcommand*{\checkthelayout}[1]{}
\renewcommand*{\fixthelayout}{}
%
% \newlength{\stockheight}
% \newlength{\trimtop}
% \newlength{\trimedge}
% \newlength{\stockwidth}
% \newlength{\spinemargin}
% \newlength{\foremargin}
% \newlength{\uppermargin}
% \newlength{\headmargin}
%
\renewcommand*{\typeoutlayout}{}
\renewcommand*{\typeoutstandardlayout}{}
\renewcommand*{\settypeoutlayoutunit}[1]{}
\renewcommand*{\fixpdflayout}{}
\renewcommand*{\fixdvipslayout}{}

\renewcommand*{\medievalpage}[1][]{}
\renewcommand*{\isopage}[1][]{}
\renewcommand*{\semiisopage}[1][]{}

\renewcommand{\setpagebl}[3]{}
\renewcommand{\setpageml}[3]{}
\renewcommand{\setpagetl}[3]{}
\renewcommand{\setpagetm}[3]{}
\renewcommand{\setpagetr}[3]{}
\renewcommand{\setpagemr}[3]{}
\renewcommand{\setpagebr}[3]{}
\renewcommand{\setpagebm}[3]{}
\renewcommand{\setpagecc}[3]{}
```

§ 555.4 Text and fonts

```
\let\miniscule\tiny
\let\HUGE\Huge

\renewcommand*{\abnormalparskip}[1]{}
\renewcommand*{\nonzeroparskip}{}
\renewcommand*{\traditionalparskip}{}
```

```
\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}{}
```

§ 555.5 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}% already provided by memoir
```

§ 555.6 Abstracts

```
\renewcommand*{\abstractcol}{}
\renewcommand*{\abstractintoc}{}
\renewcommand*{\abstractnum}{}
\renewcommand*{\abstractrunin}{}
```

§ 555.7 Docment divisions

\book * (⟨*2:PDF name*⟩) [⟨*3:TOC name*⟩] [⟨*4:PDF name*⟩] (⟨*5:PDF name*⟩) {⟨*6:name*⟩}

```
\DeclareDocumentCommand{\book}{s d() o o d() m}{%
    \LWR@section{#1}{#3}{#6}{book}%
}

\def\@apppage{%
    \part*{\appendixpagename}
}
\renewcommand\mempreaddapppagetotochook{}
\renewcommand\mempostaddapppagetotochook{}

\def\@sapppage{%
    \part*{\appendixpagename}
}


\DeclareDocumentCommand{\mainmatter}{s}{%
    \booltrue{LWR@mainmatter}%
}

\DeclareDocumentCommand{\frontmatter}{s}{%
    \boolfalse{LWR@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]{}

\renewcommand*{\hangsecnum}{}
\renewcommand*{\defaultsecnum}{}

\renewcommand*{\sechook}{}
\renewcommand{\setsechook}[1]{}
\renewcommand*{\subsechook}{}
\renewcommand{\setsubsechook}[1]{}
\renewcommand*{\subsubsechook}{}
\renewcommand{\setsubsubsechook}[1]{}
\renewcommand*{\parahook}{}
\renewcommand{\setparahook}[1]{}
\renewcommand*{\subparahook}{}
\renewcommand{\setsubparahook}[1]{}

\RenewDocumentCommand{\plainbreak}{s m}{\begin{center}~\end{center}}

\RenewDocumentCommand{\fancybreak}{s +m}{%
    \begin{center}#2\end{center}%
}

\RenewDocumentCommand{\plainfancybreak}{s m m +m}{%
    \begin{center}#4\end{center}%
}

\RenewDocumentCommand{\pfbreak}{s}{%
    \begin{center}
    \pfbreakdisplay
    \end{center}
}

% \newlength{\pfbreakskip}
\renewcommand{\pfbreakdisplay}{*\quad*\quad*}

\renewcommand{\makeheadstyles}[2]{}
\renewcommand*{\headstyles}[1]{}
```

§ 555.8 Pagination and headers

```
\renewcommand*{\savepagenumber}{}
\renewcommand*{\restorepagenumber}{}
\renewcommand*{\uppercaseheads}{}
\renewcommand*{\nouppercaseheads}{}

\renewcommand*{\bookpagemark}[1]{}
\renewcommand*{\partmark}[1]{}
\renewcommand*{\bibmark}{}
\renewcommand*{\indexmark}{}
\renewcommand*{\glossarymark}{}

\LWR@origpagestyle{empty}
\renewcommand*{\ps@empty}{}
\renewcommand*{\makepagestyle}[1]{}
\renewcommand*{\emptypshook}{}%
% \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}{}
```

§ 555.9 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}{}
```

§ 555.10 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]{}
```

§ 555.11 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

}
{\end{center}}

\renewcommand*{\sidecapstyle}{}
\renewcommand*{\overridescapmargin}[1]{}
% \newlength{\sidecapraise}
\renewcommand*{\sidecapfloatwidth}{\linewidth}

\LetLtxMacro\ctabular\tabular
\LetLtxMacro\endctabular\endtabular

\renewcommand{\autorows}[5][]{%
    #5%
}

\renewcommand{\autocols}[5][]{%
    #5%
}
```

§ 555.12 Page notes

```
\renewcommand*{\feetabovefloat}{}
\renewcommand*{\feetbelowfloat}{}
\renewcommand*{\feetatbottom}{}

\renewcommand*{\verbfootnote}[2][]{%
    \PackageError{lwarp,memoir}%
    {Verbatim footnotes are not yet supported by lwarp.}%
    {This may be improved some day.}%
}

\renewcommand*{\plainfootnotes}{}
\renewcommand*{\twocolumnfootnotes}{}
\renewcommand*{\threecolumnfootnotes}{}
```

```
\renewcommand*{\paragraphfootnotes}{}
\renewcommand*{\footfudgefiddle}{}

\renewcommand*{\newfootnoteseries}[1]{%
    \PackageError{lwarp,memoir}%
    {Memoir footnote series are not yet supported by lwarp.}%
    {This may be improved some day.}%
}

\renewcommand*{\plainfootstyle}[1]{}
\renewcommand*{\twocolumnfootstyle}[1]{}
\renewcommand*{\threecolumnfootstyle}[1]{}
\renewcommand*{\paragraphfootstyle}[1]{}

\renewcommand*{\footfootmark}{}
\renewcommand*{\footmarkstyle}[1]{}

% \newlength{\footmarkwidth}
% \newlength{\footmarksep}
% \newlength{\footparindent}

\renewcommand*{\foottextfont}{}

\renewcommand*{\marginparmargin}[1]{}
\renewcommand*{\sideparmargin}[1]{}

\LetLtxMacro\sidepar\marginpar
\renewcommand*{\sideparfont}{}
\renewcommand*{\sideparform}{}
\LWR@providelength{\sideparvshift}

\renewcommand*{\parnopar}{}

\renewcommand{\sidebar}[1]{\begin{quote}#1\end{quote}}
\renewcommand*{\sidebarmargin}[1]{}
\renewcommand*{\sidebarfont}{}
\renewcommand*{\sidebarform}{}
% \newlength{\sidebarhsep}
% \newlength{\sidebarvsep}
% \newlength{\sidebarwidth}
% \newlength{\sidebartopsep}
\renewcommand{\setsidebarheight}[1]{}
\renewcommand*{\setsidebars}[6]{}
\renewcommand*{\footnotesatfoot}{}
\renewcommand*{\footnotesinmargin}{}

\LetLtxMacro\sidefootnote\footnote
\LetLtxMacro\sidefootnotemark\footnotemark
\LetLtxMacro\sidefootnotetext\footnotetext

\renewcommand*{\sidefootmargin}[1]{}
% \newlength{\sidefoothsep}
% \newlength{\sidefootvsep}
% \newlength{\sidefootwidth}
% \newlength{\sidefootadjust}
```

```
% \newlength{\sidefootheight}
\renewcommand*{\setsidefootheight}[1]{}
% \renewcommand*{\sidefootfont}{}% in docs but not in the package
\renewcommand*{\setsidefeet}[6]{}
\renewcommand*{\sidefootmarkstyle}[1]{}
\renewcommand*{\sidefoottextfont}{}
\renewcommand*{\sidefootform}{}

\renewcommand*{\continuousnotenums}{\pncontopttrue}% from pagenote
\renewcommand*{\notepageref}{}
\renewcommand*{\prenotetext}{}
\renewcommand*{\postnotetext}{}
\renewcommand*{\idtextinnotes}[1]{}
\renewcommand*{\printpageinnotes}[1]{}
\renewcommand*{\printpageinnoteshyperref}[1]{}
\renewcommand*{\foottopagenote}{}
\renewcommand*{\pagetofootnote}{}
```

§ 555.13 Decorative text

```
\renewcommand*{\epigraphposition}[1]{}
\renewcommand*{\epigraphtextposition}[1]{}
\renewcommand*{\epigraphsourceposition}[1]{}
\renewcommand*{\epigraphfontsize}[1]{}
\renewcommand*{\epigraphforheader}[2][]{}
\renewcommand*{\epigraphpicture}{}
```

§ 555.14 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}{}
```

§ 555.15 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}{}
```

§ 555.16 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}{}
```

§ 555.17 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}{}
```

§ 555.18 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][]{}{}
```

§ 555.19 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} % space
    \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\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]{}
```

§ 555.20 Final patchwork

```
\newlistof{tableofcontents}{toc}{\contentsname}
\newlistof{listoffigures}{lof}{\listfigurename}
\newlistof{listoftables}{lot}{\listtablename}
```

File 456 **lwarp-common-multimedia.sty**

§ 556 Package common-multimedia

Pkg lwarp-common-multimedia

Common code for multimedia, movie15, and media9.

The packages multimedia, movie15, and media9 are supported.

HTML5 <audio> and <video> objects are created for .mp3 and .mp4 files.

HTML5 <embed> objects are created for http and ftp links.

\href links are created for other media types. (Unfortunately, there is not much overlap between the file types supported for print output and the file types supported by HTML5.)

For media9, a multimedia object is inserted for each addresource=, as well as each flashvars source= and src=. This may result in duplicate objects.

Undesired objects may be nullified by placing them inside \warpprintonly or the warpprint environment.

Each HTML multimedia object includes the poster text, except for <embed> objects. For movie15, the text option is supported to specify the poster text.

The width, height, and totalheight options are supported. The HTML object is scaled according to the display width, correctly compensating for either tall or wide viewports.

Other options are ignored.

media9 \addmediapath is supported. It is assumed that the same path structure will exist for the HTML document.

HTML5 media controls are always specified for each <audio> and <video> object.

media9 slideshows are not supported.

\hyperlinkmovie, \movieref, and \mediabutton are not supported.

3D objects are not supported.

If using a YouTube™ video, use an "embedded" URL with .../embed/... instead of .../v/...

for HTML output:

```
\ProvidesPackage{lwarp-common-multimedia}[2019/04/22]

\RequirePackage{xkeyval}

\define@key{LWR@multimedia}{width}{\setlength{\LWR@multimedia@width}{#1}}
\define@key{LWR@multimedia}{height}{\setlength{\LWR@multimedia@height}{#1}}
\define@key{LWR@multimedia}{totalheight}{\setlength{\LWR@multimedia@height}{#1}}

\newlength{\LWR@multimedia@width}
\newlength{\LWR@multimedia@height}
\newlength{\LWR@multimedia@maxdimension}
```

\LWR@multimedia@printsize

Proportional to \linewidth and the viewport's smaller dimension. This scales each object such that it will always fit on the screen, even if a tall or wide object inside a tall or wide viewport.

```
\newcommand*{\LWR@multimedia@printsize}{%
    \setlength{\LWR@multimedia@maxdimension}{%
        \maxof%
            {\linewidth}%
            {\maxof{\LWR@multimedia@width}{\LWR@multimedia@height}}%
    }%
    \setlength{\LWR@multimedia@maxdimension}{1.1\LWR@multimedia@maxdimension}%
    \ifdimgreater{\LWR@multimedia@width}{0pt}{%
        width:%
            \LWR@printpercentlength%
                {\LWR@multimedia@width}%
                {\LWR@multimedia@maxdimension}vmin ; % space
    }{}%
    \ifdimgreater{\LWR@multimedia@height}{0pt}{%
        height:%
            \LWR@printpercentlength%
                {\LWR@multimedia@height}%
                {\LWR@multimedia@maxdimension}vmin ; % space
    }{}%
}
```

\LWR@multimedia@fileAV {⟨*poster text*⟩} {⟨*filename*⟩} {⟨*audio/video*⟩} {⟨*mimetype*⟩}

Creates a video or audio from a file. The 2019/10 update of the LaTeX kernel may cause extra quotes to be added in the filenames. They are removed here.

```
\newcommand*{\LWR@multimedia@fileAV}[4]{%
\IfFileExists{#2}{% also sets \@filef@und
\StrSubstitute[100]{\@filef@und}{"}{}[\LWR@parsedfilename]%
```

The container is sized as desired.

```
    \ifstrequal{#3}{audio}{%
        \begin{BlockClass}{AVviewport}
    }{%
        \begin{BlockClass}[\LWR@multimedia@printsize\ margin:auto]{AVviewport}
    }
```

Paragraph tags are unnecessary for the A / V tags.

```
    \LWR@stoppars
```

The A / V element is 100% of the container.

```
    \LWR@htmltag{%
        #3\ % space
        \ifstrequal{#3}{audio}{}{%
            width=\textquotedbl{}100\%\textquotedbl\ % space
            height=\textquotedbl{}100\%\textquotedbl\ % space
        }%
        controls%
    }\LWR@orignewline
```

The file source and type:

```
    \LWR@htmltag{%
        source % space
        src=\textquotedbl%
        \LWR@parsedfilename\unskip\textquotedbl\ % space
        type=\textquotedbl{}#4\textquotedbl}
```

The poster text inside paragraph tags, along with a reference to the file.

```
    \LWR@startpars
    \href{\LWR@parsedfilename}{#1}
    \LWR@stoppars
```

Finish.

```
    \LWR@htmltag{/#3}\LWR@orignewline
    \end{BlockClass}
}{%
    \PackageError{lwarp-common-multimedia}
    {File '#2' not found}
    {Perhaps an incorrect path?}
}%
}
```

`\LWR@multimedia@httpAV` {⟨*poster text*⟩} {⟨*filename*⟩} {⟨*audio/video*⟩} {⟨*mimetype*⟩}

Creates a video or audio from a URL link.

```
\newcommand*{\LWR@multimedia@httpAV}[4]{%
```

The container is sized as desired.

```
    \ifstrequal{#3}{audio}{%
        \begin{BlockClass}{AVviewport}
    }{%
        \begin{BlockClass}[\LWR@multimedia@printsize\ margin:auto]{AVviewport}
    }
```

Paragraph tags are unnecessary for the A / V tags.

```
    \LWR@stoppars
```

The A / V element is 100% of the container.

```
    \LWR@htmltag{%
        #3\ % space
        \ifstrequal{#3}{audio}{}{%
            width=\textquotedbl{}100\%\textquotedbl\ % space
            height=\textquotedbl{}100\%\textquotedbl\ controls%
        }%
    }\LWR@orignewline
```

The file source and type:

```
    \LWR@htmltag{%
        source % space
        src=\textquotedbl#2\textquotedbl\ % space
        type=\textquotedbl#4\textquotedbl}
```

The poster text inside paragraph tags, along with a reference to the URL.

```
    \LWR@startpars
    \href{#2}{#1}
    \LWR@stoppars
```

Finish.

```
    \LWR@htmltag{/#3}\LWR@orignewline
    \end{BlockClass}
}
```

`\LWR@multimedia@AV` {⟨*poster text*⟩} {⟨*filename*⟩} {⟨*audio/video*⟩} {⟨*mimetype*⟩}

Creates an audio or video from a file or a URL.

```
\newcommand*{\LWR@multimedia@AV}[4]{%
    \IfBeginWith{#2}{http}%
        {\LWR@multimedia@httpAV{#1}{#2}{#3}{#4}}%
        {%
            \IfBeginWith{#2}{HTTP}%
                {\LWR@multimedia@httpAV{#1}{#2}{#3}{#4}}%
                {\LWR@multimedia@fileAV{#1}{#2}{#3}{#4}}%
        }%
}
```

`\LWR@multimedia@embed` {⟨*poster text*⟩} {⟨*URL or filename*⟩} {⟨*mime type*⟩}

Embeds multimedia of an arbitrary type. The poster text is not used, as it would appear along with the video if the <embed> element is supported.

```
\newcommand*{\LWR@multimedia@embed}[3]{%
    \begin{BlockClass}[width:100\%]{AVviewport}%
    \LWR@stoppars
```

```
        \LWR@htmltag{%
            embed % space
            \ifblank{#3}{}{type=\textquotedbl#3\textquotedbl\ }%
        style=\textquotedbl\LWR@multimedia@printsize\ margin:auto\textquotedbl\ % space
            src=\textquotedbl#2\textquotedbl\ % space
        }%
    \LWR@startpars
    \end{BlockClass}
}
```

Error message if the comment character is used among the arguments of `\LWR@multimediab`.

`\LWR@multimedia@percenterror`

```
\newcommand*{\LWR@multimedia@percenterror}{%
    \PackageError{lwarp-media9}
    {%
        Do not use a percent comment
        between\MessageBreak
        \protect\includemedia\space arguments%
    }
    {Percent is changed to a regular character to allow its use inside a URL.}
}
```

`\LWR@multimediab` `[`⟨*options*⟩`]` `{`⟨*poster text*⟩`}` `{`⟨*filename*⟩`}`

Creates multimedia. Examines the file extension to determine the type. If not a supported type, creates an embedded object if it has a URL. If neither, create a link to the unsupported object.

```
\newcommand*{\LWR@multimediab}[3][]{%
```

Error if the percent character appears among the arguments. This could happen since the comment character has been temporarily disabled, for use in a URL.

```
    \if#1\@percentchar\LWR@multimedia@percenterror\fi%
    \if#2\@percentchar\LWR@multimedia@percenterror\fi%
    \if#3\@percentchar\LWR@multimedia@percenterror\fi%
```

Paragraph handling:

```
    \LWR@stoppars%
```

Record the desired size.

```
    \setlength{\LWR@multimedia@width}{0pt}%
    \setlength{\LWR@multimedia@height}{0pt}%
    \setkeys*{LWR@multimedia}{#1}%
```

If a known A / V type, create an HTML5 <video> or <audio>.

```
    \IfEndWith{#3}{.mp4}{\LWR@multimedia@AV{#2}{#3}{video}{video/mp4}}{%
    \IfEndWith{#3}{.MP4}{\LWR@multimedia@AV{#2}{#3}{video}{video/mp4}}{%
    \IfEndWith{#3}{.mp3}{\LWR@multimedia@AV{#2}{#3}{audio}{audio/mpeg}}{%
    \IfEndWith{#3}{.MP3}{\LWR@multimedia@AV{#2}{#3}{audio}{audio/mpeg}}{%
```

If an arbitrary URL, embed it.

```
    \IfBeginWith{#3}{http}{\LWR@multimedia@embed{#2}{#3}{}}{%
    \IfBeginWith{#3}{HTTP}{\LWR@multimedia@embed{#2}{#3}{}}{%
    \IfBeginWith{#3}{ftp}{\LWR@multimedia@embed{#2}{#3}{}}{%
    \IfBeginWith{#3}{FTP}{\LWR@multimedia@embed{#2}{#3}{}}{%
```

If unknown, create a link to it.

```
        \href{#3}{#2}% unknown format
    }}}}}}}%
```

Paragraph handling:

```
    \LWR@startpars%
    \endgroup%
}
```

Catcodes which may apper in a URL.

```
\newrobustcmd*{\LWR@multimedia}{%
    \begingroup%
    \LWR@linkmediacatcodes%
    \LWR@multimediab%
}
```

Change History

§ 557 ## Chg Hist

For the most recent changes, see page 1118.

v0.10
General: 2016/03/08 Initial version . . 1
v0.11
General: 2016/03/11 1
Added section: Operating-System portability. 213
Added section: Selecting the operating system. 114
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. 364
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. 487
\LWR@LwarpStart: \up and \fup . . . 383
General: 2016/03/24 1
Fix dollar-redefined bug for newer package. 985
Removed package: subfig 1
Test Suite: Ordinals, Subcaption . . . 1
v0.14
\LWR@htmlsectionfilename: Fix: Links to home page. 324
General: 2016/03/31 1
floatrow: Added. 732
Docs: Commands for a successful HTML conversion. 118
Docs: Commands into a warpprint environment. 115
Docs: Newclude limitations. . . . 165
Docs: Table: Cross-referencing data structures. 471
Docs: Table: Float data structures. 483
Docs: Trademarks section. 191
Docs: Troubleshooting cross-references. 186
Test Suite: Assigned cleveref name for Test Float. 1
Test Suite: Floatrow 1
v0.15
General: 2016/04/06 1
Added. 737
Ampersand (&): Fixed handling when passed as an argument. . . 417
Docs: Added warning icons for items needing special attention. 194
Docs: Clarify print/HTML output. 114
Docs: Moved the supported features table to the introduction. 65
Files: lwarp_formal.css added. 1
Fix: steps counter 736
Fixed & handling. 735
Test Suite: test_suite_formal.css file added. 1
v0.16
General: 2016/04/11 1
\titlingpage: Improved print-output spacing. 390
xfrac: Adjusted for the use of any font: 1044
Added XeLaTeX, LuaLaTeX support. 195
Docs: Font and UTF-8 support. . . 98
Docs: Moved location of \usepackage{lwarp}. 101
Docs: Text not converting. 186
Lwarp no longer selects fonts. 98, 223
Removed package: suffix 1
Test Suite: Improved titlingpage. 390
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. 324
General: 2016/04/14 1
mdframed: Added. 826

Test Suite: Fix: Print-version front-matter page numbers. 1
Test Suite: Mdframed 1
v0.18
\LWR@myshorttoc: Reorganize \HomeHTMLFilename logic. 491
\LWR@newhtmlfile: sideTOC after title, improving responsive design. . . 363
\LWR@requesttoc: Reorganize \HomeHTMLFilename logic. 384
\LWR@subhyperref: Improved HTML output linebreaks. 480
\LWR@subhyperrefclass: Improved HTML output linebreaks. 480
\LWR@subinlineimage: Suppress extra space. 481
\hspace: \hspace supported. 579
General: 2016/05/19 1
graphics: Add: svgz file extension. 762
graphics: Fix: \linewidth, \textwidth, \textheight inside a minipage. 762
graphics: Improved HTML output linebreaks. 762
graphics: em, ex, %, px dimensions preserved. 762
File: lwarp.css: Improved TOC outline display. 1
Files: lwarp.css and lwarp_formal.css: Improved responsive design. 1
Microtype disabled during HTML generation 223
PDF Unicode input characters. . 209
Test Suite: Verse package 1
lateximage: pdfcrop: --hires added. 534
Reorganize \HomeHTMLFilename logic. 534
Suppress extra space. 534
verse: Supports verse, memoir packages. 1021
minipage: Fix: \linewidth, \textwidth, \textheight inside a minipage. 555
v0.19
\HTMLFilename: Docs: Escape filename underscores. 323
\HomeHTMLFilename: Docs: Escape filename underscores. 323
\LWR@LwarpStart: Enabled \\ equal to \newline. 381
\LWR@doequation: MathJax support. 521
\LWR@doubledollar: MathJax support. 515
\LWR@filestart: lwarp_mathjax.txt loaded. 379
\LWR@minipagestartpars: Suppresses paragraph tags between minipages. 578
\LWR@subsingledollar: MathJax support. 513
\LateximageFontSizeName: Add: User-adjustable math/lateximage font size. 530
\hspace: Fix: \hspace length computations. 579
\minipagefullwidth: Added: No width tag for the next minipage in HTML. 554
\warpHTMLonly: Added. 221
\warpprintonly: Replaces \rowprintedonly. 221
\xfracHTMLfontsize: Added. . . . 1044
General: 2016/06/08 1
CSS for table note item. 982
MathJax support added. 519, 525, 526
multirow: Added optional args. . . 851
Adapts to tikz version. 985
Avoids MathJax. 507
cleveref: Loaded \AtEndPreamble. 549
Docs: Math options. 101
Docs: Table: Cross-referencing data structures, updated. 471
File: lwarp.css: tnoteitemheader added. 1
File: lwarp_mathjax.txt added. . . 1
Introduction: MathJax support mentioned. 62
Options: mathsvg and mathjax . 216
Supports colored \rule. 1035
titleps: null \pagestyle and \thispagestyle for HTML. 986
v0.20
\BlockClassSingle: Renamed from "LWR@htmldivclassline". 337
\HTMLDescription: Added \NewHTMLdescription. (Renamed in v0.30.) 347
\HTMLFilename: No longer escape underscores. 323
\HomeHTMLFilename: No longer escape underscores. 323
\InlineClass: Renamed from "inlineclass". 337
\LWR@LwarpStart: Fix: math cross references. 383
\LWR@closeparagraph: \unskip extra spaces. 341

No break tags in the start/end of a tabular. 341
\LWR@endofline: Fix: \\ 577
\LWR@filestart: Adds meta description. 379
\LWR@htmldivclass: Added optional style. 335
\LWR@htmlelementclass: Added optional style. 335
\LWR@htmlsectionfilename: HTMLFilename: removed additional trailing '-', and may be empty. 324
Sections called "Index" or "index" have an underscore prepended to their filenames if no prefix. 324
\LWR@longtabledatacaptiontag: Fix: Pars in captions. 454
\LWR@section: Combined higher-level sections together into files. 370
\LWR@setOSWindows: Auto-detects operating system. 215
\LWR@subhtmlelementclass: Factored code. 334
\SetHTMLFileNumber: Add: Control file numbers. 323
\cpagerefFor: User-redefinable word for page references. 550
\dotfill: Inserts an ellipsis. 577
\hfill: Inserts a \qquad. 577
\hrulefill: Inserts a short rule. . . 577
\hspace: Add: Supports HTML thin breakable space. 579
\hyperindexref: Print mode provided in case hyperref not used. 501
\pageref: Added. 478
\tracinglwarp: Added. 237
General: 2017/02/09 1
afterpage: Added. 598, 603
alltt: Added. 608
bookmark: Added. 641
caption and subcaption supported. . 1
cleveref and referencing patches: Applied \AfterEndPreamble. . . . 549
draftwatermark: Added. 697
eso-pic: Added. 710
everypage: Added. 711
extramarks: Added. 711
fancyhdr: Added. 717
float: Improved float caption type handling. 729
graphics: Fix: Expands filename. . 762
graphics: Fix: \linewidth in a floatrow. 762
hyperref: Additional user macros. 772
keyfloat: Added. 788
letterspace: User-interface emulated. 797
listings: Added. 801
ltcaption: Added. 810
lwarp-newproject: Added. 249
microtype: User-interface emulated. 841
needspace: Added. 860
nowidow: Added. 872
placeins: Added. 909
ragged2e: Added. 917
setspace: Improved support. . . . 935
textpos: Added. 977
titleps: Added. 986
titlesec: Added. 989
titletoc: Added. 991
titling: Improved compatibility. . 992
tocloft: Added. 1001
wallpaper: Added. 1026
wrapfig: Added. 1029
xetexko-vertical: Added. 1043
Added @, <, > columns. 410
Added single-expansion data arrays. 320
Code factored into independent lwarp_html files. 593
Docs: Examples for generating HTML file names. 112
Docs: Improved index. 1
Enhanced titling support. 389
File: lwarp.css: Minor fixes for validation. 1
File: lwarpmk used to compile print, HTML, indexes, and latexmages. 1
Fix: \linewidth in a floatrow. . . 735
Moved sidebar and example code to test suite. 1
Page geometry set to 6in wide with large margins. 224
Parallel versions of aux files for print/HTML. 1
Removed reliance on make, grep, gawk. 1
Tabular: \unskip extra spaces. . . 410
Test Suite: HTML meta descriptions. 1
verbatim: Added. 400
BlockClass: Added optional style. . 336
Renamed from "blockclass". . . . 336
LWR@nestspan: Fix: Minipages inside a span. 332
v0.21
\LWR@LwarpStart: Changed latexmages to a .txt file. 381

\LWR@filestart: Skip title if not given. 379
\LWR@newhtmlfile: Skip title if not given. 363
\marginpar: Fixed source listing. . . 354
\marginparBlock: Fixed source listing. 354
General: 2017/02/23 1
fontenc: Added. 742
lwarpmk: Fix: **lwarpmk again** for WINDOWS. 299
lwarpmk: Fix: **lwarpmk limages** for WINDOWS. 299
lwarpmk: Fix: *lwarpmk* uses lateximages text file instead of shell script. 299
Add: Errors for misplaced packages. 195
Docs: Added internet class. 71
Docs: Added TeX2page, GladTeX. . 71
Docs: Installing on WINDOWS. . . . 77
File: lwarp_tutorial.txt added. . 81
v0.22
\LWR@parseDcolumn: Added tabular D column. 426
\LWR@parsebangcolumn: Added tabular ! column. 422
\LWR@parsetablecols: Unknown table column types become l. Added tabular D, !, X columns. . 430
\LWR@printmccoldata: Added tabular D, !, and X columns. 449
General: 2017/03/02 1
abstract: Added. 595
changepage: Added. 654
dcolumn: Added. 693
ftnright: Added. 749
geometry: Nullified commands. . 751
layout: Added. 794
lscape: Added. 809
mcaption: Added. 825
nameref: Added. 858
nextpage: Added. 860
parskip: Added. 892
showkeys: Added. 937
sidecap: Added. 938
tabularx: Added. 970
varioref: Supported. 126
verse: Added. 1021
v0.23
\LWR@parsetablecols: Fix for vert bar column type. 430
\LWR@printmccoldata: Fix for vert bar column type. 449
General: 2017/03/02 1
v0.24
\LWR@htmlfileref: Fix: Index links while \tracinglwarp. 473
\hspace: Add: \hspace \fill converts to 2em 579
\hypertocfloat: List of floats responds to lofdepth, lotdepth. 497
General: 2017/03/15 1
floatrow: Support for subfig. . . . 732
subfig: Added. 961
tikz: For tikz v3.0.0 or later, auto-loads tikz babel library if necessary. 985
Docs: Filename underscore. 106, 120
Fix for inline images. 985
No longer preloads subcaption; conflicted with subfig. 228
picture: Fix for inline images. . . . 552
v0.25
\LWR@loadnever: Added the ability to prevent conflicting packages. . . 197
\addcontentsline: Handles theorems. 489
General: 2016/03/22 1
amsthm: Added. 612
ellipsis: Added. 699
emptypage: Added. 700
framed: Added. 746
lips: Added. 801
mdframed: Help avoid hyphenation. 827
ntheorem: Added. 873
showidx: Added. 937
theorem: Added. 977
Basic LaTeX theorems: improved css. 401
Docs: Adds credits for patched code. 1
Docs: Testing lwarp. 182
Fix: Allows XeLaTeX and LuaLaTeX to preload graphics and graphicx. 199
v0.26
General: 2017/03/31 1
lwarp.css: Improved responsive marginpar and marginblock. . . 256
cutwin: Added. 692
endnotes: Added. 702
floatflt: Added. 731
footmisc: Added. 742
footnotehyper: Added. 745
footnote: Added. 744
marginfix: Added. 819
marginnote: Added. 820
mparhack: Added. 846
pagenote: Supported as-is. 887

sidenotes: Added. 939
Docs: Improved MiKTEX install instructions. 76, 78
Dollar span avoided in a lateximage. 507
Footnotes now are LATEX boxes instead of pagenotes. 348
lateximage: Labels track page numbers of lateximages. 534
Print mode now uses a minipage of \linewidth. 534
picture: Fix for \makebox in picture. 552
v0.27
\LWR@@footnotetext: Fix for table footnote par tags. 350
General: 2017/04/04 1
lettrine: Added. 798
microtype: Fix with XeLATEX, LuaLATEX. 841
soul: Added. 952
ulem: Added. 1015
Docs: Installing utilities for MACOS. 79
Docs: Limitations of saveboxes. . 121
Page geometry modified to reduce line overflow. 224
v0.28
\@wrindex: Improved indexing. . . . 500
\HTMLAuthor: Added \HTMLauthor. (Renamed in v0.30.) 347
\LWR@LwarpEnd: If FormatEPUB or FormatWP, no bottom nav. 384
\LWR@LwarpStart: FormatWordProcessor forces single-file output. 381
\LWR@filestart: Adds HTML meta author. 379
\LWR@forcenewpage: Forces new PDF page before major environments. 328
\LWR@htmlcomment: Breaks ligatures in HTML comments. 334
\LWR@newhtmlfile: If FormatEPUB or FormatWP: skips headers, footers, nav. 363
\LWR@parsetablecols: Added L, C, R, J column types. 430
\LWR@startref: Removed space. . . 476
\chapter: If EPUB, prints footnotes before each section. 377
\hyperindexref: Improved indexing. 501
\textup: Fixed span class. 566
General: 2017/04/14 1
glossaries: Added. 753
graphics: Adapts to graphics syntax. 762
graphics: Added. 754
tabularx: Fix for optional pos. . . . 970
tabulary: Added. 970
lwarpmk: Add: **printglossary** and **htmlglossary** commands. 299
Added boolean FormatEPUB. . . . 242
Added boolean FormatWP. 242
Added boolean HTMLDebugComments. 237
Added boolean HTMLMarkFloats, changed to WPMarkFloats as of v0.42. 243
Docs: Modfying lwarpmk and index processing. 182
File: lwarp_mathjax.txt: Updated CDN repository. 297
Forced oneside to maintain large right margin. 224
v0.29
General: 2017/04/15 1
*.lwarpmkconf: Add: language option for config files. 256
lwarpmk.conf: Add: language option for config files. 256
graphics: Fix: Error when no optional arguments. 762
lwarpmk: Add: language option for config files. 299
Add: lwarpmklang option for lwarp. 217
Docs: Using a glossary 92
v0.30
\CSSFilename: Renamed from \NewCSS. 346
\HTMLAuthor: Renamed from \HTMLauthor. 347
\HTMLDescription: Renamed from \NewHTMLdescription. 347
\HTMLFirstPageTop: Renamed from \SetFirstPageTop. 345
\HTMLLanguage: Renamed from \MetaLanguage. 378
\HTMLPageBottom: Renamed from \SetPageBottom. 345
\HTMLPageTop: Renamed from \SetPageTop. 345
General: 2017/04/29 1
lwarp-newproject removed, and combined with lwarp. 249
lwarpmk: Add: xdyfile configuration option. 299
lwarpmk: Fix: *xindy* and *texindy* adjusted for *pdflatex*, *xelatex* and *lualatex*. 299

lwarpmk: Fix: *xindy* now used for print index generation with *latexmk*. 299
lwarpmk: `language` now used for both index and glossary generation. 299
File: `lwarp_html.xdy` renamed to `lwarp.xdy`. 295
Fix: `*.css` files only written in print mode. 256
Fix: `lwarp.xdy` only written in print mode. 295
Fix: `lwarp_mathjax.txt`: Only written in print mode. 297
Option `lwarpmklang` changed to `IndexLanguage`. 217
Option `OSWindows` replaces macro `\warpOSwindows`. 218
Option `xdyFilename` added. 217
Option *latexmk* replaces macro `\UseLatexmk`. 219
Options `HomeHTMLFilename` and `HTMLFilename` replace macros `\HomeHTMLFilename` and `\HTMLFilename`. 218
v0.31
General: 2017/05/15 1
keyfloat: Improved compatibility. 788
v0.32
`\RequirePackage`: Fix: Ignores blanks in package list. 232
General: 2016/06/09 1
glossaries: Prevent error with `\glo@name` not defined. 502
lwarpmk: Fix: **`io.lines()`** changed to **`file:lines()`** due to *luatex* changes. 299
v0.33
`\HTMLAuthor`: Fix: Provides empty default author if none given. . . . 347
`\LWR@loadbefore`: Fix: No `\PackageError` if already loaded. 197
`\LWR@parseatcolumn`: Fix: Column alignment with leftmost `@`. . . . 421
`\LWR@tabledatasinglecolumntag`: Fix: Macros in tabular could cause extra data cell. 437
`\LWR@vspace`: Add: `\vspace` nullified. 581
`\StartDefiningTabulars`: Add: Avoids error: `Misplaced alignment tab character &`. . . 320
General: 2017/07/10 1
amsmath: Removed `fleqn` option. 609
fancyhdr: Fix: Optional args for `\lhead`, etc. 717
Add: Tabular at and bang columns now have their own HTML columns. 410
cleveref: Fix: Loaded `\AtEndPreamble`. 549
Fix: Incorrectly-inline math environments. 526
New handling of `&` to localize catcode changes. 410
v0.34
`\@fnsymbol`: Text symbols instead of math. 392
`\InlineClass`: Moved optional argument in front of mandatory. 337
`\LWR@htmldivclass`: Moved optional argument in front of mandatory. 335
`\LWR@htmlelementclass`: Moved optional argument in front of mandatory. 335
`\LWR@htmlelementclassline`: Moved optional argument in front of mandatory. 335
`\LWR@htmlspanclass`: Moved optional argument in front of mandatory. 333
`\LWR@nullfonts`: Improved font control. 568
`\LWR@restoreorigformatting`: booktabs: Works inside `lateximage`. 503
Improved font control. 503
`\LWR@subhtmlelementclass`: Moved optional argument in front of mandatory. 334
`\LWR@tabledatacolumntag`: booktabs: Works inside `lateximage`. 461
`\fboxBlock`: Added. 561
`\makebox`: Fix: Handles paren arg. . 559
General: 2017/08/08 1
babel-french: Adds fixed-width HTML spaces to punctuation. . . 330
balance: Added. 631
booktabs: Works inside `lateximage`. 464, 641
boxedminipage2e: Added. 644
crop: Added. 689
enumerate: Added. 703
enumitem: Added, no longer required. 703
everyshi: Added. 711
fancybox: Added. 713
fancyvrb: Added, no longer required. 719
figcaps: Added. 725
filecontents: Required. Patched for morewrites. 226

floatpag: Added. 732
flushend: Added. 737
fullpage: Added. 749
hyperxmp: Added. 777
idxlayout: Added. 778
marginfit: Added. 819
mdframed: Improved mdtheorem patch. 833
moreverb: Added. 844
paralist: Added. 887
pdflscape: Added. 896
pdfsync: Added. 899
prelim2e: Added. 911
rotfloat: Added. 923
savetrees: Added. 924
shadow: Added. 936
syntonly: Added. 969
titleps: No longer required. 986
titleref: Prevented. 989
xmpincl: Added. 1047
Added. 1039
Docs: Horizontal space limitations. 1
Docs: Misplaced alignment character. 186
File: lwarp_mathjax.txt: Version change. 297
File: README.txt: updated. 1
Fix: Added the eqnarray environments. 526
Improved font control. 564
Lists refactored to remove enumitem requirement. 402
Verbatim refactored to remove fancyvrb requirement. 398
tabular: booktabs: Works inside lateximage. 466
lateximage: Fix: lateximage with minipage, \parbox, \makebox, \fbox, \framebox, \raisebox, \scalebox, \reflectbox. 534
BlockClass: Moved optional argument in front of mandatory. 336
fminipage: Added. 561
LWR@nestspan: Fix: Minipages, BlocksClass, and lists inside a span. 332
v0.35
General: 2017/08/08 1
Fix: \textbf and related. 564
v0.36
\LWR@@footnotetext: Extra HTML source space after paragraphs. . 350
Force HTML superscripts. 350
\LWR@HTMLsanitizeexpand: Fix for babel-french. 361
\LWR@closeparagraph: Extra HTML source space after paragraphs. . 341
\LWR@currenttextcolor: Fix for \rule when xcolor not loaded. . 575
\LWR@nullfonts: Fix: Filenames while using MathJax. 568
\LWR@restoreorigformatting: siunitx: Improved super/subscripts in a lateximage. 503
\LWR@section: Improved spacing. . 370
\LWR@stoppars: Extra HTML source space after paragraphs. 343
\LWR@subHTMLsanitize: Fix for babel-french. 360
\fbox: Fix: Uses \fboxrule and \fboxsep. 560
\framebox: Fix: Handles width and horiz position. 560
\makebox: Fix: Handles width and horiz position. 559
General: 2017/08/17 1
babel-french: Adjustements for French variants, load order, footnotes, ellipses. 330
footnote: Extra HTML source space after paragraphs. 744
siunitx: Fix for babel-french. . . . 542
siunitx: Improved symbol support. 941
transparent: Added. 1011
upref: Added. 1020
xcolor: Added \fcolorboxBlock, \colorboxBlock. 1031
xcolor: Fix: Background none in print mode. 1031
xcolor: Refactored \LWR@colorstyle. 1035
xcolor: Uses \fboxrule and \fboxsep. 1031
xcolor: \fcolorbox etc. now work inside lateximage. 1031
Docs: Reorganized: Special cases and limitations. 118
Source: Improved formatting. . . . 1
tabular: Fix for babel-french. . . . 466
lateximage: Footnotes appear in regular text instead of the lateximage minipage. 534
v0.37
\@include: Maintains independent aux files for HTML. 235
General: 2017/08/19 1
LaTeX accents: Added. 247
babel-french: Adjustment for load order. 330

color: Prevented. 686
comment: Maintains independent cutfiles for print, HTML. 222
siunitx: Improved symbol support. 941
textcomp: Improved support. . . 973
lwarpmk: Removes additional HTML aux files. 299
File handles reorganized. 235
v0.38
\@seccntformat: Added for appendix. 370
\ForceHTMLPage: Added. 367
\ForceHTMLTOC: Added. 368
\LWR@section: \part* starts a new HTML page, for appendix. 370
Modified spacing, uses \numberline. 370
\numberline: Added trailing \quad. 495
\part: Fix with article class. 376
General: 2017/08/27 1
appendix: Added. 617
arabicfront: Added. 619
chappg: Added. 657
color: Forces xcolor as well. . . . 686
fix2col: Added. 727
fncychap: Added. 738
grffile: Added. 767
metalogo: Added. 837
nonumonpart: Added. 871
nopageno: Added. 871
pagenote: Option page disabled. 887
realscripts: Added. 917
relsize: Added. 920
romanbarpagenumber: Added. . . 922
romanbar: Added. 922
scalefnt: Added. 925
siunitx: Removed from lwarp core. 941
textcomp: Removed from lwarp core. 973
tocbibind: Added. 998
xltxtra: Added. 1046
lwarpmk: Added **print1** and **html1** actions. 299
Added \markboth, \sloppy, etc. . 328
Docs: Enhanced *Supported Features* table. 65
Docs: Index, tocbibind. 133
Docs: Starred sections. 129
v0.39
\@maketitle: titling version. 995
Native LaTeX version. 393
Removed minipages. 393, 995
Supports authblk with s of class oneauthor instead of tabular. 393, 995
\AddSubtitlePublished: Added. . . 394
\LWR@domulticolumn: Add: Optional vpos and # rows. 452
\LWR@restoreorigformatting: Appended with \appto instead of calling various macros. 503
\LWR@tabledatacolumntag: Don't start a data cell if see \TabularMacro. 461
\ResumeTabular: Added. 460
\TabularMacro: Added. 460
\multicolumnrow: Added. 458, 852
\printauthor: Removed minipages. 390
Supports authblk with s of class oneauthor instead of tabular. 390
\thanksmarkseries: Removed minipage footnotes. 996
General: 2017/09/05 1
a4wide: Added. 595
a4: Added. 594
a5comb: Added. 595
addlines: Added. 603
anysize: Added. 616
authblk: Added. 626
bigdelim: Added. 637
bigstrut: Added. 639
ebook: Added. 698
fullwidth: Added. 750
midpage: Added. 842
multirow: Add: New optional vpos argument. 851
multirow: Add: Supports left/right border for bigdelim. 851
multirow: Fix: Long text argument. 851
supertabular: Added. 967
textarea: Added. 972
titling: Improved compatibility. . 992
titling: Removed extraneous center environments. 993
typearea: Added. 1013
xtabular: Added. 1049
zwpagelayout: Added. 1054
Docs: Reorganized tabular discussion. 156
Titlepage \published and \subtitle removed. \AddSubtitlePublished restores. 394
titlepage: Clear pending footnotes. 388
Removed minipages. 388
titlingpage: Clear pending footnotes. 993
v0.40
\@chapcntformat: Added for tocbibind, anonchap. 370

\LWR@HTMLhline: Added. 464
\LWR@nullfonts: Fix: Long arguments for expandable command. 568
\LWR@restoreorigformatting: Improved LaTeX logos inside a lateximage. 503
Improved symbols inside a lateximage. 503
Nullified \InlineClass, etc. inside a lateximage. 503
\LWR@tabledatacolumntag: Fix for bigdelim: \ldelim, \rdelim. . . . 461
\chapter: Added support for quotchap. 377
\multicolumnrow: Fix: Adapts to older multirow and xparse. 458
\simplechapterdelim: Added for tocbibind, anonchap. 370
\underline: Added. 575
General: 2017/09/25 1
adjmulticol: Added. 602
anonchap: Added. 616
bigdelim: Improved documentation. 637
cuted: Added. 692
dblfnote: Added. 693
fnpos: Added. 739
graphics: Add: Full \graphicspath support. 762
graphics: Moved out of the lwarp core. 754
graphics: Restores \includegraphics and \DeclareGraphicsExtensions in a lateximage. 754
graphicx: Moved out of the lwarp core. 767
grffile: Directly supported. 767
midfloat: Added. 842
multirow: Improved bigdelim borders. 851
pfnote: Added. 902
quotchap: Added. 915
sectsty: Added. 934
stabular: Added. 957
tabls: Added. 970
textcomp: Additional symbols, improved XeLaTeX and LuaLaTeX support. 973
tocbibind: Improved for \simplechapter. 998
xfrac: No longer preloaded. 228
xltxtra: Fix for \showhyphens with XeLaTeX. 1046
v0.41
\LWR@addcmidruletrim: Add: \cmidrule trims. 442
\LWR@clearmidrules: Add: \cmidrule trims. 439
\LWR@closetabledatacell: Add: Mute > for \bottomrule. 416
Fix: At/bang column with \multirow. 416
Fix: Cancel < for \multicolumn. . 416
\LWR@domulticolumn: Add: \cmidrule trims. 452
Added vertical rules. 453
\LWR@nullifyNoAutoSpacing: babel-french: Fix: \NoAutoSpacing in a tabular. . . . 465
\LWR@parsebarcolumn: Added vertical rules. 424
\LWR@printatbang: Add: \cmidrule trims. 435
Add: Mute at and bang columns for \bottomrule. 435
\LWR@printbartag: Added vertical rules. 435
\LWR@subaddcmidruletrim: Added. 441
\LWR@subcmidrule: Add: \cmidrule trims. 440
\LWR@tabledatasinglecolumntag: Add: \cmidrule trims. 437
Add: Mute < for \bottomrule. . . 437
\LWR@tabularfinishrow: Unfinished tabular rows automatically filled. 418
\mcolrowcell: Added for \multicolumrow cells. 464
General: 2017/10/07 1
booktabs: Improved rules. 642
multirow: Add: \cmidrule trims. 851
Added vertical rules. 851
Fix: < spec. 852
tabular: Fix: \NoAutoSpacing in a tabular with babel-french. . . . 466
v0.42
\@ensuredmath: Improved \ensuremath. 517
\@textsubscript: Added. 574
\@textsuperscript: Added. 574
\LWR@HTMLhline: If FormatWP force explicit border. 464
\LWR@addformatwpalignment: If FormatWP add explicit style for cell alignment. 443
\LWR@addrulewidth: If FormatWP force explicit border. 442

\LWR@amsmathbody: Fix: Numbering and naming AMS math environments. 532
\LWR@amsmathbodynumbered: Fix: Numbering and naming AMS math environments. 532
\LWR@doequation: If FormatWP print LaTeX expression. 521
\LWR@domulticolumn: If FormatWP add cell alignment. 453
\LWR@doubledollar: If FormatWP print LaTeX expression. 515
Improved \ensuremath. 515
Improved line spacing with mathjax. 515
\LWR@floatbegin: If FormatWP add a text frame. 484
\LWR@floatend: If FormatWP add a text frame. 485
\LWR@htmlmathlabel: If FormatWP print LaTeX expression. 525
\LWR@rememberтag: Fix: Numbering and naming AMS math environments. 532
\LWR@restoreorigformatting: Improved \ensuremath. 504
\LWR@subaddcmidruletrim: 0pt if no rule given. 441
\LWR@subsingledollar: If FormatWP print LaTeX expression. 513
\LWR@tabledatasinglecolumntag: If FormatWP add cell alignment. . . 437
\LaTeX: If FormatWP use explicit style. 585
\TeX: If FormatWP use explicit style. 585
\hspace: If FormatWP add \quads. . . 580
\listoffigures: Added boolean WPMarkLOFT. 493
\listoftables: Added boolean WPMarkLOFT. 493
\marginpar: If FormatWP emulate a wrapfig. 354
\marginparBlock: If FormatWP emulate a wrapfig. 354
\rule: If FormatWP add \quads. . . . 583
\tableofcontents: Added boolean WPMarkTOC. 492
\underline: If FormatWP, use explicit styles for \underline, etc. 575
General: 2017/10/30 1
\textbf and related: If FormatWP, use explicit styles for \textsc, etc. 564
algorithmicx: If FormatWP add \quads. 608
booktabs: If FormatWP force explicit border. 642
epigraph: If FormatWP add HTML styles. 704
fancybox: If FormatWP add HTML styles. 713
floatflt: Added width. 731
graphics: Fix: Class key. 758
graphics: Fix: Filename expansion. 762
graphics: If FormatWP, use explicit size. 759
keyfloat: If FormatWP add explicit HTML style. 793
moreverb: Simplified formatting of listings. 844
multirow: If FormatWP add cell alignment. 852
overpic: Added. 886
realscripts: Fix for subscripts in a lateximage. 917
sidenotes: If FormatWP add explicit HTML style. 939
siunitx: Improved \ensuremath. . 942
soul: If FormatWP, add explicit styles. 952
textcomp: Improved \interrobangdown. 973
wrapfig: If FormatWP add explicit HTML style. 1029
Added boolean WPMarkLOFT. . . . 243
Added boolean WPMarkMath. . . . 244
Added boolean WPMarkMinipages. 243
Added boolean WPMarkTOC. 243
Added boolean WPTitleHeading. 244
Docs: Added support page. 2
Docs: Improper \prevdepth. . . . 186
Docs: Reorganized math limitations 145
File: lwarp_mathjax.txt: Updated siunitx script. 297
Fix: Numbering and naming AMS math environments. 531
If FormatWP, shift section headings. 244
tabbing: Added. 400
lateximage: Fix: Numbering and naming AMS math environments. 534
center: If FormatWP use explicit text-align. 539
minipage: Added boolean WPMarkMinipages. 557
If FormatWP add a text frame. . . . 555
eqnarray: Fix: Numbering and naming AMS math environments. 527
If FormatWP print LaTeX expression. 527

LWR@BlockClassWP: Added to factor code. 337
LWR@figcaption: If FormatWP forces italic captions. 488
v0.43
\LWR@domulticolumn: Fix for vertical rules. 453
Fix: Multicolumn trim. 452, 453
\LWR@maybeprintpendingfootnotes: Added FootnoteDepth. 353
\LWR@nullfonts: Fix: Nullify dollar inside filesnames. 568
\LWR@parsetablecols: Ignore spaces in col spec. 430
\LWR@printmccoltype: Added vertical rules. 448
\LWR@section: Fix: Expansion in comparison. 371
Fix: Math in section name. . 373, 375
Fix: Nullify fonts inside HTML comment. 372
\TabularMacro: \newcommand instead of \relax to fix supertabular and xtab. 460
\href: Made robust. 480
\nameref: Made robust. 478
\nolinkurl: Made robust. 481
\url: Made robust. 481
General: 2017/11/08 1
LWR@currentautosec: Added. . . . 370
breakurl: Added. 645
hyperref: Made robust. 772, 774, 775
hyperref: \Gauge added. 776
luatodonotes: Added. 812
todonotes: Added. 1008
Added FootnoteDepth. 349
Docs: HTML settings table. 106
Docs: Reorganized HTML customization. 106
v0.44
\@currentlabelname: Adjustment for koma-script. 471
\HTMLTitle: Added. 347
\LWR@addformatwpalignment: Fix for multicolumn alignment if FormatWP. 443
\LWR@backgroundcolor: Added. . . 1035
\LWR@filestart: Add \HTMLTitle. . 380
Fix \HTMLAuthor. 379
\LWR@listitem: Added list and trivlist. 404
\LWR@patchlists: Added list and trivlist. 407
\LWR@strresult: Fix: \providecommand. 414
\LWR@textcurrentcolor: xcolor: Added \LWR@textcurrentcolor. 1034
\addcontentsline: Automatic \LWR@newfloatanchor. 489
\chapter: Add preamble for koma-script. 377
\marginparBlock: Added. 354
\nopagecolor: xcolor: Fix for \nopagecolor. 1036
\part: Add preamble for koma-script. 376
\title: Added \thetitle. 347
General: 2017/11/22 1
algorithmicx: Improved comment symbol. 608
atbegshi: Added. 623
cancel: Added. 648
changepage: Additional options. 654
easy-todo: Added. 697
fancyref: Added. 718
fixmetodonotes: Added. 728
fixme: Added. 727
fontenc: Allowed after lwarp. . . . 742
hang: Added. 768
ifoddpage: Added. 778
ltxtable: Added. 811
luatodonotes: Improved. 812
lwarp-patch-komascript: Added. 1055
overpic: Fix: Groups for lateximages. 886
pdfsync: Fixes. 899
preview: Added. 911
scrextend: Added. 926
scrhack: Added. 928
scrlayer-notecolumn: Added. . . . 930
scrlayer-scrpage: Added. 930
scrlayer: Added. 928
section: Added. 932
soulpos: Added. 954
soulutf8: Added. 954
supertabular: Fix for caption. . . . 967
tikz: Fix: Groups for lateximages. 985
tocbasic: Added. 997
tocloft: Added \newlistentry. . 1005
tocloft: Improved \newlistof. . 1006
tocstyle: Added. 1007
todonotes: Improved. 1008
todo: Added. 1007
typearea: Added expert commands. 1013
watermark: Added. 1027
xcolor: Added \LWR@currenttextcolorstyle. 1034
xcolor: Added \LWR@findcurrenttextcolor. . 1034
xtab: Fix for caption. 1050

Adjustment for koma-script. . . . 207
AMS environments: Fix: Groups for lateximages. 609
If pdfLaTeX, require T1 and UTF-8 encoding. 208
picture: overpic: Fix: Groups for lateximages. 552
list: Added list and trivlist. . . 405
LWR@nestspan: Added list and trivlist. 332
v0.45
\@currentHref: Added. 479
\@donoparitem: Modified for HTML. 402
\@item: Modified for HTML. 403
\@mklab: Modified for HTML. 402
\CSSFilename: Improved filenames with underscores. 346
\LWR@LwarpStart: Fix: Lateximages on incorrect pages with Mathjax. 383
\LWR@newautoidanchor: Fix: No anchor if frozen autoid. 486
\LWR@notmemoirloadafter: Added. 196
\LWR@printpendingmpfootnotes: Added. 353
\LWR@startref: Fix: Labels with underscores. 477
\LWR@subhyperref: Improved URLs with underscores. 480
\LWR@subhyperrefclass: Improved URLs with underscores. 480
\LWR@sublabel: Fix: Labels with underscores. 474
\LWR@tabledatacolumntag: Fix: Empty line between rows. 463
\chapter: Add optional heading title for memoir. 377
\newpage: Added. 577
\nolinkurl: Fix: Underscore in URL. 481
\normalmarginpar: Added. 355
\reversemarginpar: Added. 355
\section: Add optional heading title for memoir. 377
\tableofcontents: Fix: Empty sidetoc. 492
Fix: Patch \AtBeginDocument. . . 492
\url: Improved URLs with underscores. 481
General: 2018/01/14 1
array: Added. 619
babel-french: Robust commands. 330
backref: Added. 630
breakurl: Fix: Underscore in URL. 645
changebar: Added. 653
cite: Added. 684
continue: Added. 688
endfloat: Added. 701
epigraph: Support for memoir. . . 704
fancyvrb: Improvements. . . . 719, 721
flafter: Added. 729
fltrace: Added. 737
footnpag: Added. 745
fwlw: Added. 750
graphics: Improved URLs with underscores. 762
hanging: Added. 770
hyperref: Fix: Underscore in URL. 772, 773
lwarp-patch-memoir: Added. . . 1058
memhfixc: Added. 837
memoir: Added. 588
natbib: Added. 858
pagesel: Added. 887
prettyref: Added. 911
subfigure: Added. 966
subfig: Fix for subcaption end tag. 964
subfig: Fix: Math in subcaptions. 962
textfit: Added. 976
titleref: Added. 989
turnthepage: Added. 1013
Allows memoir's preloaded packages. 199
Docs: xparse warnings. 166
Docs: Fix for double hyphens. . . . 79
Docs: Improved install instructions. 80
Docs: Improved MiKTEX install instructions. 76
Docs: Moved table so doesn't interfere with install docs. 75
File: lwarp_mathjax.txt: Allow MathJax inside tabbing. 297
File: lwarp_mathjax.txt: Allow MathJax inside verse. 297
Fix: Empty sidetoc. 492
Improved: Robust \, \ , and \textellipsis commands. 576
Separate LWR@thisautoidWP for word processor s. 486
thebibliography: Patched to emphasize titles. 503
minipage: Fix: Improper \prevdepth. 557
v0.46
\LWR@closeparagraph: Fix: Tabular empty lines. 342
\LWR@closeprevious: Fix: Stack unnesting. 328
\LWR@forcenewpage: Fix: Improper \prevdepth. 328
\LWR@lookforpackagename: Fix: Spaces in \usepackage. 231
\LWR@popclose: Fix: Stack unnesting. 319

\LWR@providelength: Added. 210
\LWR@pushclose: Fix: Stack unnesting. 318
\LWRPrintStack: Name changed from \PrintStack. 327
General: 2018/01/23 1
LWR@tabularpardepth added. . . 413
amsthm: Adapted to trivlist changes. 614
mdframed: Fixes for SVG math or lateximage in title. 829
mdframed: Fixes for footnotes. . 830
ntheorem: Adapted to trivlist changes. 873
theorem: Adapt to trivlist changes. 979, 981
tabular: Fix: Tabular empty lines. . 469
list: Fix: Stack unnesting. 405
v0.47
\LWR@HTML@caption@begin: Fix: Argument passed to \LWR@origcaption@begin. 488
\LWR@LwarpStart: Fix for SVG math in \nameref. 384
\LWR@WPcell: Fix: Line wrap at HTML hyphen. 443
\LWR@createautosec: Fix: Line wrap at HTML hyphen. 369
\LWR@domulticolumn: Fix: Line wrap at HTML hyphen. 452
\LWR@floatbegin: Fix: Line wrap at HTML hyphen. 484
\LWR@htmlclosecomment: Add \mbox to prevent line breaks. 333
\LWR@newautoidanchor: Fix: Line wrap at HTML hyphen. 486
\LWR@printopenlist: Fix: Line wrap at HTML hyphen. 402
\LWR@startref: Fix: Line wrap at HTML hyphen. 476
\LWR@sublabel: Fix: Line wrap at HTML hyphen. 475
\LWR@subsingledollar: Added SVG math image baseline adjust and em sizing. 513
\LWR@subsingledollarsvg: Fix: Line wrap at HTML hyphen. 511
\captionlistentry: Fix: Line wrap at HTML hyphen. 489
\hypertoc: Fix: Line wrap at HTML hyphen. 496
\hypertocfloat: Fix: Line wrap at HTML hyphen. 497
General: 2018/01/30 1
adjmulticol: Fix: Line wrap at HTML hyphen. 603
blowup: Added. 640
caption: Added. 649
caption: Also loads lwarp-caption. 234
changepage: Fix for pagecheck macros. 654
endheads: Added. 701
epigraph: Fix: Line wrap at HTML hyphen. 704
hanging: Fix: Line wrap at HTML hyphen. 770
hang: Fix: Line wrap at HTML hyphen. 768
keyfloat: Fix for SVG math in captions. 790
midpage: Fix: Line wrap at HTML hyphen. 842
multirow: Fix: Line wrap at HTML hyphen. 851
multitoc: Added. 853
ntheorem: Fix: Line wrap at HTML hyphen. 877
realscripts: Fix: Line wrap at HTML hyphen. 917
scrextend: Fix: Line wrap at HTML hyphen. 926
sectionbreak: Added. 933
sidenotes: Fix for SVG math in captions. 939
subfig: Fix for SVG math in captions. 962
subfig: Fix: Support \nameref. . . 962
xurl: Added. 1052
lwarpmk: *pdfcrop*: Removed hires option for improved crop accuracy. 299
lateximage: Added CSS style option. 534
Fix: Line wrap at HTML hyphen. . 538
center: Fix: Line wrap at HTML hyphen. 539
minipage: Fix: Line wrap at HTML hyphen. 555, 556
flushleft: Fix: Line wrap at HTML hyphen. 539
flushright: Fix: Line wrap at HTML hyphen. 539
enumerate: Fix: Line wrap at HTML hyphen. 406
itemize: Fix: Line wrap at HTML hyphen. 406
LWR@BlockClassWP: Fix: Line wrap at HTML hyphen. 337

v0.48
\@@@setcpageref: Fix for new v0.21 of cleveref. 551
\@@@setcref: Fix for new v0.21 of cleveref. 550
\@@@setcrefrange: Fix for new v0.21 of cleveref. 550
\@biblabel: Improved bibliography label. 502
\@item: Honors \makelabel. 403
\@maketitle: Fix: Errors with IEEEtran class. 393
\LWR@LwarpStart: Adjusted space around captions. 382
\LWR@ProvidesPackageDropB: Fix: Options with braces. 234
\LWR@addtabularhrulecolor: colortbl: Added. 444
\LWR@addtabularrulecolors: colortbl: Added. 445
\LWR@closetabledatacell: colortbl: Added. 416, 417
\LWR@lookforpackagename: Fix: Parsing similar package names. . 230
\LWR@newautopagelabel: Fix: TOC, LOF, LOT links. 360
\LWR@newhtmlfile: Fix: TOC, LOF, LOT links. 366
\LWR@nullfonts: Fix: \newline in title. 568
\LWR@parsedrequirepackagenames: Fix: Parsing similar package names. 229
\LWR@parsetablecols: Fix: Ignore optional tabular column arguments. 431
\LWR@restoreorigformatting: Fix: Spacing in SVG math, latexmage, Ti*k*z. 504
\LWR@section: Fix: TOC, LOF, LOT links. 375
\LWR@tabledatasinglecolumntag: colortbl: Added. 438
\LWR@textcurrentfont: Added. Improves font control. 571
\bibliography: Fix: \BaseJobname for bibliography. 502
\centerline: Added. 540
\l@part: Adapts to classes without \part. 498
\leftline: Added. 540
\mbox: Nullified for HTML. 558
\rightline: Added. 540
\thempfootnote: Removed \itshape. 353
General: 2018/02/14 1
acronym: Added. 601
acro: Added. 599
chapterbib: Added. 657
colortbl: Added. 433, 444, 686
fancyref: Now directly supported. 718
graphics: Fix: Virtual page size limited to a group. 762, 763
hypcap: Added. 771
hypernat: Added. 771
hyperref: \texorpdfstring now uses the TEX string. 775
luatodonotes: Improved \todototoc. 812
siunitx: Changes fraction to symbol. 944, 945
siunitx: Improved SVG math. 942, 943
siunitx: Improved color output. . 942
stfloats: Added. 960
todonotes: Improved \todototoc. 1008
vmargin: Added. 1023
xfrac: Fix: Added groups around super/subscripts to localize LWR@nestspan changes. 1044
Docs: Converting an existing document. 96
Improved font control. . . . 571, 572
tabular: colortbl: Added. 467
lateximage: Print mode boxed to natural width. 538
abstract: Allow optional name. . . 395
v0.49
\LWR@addtabularcellcolor: xcolor: Added tabular row colors. 447
\LWR@domulticolumn: xcolor: Added tabular row colors. 453
\LWR@printlength: Fix: Group printlen changes. 228
\affiliation: Fix: Adapts to classes which already provide. 388
\href: Fix: Adapt to classes. 480
\noalign: Fix: \noalign inside tabular. 464
\url: Fix: Adapt to classes. 481
General: 2018/02/19 1
amsmath: Fix: Patches for \eqref. 609
eso-pic: Fix for \AddToShipoutPicture. 710
figsize: Added. 726
fnlineno: Added. 739
hypdestopt: Added. 771
hyphenat: Added. 777
lineno: Added. 798
luacolor: Added. 811
pagegrid: Added. 886

pdfrender: Added. 899
resizegather: Added. 922
vertbars: Added. 1023
vwcol: Added. 1024
xcolor: Added tabular row colors. 433, 1040
Fix: Adapt to classes. 577
v0.50
\@ensuredmath: Fix: Use lateximage even if MathJax. 517
Improved SVG math alt tags. . . . 517
\LWR@@footnotetext: Robustify macros. 351
\LWR@atbeginverbatim: Improved column alignment. 399
\LWR@doequation: Improved SVG math display. 521
\LWR@doubledollar: Improved SVG math alt tags. 515
Improved SVG math display. 515
\LWR@htmlrefsectionfilename: Fix: SVG math in a section name. . . 325
\LWR@newhtmlfile: Fix: SVG math in a section name. 365
\LWR@nullfonts: Fix: \underline in sectioning file name. 570
\LWR@overline: Added. 575
\LWR@subsingledollar: Fix: Use lateximage even if MathJax. . . . 513
Improved SVG math alt tags. . . . 513
\LWR@subsingledollarsvg: MD5 hash avoids duplicate SVG math. 511
\LWR@vspace: Robustify macros. . . . 581
\newline: Robustify macros. 577
\textsubscript: Robustify macros. 574
\textsuperscript: Robustify macros. 574
General: 2018/03/03 1
lwarp.css: Improved SVG display math centering. 256
lwarp_one_limage.txt: Added. . 296
amsmath: Fix: Upright tags for SVGmath. 609
axodraw2: Added. 630
bytefield: Added. 647
dblfloatfix: Added. 693
diagbox: Added. 694
epstopdf: Added. 706
listings: Force flexible columns. . 801
morefloats: Added. 843
nonfloat: Added. 871
ntheorem: Fix: Not standard nor amsthm selected. 880
pbox: Added. 893
phfqit: Added. 902
schemata: Added. 925
siunitx: Fix: Loads xcolor. 941
siunitx: Improved SVG math alt tags. 943
siunitx: Improved units. 543, 942, 945
xy: Added. 1052
lwarpmk: Error if lateximages.txt does not exist. 299
lwarpmk: Error if lwarpmk.conf points to lwarp. 299
lwarpmk: Improved error messages. 299
lwarpmk: MD5 hash avoids duplicate SVG math. 299
lwarpmk: Multiprocess support making lateximages. 299
AMS environments: Improved SVG math display. 609
Fix: Load fontspec if necessary. . 223
Robustify macros. 572
lateximage: Fix: SVG math in a section name. 537
MD5 hash avoids duplicate SVG math. 535, 538
eqnarray: Improved SVG math display. 527, 528
v0.51
\@ensuredmath: Hashes \ensuremath. 517
\@item: Restored list label space. . . 404
\LWR@HTMLsanitizeexpand: Fix: Escapes double quotes. 361
\LWR@LwarpStart: MathJax: Nullifies \ensuremath. 384
\LWR@addbaselinemarker: Improved SVG math baseline. 507
\LWR@atbeginverbatim: Adds vertical offset. 398
\LWR@doequation: Fix: \addcontentsline inside SVG math. Provides an autoid anchor. 521
\LWR@doubledollar: Fix: \addcontentsline inside SVG math. Provides an autoid anchor. 515
\LWR@findcurrenttextcolor: Added \LWR@findcurrenttextcolor when no xcolor. 575
\LWR@newautoidanchor: Fix: No autoid is inside a lateximage. . . 486
\LWR@singledollarmeasure: Fix: lateximage inside $\mathcal{AMS}$ \text. . 509
Fix: Honors text font around SVG math. 509
Improved SVG math baseline. . . . 509

Typeset SVG math only once during measurement. 509
\LWR@subHTMLsanitize: Fix: Escapes double quotes. 360
\LWR@subsingledollar: Fix: \ensuredmath inside SVG image. 514
\LWR@subsingledollarsvg: Fix: SVG math with enclosed lateximage. 510
SVG math baseline improved with invisible rule at corner. 512
\LWR@textcurrentcolor: xcolor: \LWR@textcurrentcolor if xcolor not loaded. 575
\addcontentsline: Add missing support for float mechanism if necessary. 490
No anchor ID if inside SVG image. 490
\displaymathnormal: Processing for complicated display math. 524
\displaymathother: Processing for complicated display math. 524
General: 2018/03/24 1
lwarp_one_limage.txt: *pdftocairo* -noshrink added. 296
afterpackage: No longer required. 226
chemfig: Added. 658
chemformula: Added. 659
chemgreek: Added. 664
chemmacros: Added. 665
chemnum: Added. 683
epstopdf-base: Added. 706
fancybox: Fix: Optional tag for \item in a span. 716
grid: Added. 767
listings: Forces cleared options. . 802
ltxgrid: Added. 810
mhchem: Added. 839
tikz: Fix for \tikz macro. 985
tikz: Fix for tikz with optional argument. 985
titling: Fix for \thanks mark. . . . 994
lwarpmk: *pdfcrop*: Restored hires option. 299
lwarpmk: *pdftocairo* -noshrink added. 299
AMS environments: Fix: \addcontentsline inside SVG math. Provides an autoid anchor. 609
Docs: tikz limitations. 153
Docs: Multiple authors and affiliations. 129
Docs: Things to avoid. 118
Docs: Updated Converting an existing document. 96
Fix: Remember original \# in case is redefined. 246
MathJax: Nullifies \ensuremath. . 362
Named HTML entity used for text dollar. 507
lateximage: Added additional hashing option. 534
Fix: lateximage inside $\mathcal{AMS}$ \text. 534
Processing for complicated display math. 537
alignat: Fix: Added. 612
eqnarray: Fix: \addcontentsline inside SVG math. Provides an autoid anchor. 527, 528
LWR@displaymathother: Processing for complicated display math. . . 518
LWR@equationother: Processing for complicated display math. . . . 518
v0.52
\@ensuredmath: Improved hashing expansion. 517
\@mpfootnotetext: Fix: Paragraph handling. 352
\CustomizeMathJax: Added. 362
\LWR@@footnotetext: Fix: Paragraph handling. 351
\LWR@addbaselinemarker: Warnings if lwarp_baseline_marker.png is not present or if graphicx/s not loaded. 507
\LWR@customizedMathJax: Added. . 361
\LWR@doequation: Fix: equation* now based on equation instead of displaymath. 521
Fix: equation* with split. 521
\LWR@filenamenoblanks: Fix: \FileDepth with non-utf8 encoding. 359
\LWR@nullfonts: Fix: \texorpdfstring in section names. 571
\LWR@section: Fix: Footnote numbering: Limited HTML comment if starred. 372
Fix: Footnote numbering: Use short TOC entry for HTMLDebug comments. 372
\LWR@singledollarmeasure: Added user-adjustable SVG math font scaling. 510
\LateximageFontScale: Added user-adjustable SVG math font scaling. 530
\href: Fix: #, %, &, ~, _ in URL. . . . 480

\nolinkurl: Fix: #, %, &, ~, _ in URL. 481
\theHTMLTitleSeparator: Fix: \FileDepth with non-utf8 encoding. 378
\url: Fix: #, %, &, ~, _ in URL. 481
General: 2018/04/01 1
breakurl: Fix: #, %, &, ~, _ in URL. . 645
endfloat: Updated for v2.6. 701
fancyvrb: Initial support for \VerbatimFootnotes. 713, 719
graphics: Added defaults. . . . 756, 757
graphics: Updated for v1.1a. 757
graphics: Updated for v1.1b. 757
hyperref: Fix: #, %, &, ~, _ in URL. 772–774
nicefrac: Added. 869
url: Added. 1020
lwarpmk: Fix: Memory overflow when spawning tasks. 299
lwarpmk: Fix: Skip image generation if from page 0. 299
Changed FootnoteDepth default to \subsbusection. 349
Docs: Improved install instructions. 77
Fix: MathJax script line wraps. Reduced right margin. 224
If pdfLaTeX, allow other input encoding. 208
Restore \kill in a lateximage. . 809
tabbing: Fix to allow inside lateximage. 400
lateximage: Fix for hash expansion. 535
v0.53
General: 2018/04/01 1
lwarpmk: Added **lwarpmk cleanlimages**. 299
lwarpmk: Added warning for corrupted images. 299
Docs: **lwarpmk cleanlimages**. . . . 93
Docs: **lwarpmk pdftohtml**. 93
v0.54
\LWR@afterendverbatim: Added vspace argument. 399
\LWR@atbeginverbatim: Improved column alignment. 399
\LWR@earlyloadnever: Added. 198
\LWR@endfloatalignment: Honor \centering, etc. in floats. 487
\LWR@floatalignment: Honor \centering, etc. in floats. 486
\LWR@floatbegin: Honor \centering, etc. in floats. 484
\LWR@floatend: Honor \centering, etc. in floats. 485
\LateximageFontSizeName: Defaults to normalsize. 530
\centering: Added debug comment. 539
\raggedleft: Added debug comment. 540
\raggedright: Added debug comment. 540
General: 2018/04/22 1
*.lwarpmkconf: Option IndexLanguage changed to xindyLanguage. 256
*.lwarpmkconf: Option pdftotextEnc added. 256
*.lwarpmkconf: Option xdyFilename changed to xindyStyle. 256
*.lwarpmkconf: Option xindyCodepage added. 256
lwarp.css: Fix: Text-decoration-skip: auto. . . . 256
lwarpmk.conf: Option IndexLanguage changed to xindyLanguage. 256
lwarpmk.conf: Option pdftotextEnc added. 256
lwarpmk.conf: Option xdyFilename changed to xindyStyle. 256
lwarpmk.conf: Option xindyCodepage added. 256
bibunits: Added. 636
chngpage: Added. 684
forest: Added. 746
glossaries: Fix when not using babel or polyglossia. 753
gridset: Added. 768
hyperref: Fix: \hyperref and \hyperlink with special chars in text. 774
hyperref: Fix: \ref in \hyperref and \hyperlink caused nested link. 774
lwarp-patch-memoir: Update for v3.7g. 1062
magaz: Added. 816
ragged2e: Fix: \centering, etc. . 917
textcomp: Fix for \textperthousand. 973
tikz: Fixes for \pgfpicture, minipages, fit, align, font. 985
lwarpmk: Added pdftotextenc. . 299
lwarpmk: Added xindycodepage. 299
lwarpmk: Changed language to xindylanguage. 299

lwarpmk: Changed `xdyfile` to `xindystyle`. 299
lwarpmk: Improved error if configuration file does not exist. 299
lwarpmk: Increased prominence for error for an unknown command. 299
lwarpmk: Verifies HTML version exists before `lwarpmk limages`. . 299
lwarpmk: Verifies image references before `lwarpmk limages`. 299
Add: `pdftotextEnc`. 218
Add: `xindyCodepage`. 217
Added early check for disallowed packages. 198
Docs: BibTeX. 131
Docs: Macros in sectioning names. 118
Never load aecompl. 198
Option `IndexLanguage` changed to `xindyLanguage`. 217
Option `xdyFilename` changed to `xindyStyle`. 217
`verse`: Fix: Line spacing. 397
v0.55
`\LWR@LwarpStart`: Fix: Overfull boxes in `lateximages`. 382
`\LWR@floatbegin`: Fix: Float optional args. 484
`\LWR@nullfonts`: Removed extraneous space which appeared in file links. 570
`\phantomsection`: Fix: `\ForceHTMLTOC` with `\phantomsection`. 584
General: 2018/04/26 1
clrdblpg: Added. 685
Fix: `\centering`, etc. for koma-script. 484
Fix: QED symbols in `lateximage`. 615, 884
v0.56
`\LWR@addcdashline`: arydshln: Added. 443
`\LWR@addmulticolvertrulecolor`: Adds support for dashed vertical rules. 451
Adds support for double vertical rules. 451
`\LWR@addtabularhrulecolor`: Adds support for arydshln dashed rules. 444
Adds support for double `\hlines` and `\midrules`. 444
`\LWR@addtabularrulecolors`: Adds support for dashed vertical rules. 445
Adds support for double vertical rules. 445
`\LWR@closeparagraph`: Added support for parnotes. 342
`\LWR@domulticolumn`: Adds support for dashed vertical rules. 453
Adds support for double vertical rules. 453
`\LWR@floatbegin`: Adds a `<class>` per float package style. 484
`\LWR@multicolpartext`: Fix: `\multicolumn` parameters. . . . 448
`\LWR@openparagraph`: Added support for parnotes. 340
`\LWR@parsebarcolumn`: Adds support for double vertical rules. 424
`\LWR@parsecoloncolumn`: arydshln: Added. 424
`\LWR@parsesemicoloncolumn`: arydshln: Added. 425
`\LWR@parsetablecols`: Added array `W` column. 432
`\LWR@printmccoldata`: Added array `W` column. 450
Fix: `\multicolumn` parameters. . . 449
`\LWR@printmccoltype`: Added array `W` column. 448
Adds support for dashed vertical rules. 448
Adds support for double vertical rules. 448
`\LWR@tabledatacolumntag`: Fix: `\morecmidrules` 462
`\LWR@textcurrentfont`: Added `span.textbf`, etc. 571
General: 2018/05/12 1
`*.lwarpmkconf`: Records `--shell-escape`. 256
`lwarp.css`: Added `div.textbf`, etc. 256
`lwarp.css`: Added `span.textbf`, etc. 256
`lwarpmk.conf`: Records `--shell-escape`. 256
arydshln: Added. 410, 620
lua-check-hyphen: Added. 811
paralist: Fixes for `compactenum`, `compactitem`, `compactdesc`. . . . 887
parnotes: Added. 891
quoting: Added. 916
tocenter: Added. 1000
underscore: Added. 1017
lwarpmk: Added `lwarpmk pdftosvg`. 299

lwarpmk: Supports `--shell-escape`. 299
Added `\thinspace`. 576
Docs: **`lwarpmk pdftosvg`**. 93
`LWR@blocktextcurrentfont`: Added `div.textbf`, etc. 572
v0.57
`\BlockClassSingle`: Improved print/HTML output selection. . . 337
`\InlineClass`: Improved print/HTML output selection. 337
`\LWR@ref@ignorestar`: subcaption: Fix: `\subref`. 478
`\LWR@singledollarmeasure`: Fix: Dynamic inline math expressions. 509
`\LWR@subhyperref`: Fix: Text catcodes. 480
`\LWR@subhyperreftext`: Fix: Text catcodes. 480
`\LWR@subsingledollar`: Fix: Dynamic inline math expressions. 514
`\LWR@subsingledollarsvg`: Fix: Dynamic inline math expressions. 511
`\LWR@vspace`: Improved print/HTML output selection. 581
`\MathImageAltText`: Added. 505
`\PackageDiagramAltText`: Added. . 506
`\StartDefiningMath`: Added. . . . 321
`\boxframe`: xcolor: Fix: Colored `\boxframe`. 1040
`\colorbox`: xcolor: New system for switching print and HTML outputs. 1037
`\colorboxBlock`: xcolor: New system for switching print and HTML outputs. 1037
`\fboxBlock`: Improved print/HTML output selection. 561
`\fcolorbox`: xcolor: New system for switching print and HTML outputs. 1037
`\framebox`: Improved print/HTML output selection. 560
`\href`: Fix: Text catcodes. 480
`\inlinemathother`: Added. 322
`\listof`: Fix: Provide `\l@name` if not defined. 494
`\makebox`: Improved print/HTML output selection. 559
`\mbox`: Improved print/HTML output selection. 558
`\multicolumnrow`: multirow: Improved print/HTML output selection. 852
Improved print/HTML output selection. 458
`\newfloat`: rotfloat: Added float styles. 924
rotfloat: Fix for listof sideways floats. 924
`\parbox`: Improved print/HTML output selection. 558
`\raisebox`: Improved print/HTML output selection. 563
`\reflectbox`: Improved print/HTML output selection. 766
`\resizebox`: Improved print/HTML output selection. 766
`\rotatebox`: Improved print/HTML output selection. 764
`\rule`: Fix: Colored rules. 582
`\scalebox`: Improved print/HTML output selection. 765
`\textcolor`: xcolor: New system for switching print and HTML outputs. 1036
General: 2018/06/06 1
`lwarp.css`: Added ruled, boxed, boxruled floats. 256
`lwarp.css`: Increased float vertical margins. 256
algorithm2e: Added. 604
bigdelim: Improved print/HTML output selection. 637
breakurl: Fix: Text catcodes. 645
colortbl: New system for switching print and HTML outputs. . . 686, 687
ellipsis: Added `\midwordellipsis`. 699
errata: Added. 708
float: Added float styles. 730
float: Fix: Do not pre-define `\l@name`. 730
ltablex: Added. 809
marginnote: Fix: Long optional argument. 820
multirow: Improved print/HTML output selection. 851
register: Added. 919
subcaption: Fix: `\subref`. 773
trimclip: Added. 1011
vowel: Added. 1024
xellipsis: Added. 1042
xfrac: Improved print/HTML `\scalebox` control. 1044
xltabular: Added. 1046
xpiano: Added. 1047
lwarpmk: Improved code factoring. 299
lwarpmk: Improved error handling. 299

Docs: Recompiling *lwarpmk* or CSS files. 182
Docs: Recreating the index for lwarp source. 179
MathJax: Supports \footnote, \footnotemark. 362
New system for switching print and HTML outputs. 240
minipage: Improved print/HTML output selection. 555
BlockClass: Improved print/HTML output selection. 336
fminipage: Improved print/HTML output selection. 561
LWR@BlockClassWP: Improved print/HTML output selection. . . 337
v0.58
\LWR@HTML@caption@begin: Improved print/HTML output selection. . . 488
\LWR@HTML@caption@end: Improved print/HTML output selection. . . 489
\LWR@HTML@ref: Improved print/HTML output selection. 477
\LWR@doindexentry: Adds support for \see, \seealso, \emph, \textbf, etc. 501
\LWR@hyperindexrefnullified: Adds support for \see, \seealso, \emph, \textbf, etc. 501
\LWR@indexitem: Accepts optional arg for repeatindex. 500
\dotfill: Improved print/HTML output selection. 577
\hfill: Improved print/HTML output selection. 577
\hrulefill: Improved print/HTML output selection. 577
\hyperindexref: Adds support for \see, \seealso, \emph, \textbf, etc. 501
\printindex: Fix: Extra \newpage to flush pending \index writes. . . . 816
General: 2018/07/07 1
*.lwarpmkconf: Added option makeindexstyle. 256
*.lwarpmkconf: Added options makeindex and xindy. 256
*.lwarpmkconf: Generated \AtBeginDocument. 256
lwarp.xdy: Requires makeindex.xdy. 295
lwarp.xdy: Supports bold, italic. 295
lwarp_html.ist: Added. 294
lwarpmk.conf: Added option makeindexstyle. 256
lwarpmk.conf: Added options makeindex and xindy. 256
lwarpmk.conf: Generated \AtBeginDocument. 256
array: Improved print/HTML output selection. 619
attachfile2: Added. 624
attachfile: Added. 623
cases: Added. 652
imakeidx: Added. 779
index: Added. 783
intopdf: Added. 785
lwarp-patch-komascript: Modified indexing. 1055
lwarp-patch-memoir: Fix for \specialindex. 1078
lwarp-patch-memoir: Fix for multiple indexes. 1079
makeidx: Added. Moved from lwarp core. 816
memoir: Fix for \firsthlline, \lasthline. 460
memoir: Fix for booktabs. 464
pdfpages: Added. 897
pdfx: Added. 900
repeatindex: Added. 921
splitidx: Added. 955
textcomp: Improved print/HTML output selection. 973
lwarpmk: Added makeindex and xindy options. 299
lwarpmk: Added -p option for project name. 299
lwarpmk: Added optional list of names for **lwarpmk printindex** and /cmdslwarpmk htmlindex. . 299
lwarpmk: Glossary generation now uses *makeglossaries*. 299
lwarpmk: **lwarpmk clean** removes all *.ind and *.idx files. 299
Added makeindex option. 219
Added xindy option. 219
Added option makeindexStyle. . 217
Docs: Index, *makeindex*, imakeidx. 133
Docs: Misplaced \omit. 186
Fix: memoir and ccaption. 199
Improved print/HTML output selection. 576
Replaced each \csuse with \@nameuse to force error if undefined. 1
tabbing: Improved print/HTML output selection. 400

v0.59
\LWR@addbaselinemarker: Uses .eps if DVI *latex*. 507
\LWR@latexmkcmd: Fix: --shell-escape with *latexmk*. . . 251
\LWR@writeconf: Compilation commands now preassigned by lwarp instead of being computed by *lwarpmk*. 255
\[: Fix with \displaymathnormal. . 516
General: 2018/09/07 1
Slunits: Added. 940
accsupp: Added. 599
amsmath: Moved from the lwarp core. 609
asymptote: Added. 622
axessibility: Added. 629
breqn: Added. 645
bxpapersize: Added. 647
canoniclayout: Added. 649
chemformula: Fix for \NMR. 680
draftcopy: Added. 696
epstopdf-base: Improved. 706
epstopdf: Improved. 706
fnbreak: Added. 738
graphics: Fix: Expand filename. . 762
graphics: Now works with .pdf and .eps filename extensions. 762
nccfancyhdr: Added. 859
pdftricks: Added. 900
pst-eps: Added. 913
pstricks: Added. 914
units: Added support for MathJax. 1018
xunicode: Added. 1051
lwarpmk: Added **lwarpmk epstopdf**. 299
lwarpmk: Consolidated compiling options into printlatexcmd and HTMLlatexcmd. 299
lwarpmk: Double insead of single-dashed --shell-escape option. 299
lwarpmk: Error if lwarpmk.conf format changed. 299
lwarpmk: Warning if operating system changed. 299
Added option dvipdfmx. 219
Added option dvipdfm. 219
Added option dvips. 219
Docs: **lwarpmk epstopdf**. 93
File: lwarp_mathjax.txt: Fix: Removed chapter number from tagged non-numeric MathJax equations. 297
File: lwarp_mathjax.txt: Updated to MathJax v2.7.4. 297
picture: Added an alt tag. 552
v0.60
\LWR@clearmidrules: tabular: Fix for midrules. 439
\LWR@parsenormalcolumn: tabular: Improved memory management: Not using xstring. 425
\LWR@tabledatasinglecolumntag: tabular: Improved memory management: Not using xstring. 437
\LWR@tabularendofline: Fix: Slowdown for long tables. 420
General: 2018/09/19 1
tabular: Improved memory management: Global boolean. 411, 412
tabular: Improved memory management: Not using xstring. 413, 851
2up: Added. 594
booklet: Added. 640
bophook: Added. 643
diagbox: Fix for par tags. 695
draftfigure: Added. 697
fancytabs: Added. 719
fullminipage: Added. 749
grid-system: Added. 768
layaureo: Added. 794
leading: Added. 797
listings: Fix for HTML entities. . . 802
listings: Fix if inside a list. . . 802, 805
thumbs: Added. 984
thumb: Added. 984
widows-and-orphans: Added. . 1028
v0.61
\DeclareGraphicsExtensions: Fix: EPS for DVI LaTeX. 754
\LWR@HTMLLatexCmd: Added HTMLLatexCmd option. 255
Added PrintLatexCmd option. . . 255
\LWR@addcompilecmd: Removed spaces. 250
\LWR@closetabledatacell: Fix: Par tags in tabular. 416
\LWR@hyperindexrefnullified: Made robust, ... 501
\LWRopseq: Added spaces. 215
\RequirePackage: Support up to 20 packages. 232
\inlinemathnormal: Changed name from \StopDynamicMath to \inlinemathnormal. 322

\inlinemathother: Changed name from \StartDynamicMath to \inlinemathother. 322
\listof: Fix: newfloat lists. 494
\lwarpsetup: Added. 216
General: 2018/10/13 1
lwarp.css: Footnotes text align left. 256
lwarp.css: Minipage table and footnotes: tighter margin. 256
chkfloat: Added. 684
cmdtrack: Added. 685
copyrightbox: Added. 688
dprogress: Added. 696
epsfig: Added. 705
graphics: Fix: EPS for DVI LaTeX. . 759
graphics: Set keys before using filename, for epsfig. 762
lua-visual-debug: Added. 811
pdfprivacy: Added. 899
psfragx: Added. 912
psfrag: Added. 912
pstool: Added. 913
refcheck: Added. 918
srcltx: Added. 956
srctex: Added. 957
supertabular: Fix for caption w/o opt arg. 967
thinsp: Added. 981
threadcol: Added. 981
uspace: Added. 1020
vpe: Added. 1024
xbmks: Added. 1031
xtab: Fix for caption w/o opt arg. 1050
Added HTMLLatexCmd option. . . . 218
Added PrintLatexCmd option. . . 218
Docs: \tracinglwarp 237
Docs: HTML entities. 119
Docs: Compiling using custom shell commands. 168
Docs: Fonts. 98
Docs: HTMLDebugComments 106, 237
Docs: Multiple indexes. 191
Don't write configuration files if processing pstool image. 249
Spaces redefined \AtBeginDocument. 576
v0.62
\@partcntformat: Added for ctex. . 370
\@partnameformat: Added for ctex. 370
\InlineClass: Added optional word-processing style. Replaces \LWR@HTMLtextstyle. 337
\LWR@PreloadedPackage: Added. . . 540
\LWR@ProvidesPackagePass: Fix: Unknown option error. 233
\LWR@endofline: Extra space if optional arg. 578
\LWR@filestart: Refactored. 380
\LWR@isolate: Added. 211
\LWR@textcurrentfont: Added print version. 575
Tracks depth to avoid nesting repeated font changes. 571
\colorboxBlock: Fix: Horiz white space. 1037
\fcolorbox: Fix: No longer requires xifthen. 546
\fcolorboxBlock: Fix: Horiz white space. 1038, 1039
\l@chapter: Don't define if no \chapter. Fix for algorithm2e. . . 498
\slshape: Added. 573
\textup: Fixed WP span class. 566
\theHTMLSection: Added. 379
\theHTMLTitleSection: Added. . . . 378
\theHTMLTitleSeparator: Refactored. 378
General: 2018/11/19 1
\textbf and related: Improved font detection. 564
lwarp.css: Added css for xfrac, nicefrac. 256
lwarp.css: Fixed css for \textup. 256
lwarp.css: Reduced margins in titlepage. 256
lwarp_formal.css: Fix: Font for verse. 290
2in1: Added. 594
CJKutf8: Prevented unless xeCJK. 685
CJK: Prevented unless xeCJK. . . . 685
asymptote: Improved alt tags. . 622
bitpattern: Added. 639
calc: Fix: Required for print version. 226
chngpage: Fix: Loads lwarp-chngpage. 684
ctexpatch: Added patch. 589
flippdf: Added. 729
graphics: Fix: Filename expansion. 760
graphics: Fix: FormatWP. 759
musicography: Added. 854
nicefrac: Improved font control and css, honors nice, ugly. 869
notespages: Added. 872
octave: Added. 884
pdfcomment: Added. 895
pdfmarginpar: Added. 896
register: Updated to v1.8. 919

rviewport: Added. 924
semantic-markup: Added. 934
textcomp: Fix conflict with xunicode. 975
tram: Added. 1010
twoup: Added. 1013
ulem: Improved compatibility with CJKulem. 1015
ulem: Now works in a lateximage. 1015
unitsdef: Added. 1019
units: Improved font control and CSS, honors loose, tight. 1018
xechangebar: Added. 1042
xfrac: Improved CSS. 1044
xunicode: Fix conflict with textcomp. 1051
Added early checks for CJK, CJKutf8. 198
Docs: asymptote. 155
Docs: miktex-poppler-bin-* 80
Docs: *MiKTeX Console* 76
Docs: Improved MiKTEX install instructions. 76
Docs: UTF-8 locale. 171
File: lwarp_mathjax.txt: Removed inoperable siunitx extension. . . . 297
Fix for \em. 568
Fix: Horiz white space. 1039
Logos: CSS instead of <sup>, <sub>. 584
Logos: Fix for XeTEX logo if graphics is not loaded. 584
Logos: Improved CSS. 584
Logos: Made robust. 584
fcolorminipage: Fix: Horiz white space. 1040
Fix: No longer requires xifthen. . 547
fminipage: Fix: Horiz white space. 562, 563
LWR@blocktextcurrentfont: Added print version. 576
v0.63
\LWR@HTMLLatexCmd: ujarticle and related: Compile options. 254
\LWR@LwarpStart: Fixes for xeCJK. . 381
\LWR@atbeginverbatim: Fix for xeCJK. 398
\LWR@checkloadbefore: Added. . . . 197
\LWR@checkloadfilename: Added to reduce number of lwarp-* files. 229
\LWR@checkloadnever: Added. 197
\LWR@compileuplatex: Added. 251
\LWR@createautosec: Fix for xeCJK. 369
\LWR@earlyclassloadnever: Added. 198
\LWR@filestart: Fix: Break ligature for luatexko. 380
\LWR@firstoffour: Added. 212
\LWR@htmlclosecomment: Fix: Break ligature for luatexko. 333
\LWR@isolate: Fix for xeCJK. 211
\LWR@notltjloadafter: Added more classes. 196
Added. 196
\LWR@subhtmlelementclass: Fix for xeCJK. 334
\LinkHome: Fix: Print version. . . . 326
\linkhomename: Added. 325
General: 2018/12/03 1
lwarp.css: Added CSS for vertical writing. 256
lwarp.css: Improved CSS for mdframed. 256
amsthm, mdframed: Fix for enforced load order. 613
emumitem: v3.6: Nullifiy \DrawEnumitemLabel. 703
everyhook: Fix for bxjs* classes. . 226
geometry: Fix for bxjs* classes. . . 224
mdframed: Avoid thin rules. . . . 828
mdframed: Improved font control. 831–833
stfloats: Adapted to ltj* classes. . 960
xpinyin: Added. 1048
zhlineskip: Added. 1053
Added pTEXsupport. 195
Docs: \linkhomename. 106
Docs: \sidetocname. 108
Fix: Default \LWR@mdfive. 208
Improved titles. 830, 831
pTEX: Encoding. 208
pTEX: Load upquote. 210
pTEX: No newunicodechar. 209
LWR@BlockClassWP: Fix for xeCJK. . . 337
v0.64
\LWR@HTMLLatexCmd: utarticle and related: Added. 254
\LWR@checkloadfilename: Prevented bitfield, doublespace, newthm, rplain, si. 229
\LWR@section: Support for ujarticle and related. 373
\enskip: Made robust. 579
\qquad: Made robust. 578
\quad: Made robust. 578
\theHTMLTitleSeparator: Added utarticle and related. 378
General: 2018/12/08 1
addlines: Updated to v0.3. 603
biblatex: Added patch for CTEX. . 636

bsheaders: Added. 647
gmeometric: Added. 754
marginal: Added. 819
rmpage: Added. 922
scrlayer-scrpage: Fixes. 931
scrlayer: Fixes. 930
scrpage2: Added. 931
ujarticle and related: Improved \today. 588
Added utarticle and related. 588
v0.65
\LWR@LwarpEnd: Improved css for page layout. 385
\LWR@LwarpStart: Improved css for page layout. 383
\LWR@PreloadedPackage: \AtBeginDocument to avoid option clashes. 540
\LWR@hyperindexrefnullified: Added \textsi. 501
\LWR@newhtmlfile: Error if duplicate file name. 364
Improved css for page layout. 363, 366
\LWR@nullfonts: Added \sishape. . 568
Added \textsi. 568
\LWR@restoreorigformatting: Fix: tabular*. 504
\enskip: Changed to Unicode EN SPACE. 579
\quad: Changed to Unicode EM SPACE. 578
\sishape: Added \sishape. 573
\textsi: Added. 567
General: 2018/12/22 1
lwarp.css: Added \sishape, \textsi. 256
lwarp.css: Improved css for page layout. 256
lwarp.css: Improved css for quotations. 256
lwarp.css: SideTocto left for improved \marginpars. 256
lwarp_formal.css: SideTocto left for improved \marginpars. . . . 290
lwarp_sagebrush.css: SideTocto left for improved \marginpars. . 285
bounddvi: Added. 644
embrac: Added. 700
footnoterange: Added. 745
gentombow: Added. 750
geometry: Fix for bxjs* classes. . . 224
graphics: Added \includegraphics alt key. 545, 755, 757, 758, 761, 762
lltjext: Added. 805
multicolrule: Added. 848
multicol: Added \docolaction. . 848
plarydshln: Added. 909
plextarydshln: Added. 910
plextcolortbl: Added. 911
plext: Added. 909
pxatbegshi: Added. 914
pxeveryshi: Added. 914
pxftnright: Added. 915
pxjahyper: Added. 915
tascmac: Added. 971
versonotes: Added. 1022
Added early checks for jarticle, tarticle, and related. 198
Fix for \rensuji. 588
Fix space between class and id. . 338
tabular: Added support for plext. . 466
Fix: tabular*. 466
Fix: Rule color. 467
minipage: Refactored to later allow Japanese <t/y> argument. 555
LWR@figcaption: Uses <figurecaption> instead of <figcaption>. 488
v0.66
\@mpfootnotetext: Improved HTML formatting. 352
\IgnoreMinipageWidths: Added, . . 554
\LWR@@footnotetext: Improved HTML formatting. 351
\LWR@LwarpStart: Fix: TOC, LOF, LOT links. 384
\LWR@checkloadfilename: Prevented colortab, epsf, hyper, picinpar, picins, sistyle, ucs. 229
\LWR@closeparagraph: Fix: Combined span, tabular, and lateximage. 342
Improved HTML formatting. 341
\LWR@closeparagraph@br: Factored. 340
\LWR@fboxstyle: Use current text color. 560
\LWR@filenamenoblanks: Fix: Section names detokenized. 357
Fix: Section names with macros. 358
Fix: Section names with percent. 358
Improved file name generation. . 357
Limits filename length. 359
\LWR@findcurrenttextcolor: Fix: Color if xcolor not loaded. 575
\LWR@htmlfileref: No longer use zref. 473
\LWR@htmlsectionfilename: Sanitize underscores. 324
\LWR@indentHTML: Added. 331
\LWR@lateximagedepthref: No longer use zref. 474

\LWR@lateximagenumberref: No longer use zref. 474
\LWR@lwarplabel: No longer use zref. 474
\LWR@nameref: No longer use zref. . 473
\LWR@nullfonts: Logos. 570
\LWR@openparagraph: Improved HTML formatting. 340
\LWR@section: Fix: TOC, LOF, LOT links. 375
Improved HTML formatting. 375
\LWR@setexparray: Fix with \par. . 320
\LWR@setref: No longer use zref. . . 473
\LWR@simplifyname: Added. 356
\LWR@startref: No longer use zref. 476
\LWR@stoppars: Improved HTML formatting. 344
\LWR@subhtmlelementclass: Improved HTML formatting. . . 334
\LWR@subhyperrefclass: Improved HTML formatting. 480
\LWR@subinlineimage: Improved HTML formatting. 481
\LWR@writeconf: Added ImagesDirectory and ImagesName. 255
\LinkHome: Fix: Document cross-references. 326
\UseMinipageWidths: Added, ... 554
\fbox: Fix: Removed extra space. . . 560
\hyperindexref: Fix: Long index entries. 501
\minipagefullwidth: Made \global. 554
\rotatebox: Improved HTML formatting. 765
\rule: Improved HTML formatting. . 582
\scalebox: Improved HTML formatting. 765
\textgreater: Made robust. 323
\textless: Made robust. 323
General: 2019/02/08 1
LWR@currentautosec: Fix for LOF, LOTfloat in home page. 370
lwarp.css: Added niceframe. . . . 256
lwarp.css: Improved CSS for definition lists. 256
lwarp_formal.css: Improved CSS for table notes. 290
lwarp_one_limage.txt: Image directory and prefix. 296
acronym: Fix for acronym in caption. 602
acronym: No longer uses zref. . . 602
ar: Added. 617
caption: Fix for options clash. . . 234
ed: Added. 699
extramarks: Updated to v3.10. . . 711
fancyhdr: Updated to v3.10. 717
fancyvrb: Improved HTML formatting. 714
graphics: Improved HTML formatting. 762
kotexutf: Patch for references. . . 590
memoir: Docs re: version numbers. 163
multicolrule: Updated for v1.2. . . 848
nameauth: Added. 856
register: Verified for v1.9. 919
subcaption: Added. 961
tocbasic: Updated to v3.26a. . . . 997
truncate: Added. 1013
zref: No longer used. 228
lwarpmk: Added ImagesDirectory and ImagesName. 299
lwarpmk: Fix for **cleanlimages**. . 299
Added early checks for colortab, epsf, hyper, picinpar, picins, sistyle, ucs. 198
Added option ImagesDirectory. 217
Added option ImagesName. 217
Added support for indentfirst. . . 345
Docs: Updated Converting an existing document. 96
Fix: Minipages inside multicols. 848
Improved HTML formatting. 720
Package dates added where possible. 594
Sanitize filenames. 220
tabular: Fix: Minipages inside tabular. 469
lateximage: Added \BaseJobname for multiple projects. 534
Improved HTML formatting. 535
minipage: Honor \LWR@forceminipagefullwidth. 556
v0.67
\FilenameNullify: Added. . . . 571, 576
\FilenameSimplify: Added. . . 356, 363
\LWR@doequation: xfakebold: Added support. 522
\LWR@doubledollar: xfakebold: Added support. 516
\LWR@filenamenoblanks: Improved file name generation. 357
\LWR@lookforpackagename: easyReview: Supported. 231
\LWR@nullfonts: Add'l symbols. . . 568
\LWR@simplifycustom: Added. 356
\LWR@subsingledollar: xfakebold: Added support. 514

\LWR@subsingledollarsvg:
xfakebold: Added support. 512
General: 2019/02/23 1
academicons: Added. 597
bbding: Added. 631
changes: Added. 655
color: Fix for version number. . . 686
dingbat: Added. 695
eurosym: Added. 710
fitbox: Added. 726
fontawesome5: Added. 741
fontawesome: Added. 740
foreign: Added. 745
gloss: Added. 751
karnaugh-map: Added. 785
marvosym: Added. 820
multicap: Added. 847
nomencl: Added. 870
notes: Added. 871
pifont: Added. 908
struktex: Added. 960
textcomp: Nullify in filenames. . 975
typicons: Added. 1014
umoline: Added. 1016
xfakebold: Added support. 506
xfakebold: Added. 1043
xunicode: Nullify in filenames. . 1051
AMS environments: Added xfakebold support. 609
eqnarray: xfakebold: Added support. 527, 528
v0.68
\LWR@@footnotetext: Factored for multiple foot boxes. 350
\LWR@checkloadfilename: Prevented alg, algorithmic, fncylab, pdfcprot. 229
\LWR@printpendingfootnotes: Factored for multiple footnote boxes. 353
\LWR@tabular@warpprintonly: Added. 465
General: 2019/03/05 1
bigfoot: Added. 638
fnpara: Added. 739
footnotebackref: Added. 744
hyperref: Fix for options with braces. 772
layouts: Added. 794
listings: Fix for listings v1.7. . . . 805
longtable: Improved error handling. 808
manyfoot: Added. 817
niceframe: Added. 870
perpage: Added. 901
showtags: Added. 938
tablefootnote: Added. 969
threeparttablex: Added. 983
threeparttable: Fix for caption type. 982
lwarpmk: Improved error handling if incomplete compile. 299
Prevented alg, algorithmic, fncylab, pdfcprot. 198
tabular: Fix: \warpprintonly inside tabular. 467
v0.69
\LWR@maybetocdata: Added support for tocdata. 496
\framebox: Fix: Handle paren arg. . 560
\hypertoc: Added support for tocdata. 496
\hypertocfloat: Added support for tocdata. 498
\makebox: Fix: Handle paren arg. . . 559
\multicolumnrow: multirow: Error if \multirow without \mrowcell. . 852
General: 2019/03/21 1
array: Fix for \tabularnewline. 619, 809
ctable: Added. 689
eqlist: Added. 707
eqparbox: Added. 707
ftcap: Added. 749
graphics: Warning if using scale option. 757
keyfloat: Updated for v2.00. 788
listliketab: Added. 805
minitoc: Added. 843
multirow: Error if \multirow without \mrowcell. 851
rotfloat: Requires graphicx. 923
supertabular: Fix: Clear caption after use. 968
tabularx: Require array. 970
tabulary: Require array. 970
tocdata: Added. 999
topcapt: Added. 1010
xtab: Fix: Clear caption after use. 1050
tabular: Error if \multirow without \mrowcell. 466, 468, 470
lateximage: Fix for *pdftotext* errors from font size change. 537
fminipage: Honors \minipagefullwidth. 562
v0.70
\LWR@closeparagraph: Reduced underfull \hbox warnings. 341
\LWR@lookforpackagename: changes: Updated to v3.1.2. 231

\LWR@mathjaxfilename: Added. . . . 346
\LWR@restoreorigformatting: Fix: \& in a lateximage. 504
\MathJaxFilename: Added. 346
\enddocument: If labels changed, require recompile before making limages. 385
\framebox: Fix: Accept long arg. . . 560
\makebox: Fix: Accept long arg. . . . 559
Fix: Ignore width of 0pt. 559
Fix: No width given. 559
General: 2019/04/03 1
autonum: Added. 627
caption: Fix: Extra par tags. . . . 651
changelayout: Added. 653
changes: Updated to v3.1.2. . . . 655
inputtrc: Added. 784
mathtools: Added. 821
metalogox: Added. 838
metalogo: Used in print mode. . 837
textcomp: Fix for \textinterrobang. 973
textpos: Added optional arg to textblock. 977
xunicode: Fix for \textinterrobang. 1051
AMS environments: Refactored. . 609
Ensure vector font. 209
File: lwarp_mathjax.txt: Loads autoload-all.js extension. . . . 297
File: lwarp_mathjax.txt: Updated to MathJax v2.7.5. 297
Logos: Improved for metalogox, lateximages. 584
LWR@nestspan: Improved minipage, \parbox inside a span. 332
v0.71
\@mpfootnotetext: Improved HTML formatting. 352
Reduced underfull \hbox warnings. 352
\LWR@closeparagraph: Flush left captions. 341
\LWR@closetabledatacell: Fix: Tabular par tags. 416
\LWR@stoppars: Reduced underfull \hbox warnings. 344
\expandcolspec: Added support for tabular '*' column. 427
General: 2019/04/29 1
chemfig: Updated for v1.4. 658
endfloat: Updated for v2.7. . . . 701
lwarp-common-multimedia: Added. 1083
media9: Added. 835
movie15: Added. 846
multimedia: Added. 849
textpos: Updated for v1.9.1. . . . 977
lwarpmk: If wrong lwarpmk.conf version, or wrong OS, displays the print command to recompile. . . 299
Docs: Error testing. 184
Fix: blockquotation tag. 396
Reduced underfull \hbox warnings. 652
v0.72
\LWR@newautopagelabel: Fix: References for xr, xr-hyper. . . . 360
\LWR@printmccoldata: Fix: \multicolumn parameters. . . . 449
\LWR@restoreorigformatting: Use \LWR@formatted for \bfseries, etc. 504
\chapter: Added support for hypbmsec. 377
\ebweight: nfssext-cfr: Added. . . . 572
\hypertoc: Fix: References for xr, xr-hyper. 496
\hypertocfloat: Fix: References for xr, xr-hyper. 497
\lgweight: nfssext-cfr: Added. . . . 572
\paragraph: Added support for hypbmsec. 377
\part: Added support for hypbmsec. 376
\section: Added support for hypbmsec. 377
\subparagraph: Added support for hypbmsec. 377
\subsection: Added support for hypbmsec. 377
\subsubsection: Added support for hypbmsec. 377
\texteb: nfssext-cfr: Added. 564
\textlg: nfssext-cfr: Added. 565
\textulc: fontaxes: Added. 566
\ulcshape: fontaxes: Added. 573
General: 2019/06/08 1
lwarp.css: Added backnaur. . . . 256
lwarp.css: Removed unneeded support for \sishape, \textsi. . 256
backnaur: Added. 630
boxedminipage2e: Added support for lateximages. 644
changes: Fix references for xr, xr-hyper. 655
fontaxes: Added. 571, 742
gloss: Fix references for xr, xr-hyper. 751
hypbmsec: Added. 771
minibox: Added. 842
nfssext-cfr: Added. 861

nomencl: Fix references for xr, xr-hyper. 870
pdfcrypt: Added. 896
shapepar: Added. 937
slantsc: Added. 951
soulutf8: Fix: Loads soul. 954
tabfigures: Added. 969
xr-hyper: Added. 1049
xr: Added. 1048
zhlineskip: Updated to v1.0e. . . 1053
Use \LWR@formatted for \bfseries, etc. 246, 572
v0.73
\@include: Fix: \newpage instead of \clearpage. 235
\HTMLTitle: Added default title if none specified. 347
\LWR@LwarpEnd: Fix: Empty header/footer. 385
\LWR@LwarpStart: Fix: Empty header/footer. 383
\LWR@addbaselinemarker: Improved warning messages. 507
\LWR@descitem: Fix: HTML tags. . . . 406
\LWR@forceemptyline: Added. 212
\LWR@gsavebox: Added global save boxes. 212
\LWR@htmlelementclass: Vertical space. 335
\LWR@htmlelementclassline: Vertical space. 335
\LWR@indentHTMLtwo: Added. 331
\LWR@indexitem: Fix: Avoid empty 500
\LWR@indexsubitem: Fix: Avoid empty 500
\LWR@indexsubsubitem: Fix: Avoid empty 500
\LWR@newhtmlfile: Fix: Empty header/footer. 364, 366
\LWR@nullfonts: Fix: \hspace in sectioning file name. 570
\LWR@titlingmaketitle: titling: Fix: Paragraph tags. 393, 996
Fix: Paragraph tags. 394
\attribution: Fix: Paragraph tags. . 396
\color: Added HTML support. . . . 1036
\fboxBlock: Fix: Paragraph tags. . . 561
\hspace: Fix: Avoid empty . . 580
\l@book: Added \book for memoir. . 498
\maketitle: titling: Fix: Paragraph tags. 995
Fix: Paragraph tags. 392
\marginparBlock: Fix: Paragraph tags. 354
\postbookname: Added \book for memoir. 376
\rule: Fix: Avoid empty . . . 583
General: 2019/07/11 1
lwarp.css: Added \book for memoir. 256
lwarp.css: Improved pkgtocdata formatting. 256
lwarp_formal.css: Added \book for memoir. 285, 290
boxedminipage2e: Fix: Paragraph tags. 644
caption: Fix for starred captions. 650
epigraph: Fix: Paragraph tags. . . 704
fancyvrb: Fix: Nested /<pre>. 721, 724
fancyvrb: Btrivlist: Fixed paragraph tags. 715
intopdf: Updated to v0.2.1. 785
listings: Fix: Paragraph tags. 804
lwarp-common-multimedia: Fix: No size for audio file. . . . 1084, 1085
lwarp-common-multimedia: Fix: Paragraph tags. 1086
lwarp-patch-komascript: Fix for captions. 1056
lwarp-patch-memoir: Added \book. 1062
lwarp-patch-memoir: Fix for \frontmatter* and \mainmatter*. 1063
lyluatex: Added. 814
musicography: Updated to 2019/05/28. Now supports lateximages. 854
quotchap: Fix: Paragraph tags. . . 916
quotchap: Updated to v1.2. 915
quoting: Fix: Paragraph tags. . . . 916
scrextend: Fix: Paragraph tags. . . 927
stackengine: Added. 957
threeparttable: Added measuredfigure. 982
tocdata: Honors \tocdataformat. 999
tocdata: Improved formatting. . . 999
tocdata: Updated to v2.03. 999
versonotes: Updated to v0.4. . . 1022
vwcol: Fix: Paragraph tags. 1025
xy: Fix for \xybox. 1052
xy: Improved xy, reverted \xymatrix, for qcicuit. 1052
Added \book for memoir. . . 317, 329
AMS environments: Fix: alt tags. 609
AMS environments: Fix: Paragraph tags. 609, 610
Fix: Paragraph tags. 389

Numbered HTML entity used for text dollar. 507
tabular: Fix and warning for tabular inside a 466
LWR@blocktextcurrentfont: Fix: Paragraph tags. 572
LWR@nestspan: Fix: quote, quotation inside a span. 332
v0.74
\@ensuredmath: Add \ThisAltText. 517
\AltTextClose: Added. 505
\AltTextOpen: Added. 505
\ImageAltText: Added. 505
\LWR@ThisAltText: Add \ThisAltText. 505
\LWR@addlinktitle: Added. 476
\LWR@doequation: Add \ThisAltText. 522
\LWR@doubledollar: Add \ThisAltText. 516
\LWR@lateximage@oneimage: Factored from lateximage. 533
\LWR@lateximage@oneimageb: Factored from lateximage. 532
\LWR@setcurrentfont: Factored. . . 506
\LWR@singledollar: Add \ThisAltText. 516
\LWR@singledollarmeasure: Fix: Font control. 509
\LWR@subinlineimage: Add \ThisAltText. 481
\LWR@subsingledollar: Add \ThisAltText. 514
\LWR@subsingledollarsvg: Adds star argument for lateximage. 511
\LateximageFontScale: Adjusted SVG math font scaling default to 1. . . 530
\MathImageAltText: Renamed from \mathimage. 505
\PackageDiagramAltText: Renamed from \packagediagramname. . . . 506
\ThisAltText: Add \ThisAltText. . 505
\hspace: Ignore negative space. . . . 579
General: 2019/09/02 1
lwarp.css: Added lyluatex. 256
forest: alt text. 746
geometry: Remembers user's geometry. 751
graphics: Add \ThisAltText. 757, 762, 763
lyluatex: Adapts to user's geometry. 815
lyluatex: Preserves left margin. . . 815
lyluatex: Renamef \lyluateximagename. 814
lyluatex: Split system images, assign class. 814
mhchem: Modified for new lateximage. 839
pdfpages: Adjust to user's paper size. 898
stackengine: alt tags. 957
struktex: alt text. 960
tikz: Added alt text. 985
lwarpmk: **lwarpmk clean** removes add'l files. 299
lwarpmk: **lwarpmk epstopdf** and **pdftosvg** honor directories. . . . 299
Add \ThisAltText. 611
Remembers user's geometry. . . . 224
lateximage: Add \ThisAltText. . . 538
Added second starred argument. 534, 538
Improved alt text. 535
New syntax for \LWR@subinlineimage. 538
eqnarray: Add \ThisAltText. 528
LWR@displaymathother: Uses \MathImageAltText. 518
LWR@equationother: Uses \MathImageAltText. 518
v0.75
\normalcolor: Added for HTML. . . 1034
General: 2019/09/23 1
lwarp.css: Improved marginblock. 256
keyfloat: Fix: \normalcolor. 789
wrapfig: Fix for \linewidth. . . . 1030
wrapfig: Fix for width. 1029
minipage: Fix: \linewidth. 556
v0.76
General: 2019/10/08 1
lwarp.css: Fix for small caps. . . 256
acro: Updated for v2.10. 599
xr-hyper: Updated for v6.1. . . . 1049
xr: Updated for v5.05 and xr-hyper v6.1. 1048
Docs expanded: Multiple projects. 93
File: lwarp_mathjax.txt: Updated to MathJax v2.7.6. 297
v0.77
General: 2019/10/15 1
booktabs: Updated to v1.6180339. 642
chemformula: Updated to v4.15. 660
v0.78
\LWR@linkcatcodes: babel-french: Fix: Hyperlinks. 479
Factored. 479
\LWR@linkmediacatcodes: babel-french: Fix: Hyperlinks. . . 479

Factored. 479
\LWR@nullifyNoAutoSpacing: babel-french: Fix: Hyperlinks. . . 465
\LWR@subhyperrefclass: Remove extra space. 480
\normalfont: Uses \LWR@formatted. 574
General: 2019/11/07 1
accessibility: Added. 598
babel-french: Fix: Hyperlinks. . . 330
caption: Added warning regarding passing options. 649
filecontents: Fix to overwrite existing files using new filecontents environment. . . . 226
geometry: Cleaner option handling. 751
graphics: Fix: alt tag expansion. 762
lwarp-common-multimedia: Fix links with new LaTeX kernel. . . 1084
titlesec: Fix for \titleclass. . . . 991
v0.79
\CustomizeMathJax: Fix: Sanitize for HTML. 362
\LWR@HTMLsanitizeexpand: Fix: \&. 361
\LWR@ProvidesPackagePass: Fix: catoptions. 233
\LWR@checkloadfilename: Prevented bxcjkjatype, hangul. 229
\LWR@closetabledatacell: Fix: Nested tabulars. 417
\LWR@customizeMathJax: MathJax: Hide definitions. 363
\LWR@earlyloadnever: Factored. . . 198
\LWR@forcenewautoidanchor: Factored. 486
\LWR@mathjaxwarn: Warn if using packages not supported by MathJax. 592
\LWR@parseaftercolumn: Remove outermost braces. 423
\LWR@parseatcolumn: Remove outermost braces. 422
\LWR@parsebangcolumn: Remove outermost braces. 422
\LWR@parsebeforecolumn: Remove outermost braces. 423
\LWR@setexparray: Fix: Nested tabulars. 320
\LWR@singledollarmeasure: Factored. 508
\LWR@subHTMLsanitize: Fix: \&. Factored. 360
\LWR@subsingledollarsvg: Adjust for unknown size. 511
Factored. 510
\LWR@tabularendofline: Fix: Nested tabulars. 420
\captionlistentry: Fix: Duplicate auto-id. 489
\macrotocsname: Added. 213
\makebox: Fix: Adjust for virtual page size. 559
\multicolumnrow: multirow: Fix: Nested tabulars. 852
\noalign: Fix: Nested tabulars. . . 464
General: 2020/02/01 1
MathJax: Additional macros. . . . 362
lwarp.css: Fix: Nested tabulars. 256
amsmath: Added MathJax emulation. 612
arydshln: Added MathJax emulation. 621
ar: Added MathJax emulation. . . 619
awesomebox: Added. 628
babel and polyglossia: Added info messages. 591
bigdelim: Added MathJax emulation. 638
bigstrut: Added MathJax emulation. 639
bm: Added. 640
booktabs: Added MathJax emulation. 643
booktabs: Fix for memoir with lateximage. 464, 642
braket: Added. 644
epigraph: Updated to v1.5e. 705
floatflt: Improved width control. 731
fontawesome5: Supports font size, color. 741
fontawesome: Refactored with fix for \FAthree. 740
fontawesome: Supports font size, color. 740
geometry: Also save \textwidth, \textheight. 225
graphics: Factored from \LWR@includegraphicsb. . 759, 760
graphics: Fix for negative angles. 758
ifpdf, ifptex: Provided by iftex. . . 195
ltablex: Fix: Require longtable. . . 809
ltxtable: Fix: Required packages. . 811
luatex85: Removed. 195
mathtools: Added MathJax emulation. 822
multirow: Add: MathJax emulation. 853
multirow: Fix: Centered vertical alignment. 851

niceframe: Fix: Adjust for virtual page size. 870
parallel: Added. 888
parcolumns: Added. 890
pdfcolfoot: Added. 894
pdfcolmk: Added. 894
pdfcolparallel: Added. 894
pdfcolparcolumns: Added. 895
pdfcol: Added. 893
physics: Added. 903
siunitx: Fix: \square, \cubed. . . . 946
siunitx: Improved MathJax. . 943, 944
slashed: Added. 952
steinmetz: Added. 959
subcaption: Fix with memoir. . . 1058
svg: Added. 968
transparent: Supports lateximages. 1011
unicode-math: Added. 1017
widetable: Added. 1027
witharrows: Added. 1028
xcolor: Fix: Nested tabulars. 1040, 1041
xltabular: Fix: Require ltablex. . 1046
xurl: Updated to v0.08. 1052
AMS environments: Fix: Nested. 610
Factored to \LWR@setvirtualpage. 735, 762, 793
Fix: Use newfloat instead of float. 736
Fix: Use full \linewidth. 735
Remember HTML font size. . . . 576
tabular: colortbl: Fix: Nested tabulars. 467
Fix: Nested tabulars. 470
warpMathJax: Added. 222
lateximage: Improved \linewidth. 536
minipage: Fix: \linewidth frame padding. 556
Fix: Adjust for virtual page size. . 555
fminipage: Fix: Adjust for virtual page size. 562
LWR@setvirtualpage: Factored. . . . 554

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

\$ *507*
\& *322*, 7546
\(. 10158
*-images.txt (file) *533*
*_html.aux (file) *360*, *471*, *473*, *533*
*_html.lof (file) *487*
*_html.lot (file) *487*
*_html.tex (file) *249*
\, ... *119*
--shell-escape (option) *98*
\@@@setcpageref 10946
\@@@setcref 10909
\@@@setcrefrange 10923
\@author *387*
\@begintheorem 7179
\@biblabel 9784
\@captype 9371
\@chapcntformat 6394
\@currentHref 9195
\@currentlabelname 9030
\@date *387*
\@dlbfloat 9338
\@donoparitem 7205
\@endtheorem 7191
\@ensuredmath 10168
\@float 9338
\@fnsymbol 6965
\@footnotetext 5906
\@include 1274
\@item 7218
\@makecaption 9394
\@makefnmark 5873
\@makefntext 5872
\@maketitle 56, 6988
\@mklab 7199
\@mpfootnotetext 5908
\@nbitem 7285
\@opargbegintheorem 7185
\@partcntformat 6395
\@partnameformat 6396
\@rowc@lors 7917
\@rowcolors 7916
\@seccntformat 6392
\@starttoc 9510
\@textsubscript 11717
\@textsuperscript 11713
\@title *387*
\@wrglossary 9724
\@wrindex 9712
\[. 10158
\\ *577*
$ *515*
$$ *515*
2in1 (package) *594*
2up (package) *594*

A

a4 (package) *594*
a4wide (package) *595*
a5comb (package) *595*
abstract (environment) 7053
abstract (package) *130*, *595*
\abstractname *109*, 7052
academicons (package) *597*
accessibility (package) *598*
accsupp (package) *599*
acro (package) *599*
acronym (package) *601*
\addcontentsline 9459
addlines (package) *603*
\AddSubtitlePublished 7018
adjmulticol (package) *602*
Adobe (program) *72*
\affiliation 6868
afterpage (package) *598*, *603*
algorithm2e (package) *604*
algorithmicx (package) *161*, *607*
align (environment) 75
align* (environment) 78
alignat (environment) 87
alignat* (environment) 90
alltt (package) *608*

\AltTextClose 111, 9856
\AltTextOpen 111, 9855
\AmS 12048
amsmath (package) 609
amsthm (package) 612
\and 387
anonchap (package) 616
anysize (package) 616
appendix (package) 130, 617
ar (package) 617
arabicfront (package) 619
array (package) 619
\arrayrulecolor 7924
\arrayrulecolornexttoken 7924
arydshln (package) 620
AsciiDoc (program) 72
AsciiDoctor (program) 72
Asciidoctor-LaTeX (program) 72
asymptote (package) 155, 622
atbegshi (package) 623
attachfile (package) 623
attachfile2 (package) 624
\attrib 164, 397, 1021
\attribution 7063
authblk (package) 130, 626
\author 116, 387
autonum (package) 627
autosec 369
awesomebox (package) 628
axessibility (package) 629
axodraw2 (package) 630

B

babel (package) 165
\backmatter 6370
backnaur (package) 630
backref (package) 630
balance (package) 631
\BaseJobname 5183
BaseJobname (option) 105, 217
bbding (package) 631
\bfseries 11675
biblatex (package) 636
\bibliography 9775
\BibTeX 12041
bibunits (package) 636
bigdelim (package) 161, 637
bigfoot (package) 638
bigstrut (package) 639
bitpattern (package) 639
BlockClass (environment) 5571
\BlockClassSingle 5584
blowup (package) 640
bm (package) 640
booklet (package) 640
bookmark (package) 641
booktabs (package) 641
boolean:
CombineHigherDepths 108, 356
FileSectionNames 109, 324
FormatEPUB 172, 242
FormatWP 174, 242
HTMLDebugComments 109, 237
LWR@allowanothergeometry 224
LWR@amsmultline 525
LWR@copiedsidetoc 492
LWR@doingapar 339
LWR@doingcmidrule 412
LWR@doingstartpars 339
LWR@doingtbrule 412
LWR@dynamicmath 322
LWR@emptyatbang 412
LWR@exitingtabular 413
LWR@forceminipagefullwidth 554
LWR@foundmrowcell 412
LWR@freezethisautoid 486
LWR@indisplaymathimage 506
LWR@intabularmetadata 413
LWR@isstartingequation 531
LWR@mathmacro 321
LWR@minipagefullwidth 554
LWR@minipagethispar 555
LWR@opttablecol 413
LWR@origmathjax 216
LWR@skipatbang 412
LWR@skippingmcolrowcell 412
LWR@skippingmrowcell 412
LWR@starredlongtable 415
LWR@startedrow 411
LWR@tableparcell 412
LWR@tabularcelladded 411
LWR@tabularmutemods 413
LWR@tracinglwarp 237
LWR@unknownmathsize 508
LWR@usedmultirow 412
LWR@validtablecol 413
LWR@verbtags 398
LWR@xfakebold 506
mathjax 216
usingOSWindows 215
warpingHTML 216
warpingprint 216
WPMarkFloats 175, 243
WPMarkLOFT 176, 243
WPMarkMath 176, 244
WPMarkMinipages 176, 243
WPMarkTOC 176, 243
WPTitleHeading 177, 244
bophook (package) 643
bounddvi (package) 644
boxedminipage2e (package) 644

\boxframe 194
braket (package) 644
breakurl (package) 645
breqn (package) 645
bsheaders (package) 647
BVerbatim (environment) 247
bxpapersize (package) 647
bytefield (package) 647

C

calc (package) 226
cancel (package) 648
canoniclayout (package) 649
capt-of (package) 490
caption (package) 161, 234, 490, 649
\caption@begin 9438
\caption@end 9438
\captionlistentry 9442
\captionof 9475
\CaptionSeparator 9393
cases (package) 652
center (environment) 10698
\centering 10722
\centerline 10741
changebar (package) 653
changelayout (package) 653
changepage (package) 654
changes (package) 655
chappg (package) 657
\chapter 6597
chapterbib (package) 657
chemfig (package) 657
chemformula (package) 659
chemgreek (package) 664
chemmacros (package) 665
chemnum (package) 683
chkfloat (package) 684
chngpage (package) 684
cite (package) 684
CJK (package) 685
CJKutf8 (package) 685
class (key) [Gin] 758
class:
  internet 71
  komascript 163
  memoir 163, 164, 1021
\cleardoublepage 11880
\clearpage 11880
cleveref (package) 126, 549
clrdblpg (package) 685
cm-super (package) 99
cmap (package) 100
cmdtrack (package) 685
\color 51
color (package) 153, 686
\colorbox 77
\colorboxBlock 90, 10806
colortbl (package) 160, 686
CombineHigherDepths (boolean) . . 108, 356
comment (package) 222
\ConTeXt 12036
continue (package) 688
copyrightbox (package) 688
counter:
  FileDepth 108, 355
  FootnoteDepth 109, 349
  footnoteReset 349
  lofdepth 496
  lotdepth 497
  LWR@cellcolordepth 414
  LWR@currentautosec 370
  LWR@externalfilecnt 506
  LWR@hdashedlines 412
  LWR@hlines 412
  LWR@htmlfilenumber 324
  LWR@latestautopage 488
  LWR@lateximagedepth 530
  LWR@lateximagenumber 530
  LWR@LIpage 530
  LWR@midrulecounter 415
  LWR@nextautoid 487
  LWR@nextautopage 487
  LWR@nextequation 519
  LWR@prevFileDepth 370
  LWR@previousautopagelabel 359
  LWR@spandepth 339
  LWR@splitstarcopies 427
  LWR@splitstarindex 427
  LWR@starcount 427
  LWR@startingequation 531
  LWR@tablecolspecindex 414
  LWR@tablecolspecwidth 414
  LWR@tableLaTeXcolindex 414
  LWR@tabletotalLaTeXcols 414
  LWR@tabletotalLaTeXcolsnext . . . 414
  LWR@tabularDepth 413
  LWR@tabularpardepth 413
  LWR@thisautoid 485
  LWR@thisautoidWP 486
  LWR@virtualpagedepth 553
  SideTOCDepth 108, 494
  tocdepth 108
\cpagerefFor 10945
crop (package) 689
\CSSFilename 109, 113, 5815
ctable (package) 689
\CustomizeMathJax 6184, 6235
cuted (package) 692
cutwin (package) 692

D

\date 116

dblfloatfix (package) 693
dblfnote (package) 693
dcolumn (package) 693
\DeclareGraphicsExtensions 2
\defaddtocounter 686
dejavu (package) 99
description (environment) 7366
diagbox (package) 694
dingbat (package) 695
\displaymathnormal 148, 523, 10356
\displaymathother 149, 524, 10367
ditaa (package) 171
\dotfill 11798
\doublerulesepcolor 7926
\doublerulesepcolornexttoken 7926
dprogress (package) 696
draftcopy (package) 696
draftfigure (package) 697
draftwatermark (package) 697
dvipdfm (option) 101, 219
dvipdfmx (option) 101, 219
dvips (option) 101, 219

E

easy-todo (package) 697
ebook (package) 698
\ebweight 11677
ed (package) 699
ellipsis (package) 699
embrac (package) 700
\emph 11324
emptypage (package) 700
\end@dlbfloat 9355
\end@float 9355
\enddocument 6833
endfloat (package) 701
endheads (package) 701
endnotes (package) 130, 702
\enlargethispage 11879
\enskip 578, 11824
enumerate (environment) 7341
enumerate (package) 703
enumitem (package) 703
environ (package) 228
environment:
  fminipage 122
  lateximage 529
  picture 552
  titlepage 116
  titlingpage 116
  warpall 115
  warpHTML 112, 114
  warpMathJax 115
  warpprint 111, 115
environments:
  abstract 7053
  align 75
  align* 78
  alignat 87
  alignat* 90
  BlockClass 5571
  BVerbatim 247
  center 10698
  description 7366
  enumerate 7341
  eqnarray 10416
  equation 10330
  equation* 10340
  fcolorminipage 170, 10847
  flalign 81
  flalign* 84
  flushleft 10714
  flushright 10706
  fminipage 11254
  gather 69
  gather* 72
  itemize 7335
  lateximage 10559, 10693
  list 7305
  longtable 4
  LWR@BlockClassWP 5610
  LWR@blocktextcurrentfont 11664, 11745
  LWR@creatinglwarpmk 937
  LWR@displaymathnormal 10198
  LWR@displaymathother 10203
  LWR@equationother 10217
  LWR@figcaption 9408
  LWR@glrbox 725
  LWR@nestspan 5418
  LWR@setvirtualpage 11014
  math 10197
  minipage 11040
  multline 62
  multline* 65
  picture 11005
  quotation 7087
  quote 7081
  tabbing 7164
  tabular 8865
  thebibliography 9785
  theindex 9685
  titlepage 6885
  titlingpage 14
  verbatim 7154
  verse 2, 7095
  warpall 921
  warpHTML 922
  warpMathJax 928
  warpprint 925
epigraph (package) 704
epsfig (package) 705

epstopdf (package) 154, 706
epstopdf (program) 152, 544
epstopdf-base (package) 706
eqlist (package) 707
eqnarray (environment) 10416
eqparbox (package) 707
equation (environment) 10330
equation* (environment) 10340
errata (package) 708
eso-pic (package) 710
\etalchar 131
etoolbox (package) 195
eurosym (package) 710
everyhook (package) 226
everypage (package) 711
everyshi (package) 711
\expandcolspec 7802
expl3 (package) 226
extramarks (package) 711

F

fancybox (package) 123, 712
fancyhdr (package) 717
fancyref (package) 718
fancytabs (package) 719
fancyvrb (package) 719
\fbox 122, 11225
\fboxBlock 122, 11234
\fcolorbox 107, 10808
\fcolorboxBlock 130, 10845
fcolorminipage (environment) . 170, 10847
figcaps (package) 725
figsize (package) 726
file:
*-images.txt 533
*_html.aux 360, 471, 473, 533
*_html.lof 487
*_html.lot 487
*_html.tex 249
glyphtounicode.tex 99
lwarp.css 113, 256
lwarp.ist 142, 294
lwarp.xdy 143, 295
lwarp_baseline_marker.eps 507
lwarp_baseline_marker.png 507
lwarp_formal.css 290
lwarp_mathjax.txt 296
lwarp_one_limage.cmd 296
lwarp_sagebrush.css 285
lwarp_tutorial.txt 81
lwarpmk.conf 255
lwarpmk.lua 182
project.css 113
project.lwarpmkconf 256
sample_project.css 113, 294
tutorial.tex 81
filecontents (package) 226
FileDepth (counter) 108, 355
\FilenameLimit 109, 6018
\FilenameNullify 125, 11635, 11746
\FilenameSimplify 125, 6039, 6236
FileSectionNames (boolean) 109, 324
fitbox (package) 726
fix2col (package) 727
fixme (package) 166, 727
fixmetodonotes (package) 728
flafter (package) 729
\flagverse 1021
flalign (environment) 81
flalign* (environment) 84
Flare (program) 72
flippdf (package) 729
float (package) 161, 729
floatflt (package) 731
floatpag (package) 732
floatrow (package) 162, 732
fltrace (package) 737
\flushbottom 5324
flushend (package) 737
flushleft (environment) 10714
flushright (environment) 10706
fminipage (environment) 122, 11254
fnbreak (package) 738
fncychap (package) 738
fnlineno (package) 739
fnpara (package) 739
fnpos (package) 739
fontawesome (package) 740
fontawesome5 (package) 741
fontaxes (package) 742
fontenc (package) 99, 742
fontspec (package) 99
footmisc (package) 743
footnote (package) 744
footnotebackref (package) 744
FootnoteDepth (counter) 109, 349
footnotehyper (package) 745
footnoterange (package) 745
footnoteReset (counter) 349
footnpag (package) 745
\ForceHTMLPage 129, 6345
\ForceHTMLTOC 129, 6351
foreign (package) 745
forest (package) 746
FormatEPUB (boolean) 172, 242
FormatWP (boolean) 174, 242
\framebox 11203
framed (package) 746
FrameMaker (program) 72
\frontmatter 6367
ftcap (package) 749

ftnright (package) 749
fullminipage (package) 749
fullpage (package) 749
fullwidth (package) 750
\fup 11721
\fussy 5326
fwlw (package) 750

G

gather (environment) 69
gather* (environment) 72
GELLMU (program) 71
gentombow (package) 750
geometry (package) 223, 751
gettitlestring (package) 226
[Gin]:
  class (key) 758
GladTeX (program) 71
gloss (package) 131, 751
glossaries (package) 131, 752
GlossaryCmd (option) . . . 104, 131, 219, 752
glyphtounicode.tex (file) 99
gmeometric (package) 754
graphics (package) 151, 754
graphicx (package) 151, 767
grffile (package) 153, 767
grid (package) 767
grid-system (package) 768
gridset (package) 768

H

hang (package) 768
hanging (package) 770
Hevea (program) 71
\hfill 11794
\HomeHTMLFilename 5185
HomeHTMLFilename (option) . 101, 106, 218
\href 9243
\hrulefill 11796
\hskip 119
\hspace 119, 578, 11832, 11863, 11869
htlatex (program) 71
\HTMLAuthor 110, 117, 5852
HTMLDebugComments (boolean) 109, 237
\HTMLDescription 110, 117, 5857
\HTMLentity 5167
\HTMLFilename 5184
HTMLFilename (option) 101, 106, 218
\HTMLFirstPageTop 110, 5803
htmlglossary (option) [lwarpmk] . 131, 752
HTMLIndexCmd (option) 103, 219
\HTMLLanguage 109, 6630
HTMLLatexCmd (option) 101, 168, 218
\HTMLleftmargini (length) 164, 397, 398, 1021
\HTMLPageBottom 110, 387, 5809
\HTMLPageTop 110, 5806
\HTMLTitle 109, 117, 5847
\HTMLTitleAfterSection 110, 6653
\HTMLTitleBeforeSection 110, 6648
\HTMLunicode 5175
\HTMLvleftskip (length) 164, 397, 398, 1021
hypbmsec (package) 771
hypcap (package) 771
hypdestopt (package) 771
\hyperindexref 9753
hypernat (package) 771
hyperref (package) 127, 479, 772
\hypertoc 9619
\hypertocfloat 9642
hyperxmp (package) 777
hyphenat (package) 777

I

idxlayout (package) 778
\if@titlepage 6863
ifoddpage (package) 778
ifplatform (package) 195
\IgnoreMinipageWidths . . . 121, 553, 11030
\ImageAltText 110, 9857
ImagesDirectory (option) 101, 217
ImagesName (option) 101, 217
imakeidx (package) 779
\includegraphics 316
indentfirst (package) 345
InDesign (program) 72
index (package) 783
\InlineClass 5594
\inlinemathnormal 148, 5160
\inlinemathother 148, 5156
inputenc (package) 99
inputenx (package) 99
inputtrc (package) 784
internet (class) 71
intopdf (package) 785
itemize (environment) 7335
\itshape 11689

K

karnaugh-map (package) 785
\kern 119
key:
  [Gin]:
    class 758
keyfloat (package) 163, 788
komascript (class) 163
kotex (package) 167
kvoptions (package) 215

L

\l@book 9663
\l@chapter 9665
\l@figure 9676

\l@paragraph 9674
\l@part 9664
\l@section 9671
\l@subparagraph 9675
\l@subsection 9672
\l@subsubsection 9673
\l@table 9677
\LaTeX 11990
LaTeX2HTML (program) *71*
\LaTeXe 11990
lateximage (environment) *529*, 10559, 10693
\LateximageFontScale 10489
\LateximageFontSizeName 10488
latexmk (option) *101*, *106*, *219*
latexmk (program) *169*
LatexmkIndexCmd (option) *104*, *219*
LaTeXML (program) *71*
layaureo (package) *794*
layout (package) *794*
layouts (package) *794*
leading (package) *797*
\leftline 10740
length:
\HTMLleftmargini . *164*, *397*, *398*, *1021*
\HTMLvleftskip . . . *164*, *397*, *398*, *1021*
\LWR@cmidrulewidth *439*
\LWR@heavyrulewidth *439*
\LWR@lightrulewidth *439*
\LWR@minipageheight *553*
\LWR@minipagewidth *553*
\LWR@tempheight *579*
\LWR@tempraise *579*
\LWR@tempwidth *579*
\LWR@thiscmidrulewidth *439*
\VerbatimHTMLWidth *398*
\vleftmargini *164*, *397*, *1021*
\vleftskip *164*, *397*, *1021*
letltxmacro (package) *195*
letterspace (package) *797*
lettrine (package) *798*
\lgweight 11679
LibreOffice (program) *72*
\linebreak 11873
lineno (package) *798*
\LinkHome 5243, 5248
\linkhomename *106*, 5240
Linux (program) *114*, *213*
lips (package) *801*
list (environment) 7305
listings (package) *801*
listliketab (package) *805*
\listof 9566
\listoffigures 9540
\listoftables 9553
lltjext (package) *805*
lmodern (package) *99*
lofdepth (counter) *496*
longtable (environment) 4
longtable (package) *159*, *806*
lotdepth (counter) *497*
lscape (package) *809*
ltablex (package) *809*
ltcaption (package) *810*
ltxgrid (package) *810*
ltxtable (package) *811*
lua-check-hyphen (package) *811*
lua-visual-debug (package) *811*
luacolor (package) *811*
\LuaLaTeX 12019
LuaLaTeX (program) [requirement] *75*
\LuaTeX 12019
luatodonotes (package) *166*, *812*
lwarp (package) *101*
lwarp-common-multimedia (package) . *1082*
lwarp-patch-komascript (package) . *1055*
lwarp-patch-memoir (package) *1057*
lwarp.css (file) *113*, *256*
lwarp.ist (file) *142*, *294*
lwarp.xdy (file) *143*, *295*
lwarp_baseline_marker.eps (file) *507*
lwarp_baseline_marker.png (file) *507*
lwarp_formal.css (file) *290*
lwarp_mathjax.txt (file) *296*
lwarp_one_limage.cmd (file) *296*
lwarp_sagebrush.css (file) *285*
lwarp_tutorial.txt (file) *81*
[lwarpmk]:
htmlglossary (option) *131*, *752*
printglossary (option) *131*, *752*
lwarpmk (option) *104*, *218*, *298*
lwarpmk (program) *182*, *299*
lwarpmk epstopdf (program) *152*, *544*
lwarpmk pdftosvg (program) *152*, *544*
lwarpmk.conf (file) *255*
lwarpmk.lua (file) *182*
\lwarpsetup 785
\LWR@@footnotetext 5876
\LWR@@makebox@align 11159
\LWR@@makebox@paren 11144
\LWR@@printpendingfootnotes 5942
\LWR@addbaselinemarker 9890
\LWR@addcdashline 8196
\LWR@addcmidruletrim 8162
\LWR@addcmidrulewidth 8193
\LWR@addcompilecmd 1650
\LWR@addformatwpalignment 8212
\LWR@addleftmostbartag 7996
\LWR@addlinktitle 9123
\LWR@addmathjax 10265
\LWR@addmulticolvertrulecolor . . . 8464

\LWR@addrulewidth 8167
\LWR@addtabularcellcolor 8339
\LWR@addtabularhrulecolor 8240
\LWR@addtabularrowcolor 8225
\LWR@addtabularrulecolors 8286
\LWR@afterendverbatim 7137
LWR@allowanothergeometry (boolean) . 224
\LWR@amsmathbody 10522
\LWR@amsmathbodynumbered 10528
LWR@amsmultline (boolean) 525
\LWR@applyxfakebold 9872
\LWR@atbeginverbatim 7118
\LWR@backgroundcolor 39
\LWR@beginhideamsmath 10401
LWR@BlockClassWP (environment) . . . 5610
LWR@blocktextcurrentfont (environment) 11664, 11745
\LWR@botnavigation 5266
LWR@cdashlines 439
LWR@cellcolordepth (counter) 414
\LWR@cellHTMLcolor 7921
\LWR@checkloadbefore 59
\LWR@checkloadfilename 1094
\LWR@checkloadnever 78
\LWR@clearmidrules 8077
\LWR@closeparagraph 5710
\LWR@closeparagraph@br 5700
\LWR@closeprevious 5293
\LWR@closetabledatacell 7499
\LWR@cmidrulewidth (length) 439
LWR@colafterspec 414
LWR@colatspec 414
LWR@colbangspec 414
LWR@colbarspec 414
LWR@colbeforespec 414
\LWR@columnHTMLcolor 7919
\LWR@columnspeclookahead 7630
\LWR@compilecmd 1647
\LWR@compileuplatex 1679
\LWR@convertto 681
LWR@copiedsidetoc (boolean) 492
\LWR@copyfile 1309
\LWR@createautosec 6376
LWR@creatinglwarpmk (environment) . 937
LWR@currentautosec (counter) 370
\LWR@currentcss 5814
\LWR@currenttextcolor 11732
\LWR@customizedMathJax 6183
\LWR@customizeMathJax 6219
\LWR@descitem 7347
LWR@displaymathnormal (environment) 10198
LWR@displaymathother (environment) 10203
\LWR@docdashline 8126
\LWR@docmidrule 8110
\LWR@doequation 10278
\LWR@doindexentry 9733
LWR@doingapar (boolean) 339
LWR@doingcmidrule (boolean) 412
LWR@doingstartpars (boolean) 339
LWR@doingtbrule (boolean) 412
\LWR@domulticolumn 8502
\LWR@doubledollar 10109
LWR@dynamicmath (boolean) 322
\LWR@earlyclassloadnever 89
\LWR@earlyloadnever 84
LWR@emptyatbang (boolean) 412
\LWR@endfloatalignment 9387
\LWR@endhideamsmath 10409
\LWR@endofline 11805
\LWR@ensuredoingapar 5676
LWR@equationother (environment) . 10217
\LWR@equationtag 10508
LWR@exitingtabular (boolean) 413
\LWR@expandableformatted 1371
\LWR@expandableformattedenv 1400
\LWR@expcolspec 7798
LWR@externalfilecnt (counter) 506
\LWR@fboxstyle 11219
LWR@figcaption (environment) 9408
\LWR@filenamenoblanks 6050
\LWR@filestart 6662
\LWR@findcurrenttextcolor 11735
\LWR@findword 1091
\LWR@firstoffour 707
\LWR@floatalignment 9373
\LWR@floatalignmentname 9372
\LWR@floatbegin 9302
\LWR@floatend 9340
\LWR@floatstyle 2
\LWR@footnotebox 5871
\LWR@footnotetext 5905
\LWR@forceemptyline 703
LWR@forceminipagefullwidth (boolean) 554
\LWR@forceminwidth 11208
\LWR@forcenewautoidanchor 9361
\LWR@forcenewpage 5313
\LWR@formatted 1356
\LWR@formattedenv 1386
\LWR@formatting 1355
LWR@foundmrowcell (boolean) 412
\LWR@fourthoffour 707
LWR@freezethisautoid (boolean) 486
\LWR@futurenonspacelet 7425
\LWR@FVstyle 56
\LWR@getexparray 5129
\LWR@getmynexttoken 7435
LWR@glrbox (environment) 725
\LWR@gsavebox 711
LWR@hdashedlines (counter) 412

\LWR@heavyrulewidth (length) 439
\LWR@hidelatexequation 10251
LWR@hlines (counter) 412
\LWR@HTML@caption@begin 9418
\LWR@HTML@caption@end 9429
\LWR@HTML@ref 9166
\LWR@htmlblockcomment 5508
\LWR@htmlblocktag 5510
\LWR@HTMLcline 8846
\LWR@htmlclosecomment 5480
\LWR@htmlcomment 5501
\LWR@htmldivclass 5547
\LWR@htmldivclassend 5550
\LWR@htmlelement 5561
\LWR@htmlelementclass 5533
\LWR@htmlelementclassend 5539
\LWR@htmlelementclassline 5553
\LWR@htmlelementend 5564
LWR@htmlfilenumber (counter) 324
\LWR@htmlfileref 9049
\LWR@HTMLhline 8827
\LWR@HTMLLatexCmd 1689
\LWR@htmlmathlabel 10386
\LWR@htmlmulticolumn 8562
\LWR@htmlopencomment 5480
\LWR@htmlrefsectionfilename 5230
\LWR@HTMLsanitize 6163
\LWR@HTMLsanitizeexpand 6173
\LWR@htmlsectionfilename 5195
\LWR@htmlspan 5457
\LWR@htmlspanclass 5465
\LWR@htmltag 5475
\LWR@htmltagc 5408
\LWR@hyperindexrefnullified 9738
\LWR@indentHTML 5402
\LWR@indentHTMLtwo 5405
\LWR@indexitem 9700
\LWR@indexnameref 9732
\LWR@indexsubitem 9704
\LWR@indexsubsubitem 9708
LWR@indisplaymathimage (boolean) .. 506
\LWR@instertatbangcols 7491
LWR@intabularmetadata (boolean) ... 413
\LWR@isolate 694
LWR@isstartingequation (boolean) .. 531
\LWR@itemizeitem 7328
LWR@latestautopage (counter) 488
\LWR@lateximage@oneimage 10544
\LWR@lateximage@oneimageb 10535
LWR@lateximagedepth (counter) 530
\LWR@lateximagedepthref 9052
LWR@lateximagenumber (counter) 530
\LWR@lateximagenumberref 9055
\LWR@lateximagesfile 1271
\LWR@latexmkcmd 1665
\LWR@latexmkdvipdfm 1670
\LWR@latexmkvar 1659
\LWR@lightrulewidth (length) 439
\LWR@linkcatcodes 9202
\LWR@linkmediacatcodes 9210
LWR@LIpage (counter) 530
\LWR@listitem 7290
\LWR@loadafter 18
\LWR@loadbefore 50
\LWR@loadnever 65
\LWR@longtabledatacaptiontag 8571
\LWR@lookforpackagename 1141
\LWR@lwarpconfversion 1641
\LWR@LwarpEnd 6808, 12067
\LWR@lwarplabel 9058
\LWR@LwarpStart 6742, 12067
\LWR@maketitlesetup 33, 6957
\LWR@mathjaxfilename 5825
\LWR@mathjaxwarn 12203
LWR@mathmacro (boolean) 321
\LWR@maybenewtablerow 7928
\LWR@maybeprintpendingfootnotes . 5961
\LWR@maybetocdata 9618
LWR@midrulecounter (counter) 415
LWR@midrules 438
LWR@minipagefullwidth (boolean) ... 554
\LWR@minipageheight (length) 553
\LWR@minipagestartpars 11812
\LWR@minipagestoppars 11815
LWR@minipagethispar (boolean) 555
\LWR@minipagewidth (length) 553
\LWR@multicolother 8404
\LWR@multicolpartext 8398
\LWR@multicolskip 8410
\LWR@multirowborder 3
\LWR@mynexttoken 7424
\LWR@myshorttoc 9487
\LWR@nameref 9046
LWR@nestspan (environment) 5418
\LWR@new@label 9103
\LWR@newautoidanchor 9366
\LWR@newautopagelabel 6143
\LWR@newhtmlfile 6239
LWR@nextautoid (counter) 487
LWR@nextautopage (counter) 487
LWR@nextequation (counter) 519
\LWR@notltjloadafter 37
\LWR@notmemoirloadafter 34
\LWR@nullfonts 11491
\LWR@nullifycomment 1085
\LWR@nullifyNoAutoSpacing 8852
\LWR@nulllistfills 7297
\LWR@openparagraph 5682
LWR@opttablecol (boolean) 413
\LWR@orig@setBold 9870

\LWR@orig@unsetBold 9871
\LWR@origcolspec 7474
LWR@origmathjax (boolean) *216*
\LWR@overline 11727
\LWR@parseaftercolumn 7707
\LWR@parseatcolumn 7640
\LWR@parsebangcolumn 7669
\LWR@parsebarcolumn 7719
\LWR@parsebeforecolumn 7695
\LWR@parsecoloncolumn 7744
\LWR@parseDcolumn 7793
\LWR@parsedrequirepackagenames . . 1084
\LWR@parsenormalcolumn 7769
\LWR@parsepcolumn 7781
\LWR@parsesemicoloncolumn 7765
\LWR@parsestarcolumn 7790
\LWR@parsetablecols 7830
\LWR@parsewcolumn 7785
\LWR@patcherror 689
\LWR@patchlists 7372
\LWR@pdfencoding 612
\LWR@popclose 5093
\LWR@PreloadedPackage 10745
LWR@prevFileDepth (counter) *370*
LWR@previousautopagelabel (counter) *359*
\LWR@printatbang 7971
\LWR@printbartag 7961
\LWR@printcloselist 7197
\LWR@PrintLatexCmd 1689
\LWR@printlength 1065
\LWR@printmccoldata 8413
\LWR@printmccoltype 8356
\LWR@printopenlist 7198
\LWR@printpendingfootnotes 5958
\LWR@printpendingmpfootnotes 5970
\LWR@printpercentlength 682
\LWR@printthetitle 6909
\LWR@providelength 678
\LWR@ProvidesPackageDrop 1257
\LWR@ProvidesPackageDropA 1242
\LWR@ProvidesPackageDropB 1251
\LWR@ProvidesPackagePass 1212
\LWR@pushclose 5066
\LWR@pushoneclose 6382
\LWR@quickfile 1268
\LWR@ref@ignorestar 9174
\LWR@remembertag 10512
\LWR@requesttoc 6801
\LWR@requirepackagenames 1083
\LWR@restoreorigaccents 1596
\LWR@restoreorigformatting 9808
\LWR@restoreoriglists 7400
\LWR@rowHTMLcolor 7920
\LWR@ruleHTMLcolor 7922
\LWR@sanitize 856
\LWR@sanitized 855
\LWR@secondoffour 707
\LWR@section 6399
\LWR@sectionnumber 6373
\LWR@select@html@hspace 11832
\LWR@select@html@nohspace 11863
\LWR@select@print@hspace 11866
\LWR@setcurrentfont 9873
\LWR@setexparray 5120
\LWR@setlatestname 9032
\LWR@setOSWindows 773
\LWR@setref 9040
LWR@setvirtualpage (environment) 11014
\LWR@shellescapecmd 1642
\LWR@sidetoc 9592
\LWR@simplifycustom 6038
\LWR@simplifyname 6027
\LWR@singledollar 10136
\LWR@singledollarmeasure 9931
LWR@skipatbang (boolean) *412*
LWR@skippingmcolrowcell (boolean) . . *412*
LWR@skippingmrowcell (boolean) *412*
LWR@spandepth (counter) *339*
\LWR@splitstarcontents 7801
LWR@splitstarcopies (counter) *427*
LWR@splitstarindex (counter) *427*
LWR@starcount (counter) *427*
LWR@starredlongtable (boolean) *415*
LWR@startedrow (boolean) *411*
LWR@startingequation (counter) *531*
\LWR@startingequationtag 10507
\LWR@startnewdepth 6386
\LWR@startpars 5752
\LWR@startref 9129
\LWR@stoppars 5770
\LWR@stripperiod 9031
\LWR@strresult 7472
\LWR@subaddcmidruletrim 8149
\LWR@subaddtabularcellcolor 8333
\LWR@subcdashline 8113
\LWR@subcmidrule 8095
\LWR@subhtmlelementclass 5515
\LWR@subHTMLsanitize 6151
\LWR@subhyperref 9218
\LWR@subhyperrefclass 9232
\LWR@subhyperreftext 9227
\LWR@subinlineimage 9281
\LWR@sublabel 9077
\LWR@subnewref 9160
\LWR@subsingledollar 10056
\LWR@subsingledollarsvg 9977
\LWR@subtableofcontents 9500
\LWR@subtabularhtmlcolumns 8642
\LWR@syncmathjax 10231
LWR@tablecolspec *413*

LWR@tablecolspecindex (counter) ... 414
LWR@tablecolspecwidth (counter) ... 414
\LWR@tabledatacolumntag 8731
\LWR@tabledatasinglecolumntag ... 8010
LWR@tableLaTeXcolindex (counter) ... 414
LWR@tableparcell (boolean) 412
LWR@tabletotalLaTeXcols (counter) .. 414
LWR@tabletotalLaTeXcolsnext (counter) 414
\LWR@tabular@warpprintonly 8848
LWR@tabularcelladded (boolean) 411
LWR@tabularDepth (counter) 413
\LWR@tabularendofline 7609
\LWR@tabularfinishrow 7560
\LWR@tabularhtmlcolumns 8652
\LWR@tabularleftedge 8001
LWR@tabularmutemods (boolean) 413
LWR@tabularpardepth (counter) 413
\LWR@tdaddstyle 8136
\LWR@tdendstyles 8142
\LWR@tdstartstyles 8135
\LWR@tempcolor 11733
\LWR@tempcolortwo 11733
\LWR@tempheight (length) 579
\LWR@tempraise (length) 579
\LWR@tempwidth (length) 579
\LWR@textcurrentcolor 22, 11738
\LWR@textcurrentfont 11642, 11744
\LWR@thirdoffour 707
\LWR@ThisAltText 9859
LWR@thisautoid (counter) 485
LWR@thisautoidWP (counter) 486
\LWR@thiscmidrulewidth (length) 439
\LWR@thiscolspec 8009
\LWR@thisfilename 6021
\LWR@thisnewfilename 6026
\LWR@titlingmaketitle 83, 7010
\LWR@topnavigation 5263
\LWR@traceinfo 1337
LWR@tracinglwarp (boolean) 237
LWR@trimlrules 438
LWR@trimrrules 439
\LWR@unknownengine 1654
LWR@unknownmathsize (boolean) 508
LWR@usedmultirow (boolean) 412
LWR@validtablecol (boolean) 413
LWR@verbtags (boolean) 398
LWR@virtualpagedepth (counter) 553
\LWR@vspace 11870
\LWR@WPcell 8208
\LWR@writeconf 1791
\LWR@xcolorrowHTMLcolor 7918
LWR@xfakebold (boolean) 506
\LWRopquote 768
\LWRopseq 769
\LWRPrintStack 5271
\LWRsetnextfloat 9402
lyluatex (package) 814
\LyX 12063

M

Mac OS (program) 114, 213
\macrotocsname 737
Madcap (program) 72
magaz (package) 816
\mainmatter 6363
make (program) 171
\makebox 11168
makeidx (package) 144, 816, 997, 998
\MakeIndex 12041
makeindex (option) 101, 106, 219
makeindex (program) 142
makeindexStyle (option) ... 103, 142, 217
\makelabel 402
\maketitle 47, 116, 6969
manyfoot (package) 817
marginal (package) 819
marginfit (package) 819
marginfix (package) 819
marginnote (package) 820
\marginpar 120, 354, 5982
\marginparBlock 121, 354, 5993, 6011
\markboth 5321
\markright 5322
marvosym (package) 820
math (environment) 10197
\MathImageAltText 111, 9858
MathJax (program) 146, 147
MathJax (program) [requirement] 75
mathjax (boolean) 216
mathjax (option) 101, 106, 217
\MathJaxFilename 109, 5826
mathsvg (option) 101, 106, 216
mathtools (package) 149, 821
\mbox 11141
mcaption (package) 825
\mcolrowcell 8806
mdframed (package) 124, 826
\mdseries 11673
media9 (package) 155, 835
memhfixc (package) 837
memoir (class) 163, 164, 1021
metalogo (package) 837
metalogox (package) 838
mhchem (package) 838
microtype (package) 223, 841
midfloat (package) 842
midpage (package) 842
\MiKTeX 12061
minibox (package) 842
minipage (environment) 11040
\minipagefullwidth 11028

minitoc (package) 843
morefloats (package) 843
moreverb (package) 844
movie15 (package) 155, 845
mparhack (package) 846
\mrowcell 8803
MS-Windows (program) 114, 213
multicap (package) 847
multicol (package) 847
multicolrule (package) 848
\multicolumnrow 39, 8675
multimedia (package) 155, 849
\multirow 851
multirow (package) 850
multitoc (package) 853
multline (environment) 62
multline* (environment) 65
musicography (package) 853

N

nameauth (package) 856
\Nameref 9192
\nameref 9183
nameref (package) 858
natbib (package) 858
nccfancyhdr (package) 859
needspace (package) 860
newclude (package) 165
\newfloat 4
newfloat (package) 227
\newline 11803
\newpage 11800
\newtheorem 401
newtxmath (package) 151
newunicodechar (package) 99
nextpage (package) 860
nfssext-cfr (package) 860
nicefrac (package) 150, 869
niceframe (package) 870
\noalign 8816
\nolinebreak 11874
\nolinkurl 9255
nomencl (package) 132, 870
nonfloat (package) 871
nonumonpart (package) 871
\nopagebreak 11878
\nopagecolor 76
nopageno (package) 871
\normalcolor 9
\normalfont 11708
\normalmarginpar 6008
notes (package) 871
notespages (package) 872
nowidow (package) 872
ntheorem (package) 149, 873
\numberline 9613
numindex (option) [tocbibind] 145, 998

O

octave (package) 884
OpenOffice (program) 72
option:
  --shell-escape 98
  [lwarpmk]:
    htmlglossary 131, 752
    printglossary 131, 752
  [tocbibind]:
    numindex 145, 998
  [tocloft]:
    titles 130
  BaseJobname 105, 217
  dvipdfm 101, 219
  dvipdfmx 101, 219
  dvips 101, 219
  GlossaryCmd 104, 131, 219, 752
  HomeHTMLFilename 101, 106, 218
  HTMLFilename 101, 106, 218
  HTMLIndexCmd 103, 219
  HTMLLatexCmd 101, 168, 218
  ImagesDirectory 101, 217
  ImagesName 101, 217
  latexmk 101, 106, 219
  LatexmkIndexCmd 104, 219
  lwarpmk 104, 218, 298
  makeindex 101, 106, 219
  makeindexStyle 103, 142, 217
  mathjax 101, 106, 217
  mathsvg 101, 106, 216
  OSWindows 104, 114, 213, 218
  pdftotextEnc 104, 218
  PrintIndexCmd 103, 218
  PrintLatexCmd 101, 168, 218
  warpHTML 104, 216
  warpprint 104, 216
  xindy 103, 106, 219
  xindyCodepage 103, 217
  xindyLanguage 103, 217
  xindyStyle 103, 143, 217
\OSPathSymbol 772
OSWindows (option) 104, 114, 213, 218
overpic (package) 155, 886

P

package:
  2in1 594
  2up 594
  a4 594
  a4wide 595
  a5comb 595
  abstract 130, 595
  academicons 597
  accessibility 598

accsupp 599
acro 599
acronym 601
addlines 603
adjmulticol 602
afterpage 598, 603
algorithm2e 604
algorithmicx 161, 607
alltt 608
amsmath 609
amsthm 612
anonchap 616
anysize 616
appendix 130, 617
ar 617
arabicfront 619
array 619
arydshln 620
asymptote 155, 622
atbegshi 623
attachfile 623
attachfile2 624
authblk 130, 626
autonum 627
awesomebox 628
axessibility 629
axodraw2 630
babel 165
backnaur 630
backref 630
balance 631
bbding 631
biblatex 636
bibunits 636
bigdelim 161, 637
bigfoot 638
bigstrut 639
bitpattern 639
blowup 640
bm 640
booklet 640
bookmark 641
booktabs 641
bophook 643
bounddvi 644
boxedminipage2e 644
braket 644
breakurl 645
breqn 645
bsheaders 647
bxpapersize 647
bytefield 647
calc 226
cancel 648
canoniclayout 649
capt-of 490
caption 161, 234, 490, 649
cases 652
changebar 653
changelayout 653
changepage 654
changes 655
chappg 657
chapterbib 657
chemfig 657
chemformula 659
chemgreek 664
chemmacros 665
chemnum 683
chkfloat 684
chngpage 684
cite 684
CJK 685
CJKutf8 685
cleveref 126, 549
clrdblpg 685
cm-super 99
cmap 100
cmdtrack 685
color 153, 686
colortbl 160, 686
comment 222
continue 688
copyrightbox 688
crop 689
ctable 689
cuted 692
cutwin 692
dblfloatfix 693
dblfnote 693
dcolumn 693
dejavu 99
diagbox 694
dingbat 695
ditaa 171
dprogress 696
draftcopy 696
draftfigure 697
draftwatermark 697
easy-todo 697
ebook 698
ed 699
ellipsis 699
embrac 700
emptypage 700
endfloat 701
endheads 701
endnotes 130, 702
enumerate 703
enumitem 703

environ 228
epigraph 704
epsfig 705
epstopdf 154, 706
epstopdf-base 706
eqlist 707
eqparbox 707
errata 708
eso-pic 710
etoolbox 195
eurosym 710
everyhook 226
everypage 711
everyshi 711
expl3 226
extramarks 711
fancybox 123, 712
fancyhdr 717
fancyref 718
fancytabs 719
fancyvrb 719
figcaps 725
figsize 726
filecontents 226
fitbox 726
fix2col 727
fixme 166, 727
fixmetodonotes 728
flafter 729
flippdf 729
float 161, 729
floatflt 731
floatpag 732
floatrow 162, 732
fltrace 737
flushend 737
fnbreak 738
fncychap 738
fnlineno 739
fnpara 739
fnpos 739
fontawesome 740
fontawesome5 741
fontaxes 742
fontenc 99, 742
fontspec 99
footmisc 743
footnote 744
footnotebackref 744
footnotehyper 745
footnoterange 745
footnpag 745
foreign 745
forest 746
framed 746
ftcap 749
ftnright 749
fullminipage 749
fullpage 749
fullwidth 750
fwlw 750
gentombow 750
geometry 223, 751
gettitlestring 226
gloss 131, 751
glossaries 131, 752
gmeometric 754
graphics 151, 754
graphicx 151, 767
grffile 153, 767
grid 767
grid-system 768
gridset 768
hang 768
hanging 770
hypbmsec 771
hypcap 771
hypdestopt 771
hypernat 771
hyperref 127, 479, 772
hyperxmp 777
hyphenat 777
idxlayout 778
ifoddpage 778
ifplatform 195
imakeidx 779
indentfirst 345
index 783
inputenc 99
inputenx 99
inputtrc 784
intopdf 785
karnaugh-map 785
keyfloat 163, 788
kotex 167
kvoptions 215
layaureo 794
layout 794
layouts 794
leading 797
letltxmacro 195
letterspace 797
lettrine 798
lineno 798
lips 801
listings 801
listliketab 805
lltjext 805
lmodern 99
longtable 159, 806

lscape 809
ltablex 809
ltcaption 810
ltxgrid 810
ltxtable 811
lua-check-hyphen 811
lua-visual-debug 811
luacolor 811
luatodonotes 166, 812
lwarp 101
lwarp-common-multimedia 1082
lwarp-patch-komascript 1055
lwarp-patch-memoir 1057
lyluatex 814
magaz 816
makeidx 144, 816, 997, 998
manyfoot 817
marginal 819
marginfit 819
marginfix 819
marginnote 820
marvosym 820
mathtools 149, 821
mcaption 825
mdframed 124, 826
media9 . . . . . 155, 835
memhfixc 837
metalogo 837
metalogox 838
mhchem 838
microtype 223, 841
midfloat 842
midpage 842
minibox 842
minitoc 843
morefloats 843
moreverb 844
movie15 . . . . . 155, 845
mparhack 846
multicap 847
multicol 847
multicolrule 848
multimedia 155, 849
multirow 850
multitoc 853
musicography 853
nameauth 856
nameref 858
natbib 858
nccfancyhdr 859
needspace 860
newclude 165
newfloat 227
newtxmath 151
newunicodechar 99
nextpage 860
nfssext-cfr 860
nicefrac 150, 869
niceframe 870
nomencl 132, 870
nonfloat 871
nonumonpart 871
nopageno 871
notes 871
notespages 872
nowidow 872
ntheorem 149, 873
octave 884
overpic 155, 886
pagegrid 886
pagenote 130, 887
pagesel 887
paralist 887
parallel 888
parcolumns 890
parnotes 891
parskip 892
pbox 893
pdfcol 893
pdfcolfoot 894
pdfcolmk 894
pdfcolparallel 894
pdfcolparcolumns 895
pdfcomment 895
pdfcrypt 896
pdflscape 896
pdfmarginpar 896
pdfpages 896
pdfprivacy 899
pdfrender 899
pdfsync 899
pdftricks 154, 900
pdfx 900
perltex 170
perpage 901
pfnote 902
phfqit 902
physics 151, 903
pifont 908
placeins 909
plarydshln 909
plext 909
plextarydshln 910
plextcolortbl 911
polyglossia 165
prelim2e 911
prettyref 911
preview 911
printlen 228
psfrag 154, 912

psfragx 912
pst-eps 913
pstool 155, 913
pstricks 154, 914
pxatbegshi 914
pxeveryshi 914
pxftnright 915
pxjahyper 915
pythontex 170
quotchap 915
quoting 916
ragged2e 917
realscripts 917
refcheck 918
refcount 227
register 919
relsize 120, 920
repeatindex 921
resizegather 922
rmpage 922
romanbar 922
romanbarpagenumber 922
rotating 923
rotfloat 923
rterface 170
rviewport 924
sagetex 170
savetrees 924
scalefnt 925
schemata 925
scrextend 925
scrhack 928
scrlayer 928
scrlayer-notecolumn 930
scrlayer-scrpage 930
scrpage2 931
section 932
sectionbreak 933
sectsty 934
semantic-markup 934
setspace 935
shadow 936
shapepar 937
showidx 937
showkeys 937
showtags 938
sidecap 938
sidenotes 939
SIunits 940
siunitx 150, 542, 941
slantsc 951
slashed 952
soul 952
soulpos 954
soulutf8 954
splitidx 955
srcltx 956
srctex 957
stabular 957
stackengine 957
steinmetz 959
stfloats 960
struktex 960
subcaption 161, 961
subfig 162, 961
subfigure 966
supertabular 160, 967
svg 968
sympytex 170
syntonly 969
tabfigures 969
tablefootnote 969
tabls 970
tabularx 970
tabulary 970
tascmac 971
textarea 972
textcomp 100, 120, 973
textfit 976
textpos 977
theorem 977
thinsp 981
threadcol 981
threeparttable 982
threeparttablex 160, 983
thumb 984
thumbs 984
tikz 153, 984
titleps 986
titleref 989
titlesec 989
titletoc 991
titling 130, 992
tocbasic 997
tocbibind 144, 145, 997, 998
tocdata 999
tocenter 1000
tocloft 130, 145, 616, 1001
tocstyle 1007
todo 1007
todonotes 166, 1008
topcapt 1010
tram 1010
transparent 1010
trimclip 1011
trivfloat 161, 1012
truncate 1013
turnthepage 1013
twoup 1013
typearea 1013

typicons 1014
ulem 1015
umoline 1016
underscore 1017
unicode-math 1017
units 150, 1018
unitsdef 1019
upref 1020
url 127, 1020
uspace 1020
varioref 126
varwidth 228
verbatim 227
verse 164, 1021
versonotes 1022
vertbars 1023
vmargin 1023
vowel 1024
vpe 1024
vwcol 1024
wallpaper 1027
watermark 1027
widetable 1027
widows-and-orphans 1028
witharrows 1028
wrapfig 1029
xbmks 1031
xcolor 153, 545, 1031
xechangebar 1042
xellipsis 1042
xetexko-vertical 1043
xfakebold 1043
xfrac 1044
xifthen 227
xltabular 1046
xltxtra 1046
xmpincl 1047
xparse 166, 226
xpatch 195
xpiano 1047
xpinyin 1048
xr 1048
xr-hyper 1049
xstring 228
xtab 160, 1049
xunicode 1051
xurl 1052
xy 1052
zhlineskip 1053
zwpagelayout 1054
\PackageDiagramAltText 111, 9863
\pagebreak 11875
\pagecolor 75
pagegrid (package) 886
pagenote (package) 130, 887
\pagenumbering 5327
\pageref 9178
\pagerefPageFor 9177
pagesel (package) 887
\pagestyle 5319
Pandoc (program) 72
\paragraph 6622
paralist (package) 887
parallel (package) 888
\parbox 11132
parcolumns (package) 890
parnotes (package) 891
\parsemulticolumnalignment 8449
parskip (package) 892
\part 6589
pbox (package) 893
pdfcol (package) 893
pdfcolfoot (package) 894
pdfcolmk (package) 894
pdfcolparallel (package) 894
pdfcolparcolumns (package) 895
pdfcomment (package) 895
pdfcrop (program) [requirement] 75
pdfcrypt (package) 896
pdfLaTeX (program) [requirement] 75
pdflscape (package) 896
pdfmarginpar (package) 896
pdfpages (package) 896
pdfprivacy (package) 899
pdfrender (package) 899
pdfseparate (program) [requirement] 75, 80
pdfsync (package) 899
pdftocairo (program) 152, 544
pdftocairo (program) [requirement] 75, 80
pdftotext (program) [requirement] . 75, 80
pdftotextEnc (option) 104, 218
pdftricks (package) 154, 900
pdfx (package) 900
perl (program) [requirement] 80
perltex (package) 170
perpage (package) 901
pfnote (package) 902
\phantomsection 11939
phfqit (package) 902
physics (package) 151, 903
picture (environment) 552, 11005
pifont (package) 908
placeins (package) 909
plarydshln (package) 909
Plastex (program) 71
plext (package) 909
plextarydshln (package) 910
plextcolortbl (package) 911
\PN@parnotes@auto 5681
polyglossia (package) 165

\postbookname 6569
\postchaptername 6573
\postpartname 6571
\postsectionname 6575
\prebookname 6569
\prechaptername 6573
prelim2e (package) *911*
\prepartname 6571
\presectionname 6575
prettyref (package) *911*
preview (package) *911*
\printauthor *387*, 6917, 6936
\printdate *387*, 6928, 6938
printglossary (option) [lwarpmk] *131*, *752*
\printindex 2
PrintIndexCmd (option) *103*, *218*
PrintLatexCmd (option) *101*, *168*, *218*
printlen (package) *228*
\printthanks *386*, *389*
\printtitle *387*, 6901, 6935
program:
  [requirement]:
    LuaLaTeX *75*
    MathJax *75*
    pdfcrop *75*
    pdfLaTeX *75*
    pdfseparate *75*, *80*
    pdftocairo *75*, *80*
    pdftotext *75*, *80*
    perl *80*
    XeLaTeX *75*
  Adobe *72*
  AsciiDoc *72*
  AsciiDoctor *72*
  Asciidoctor-LaTeX *72*
  epstopdf *152*, *544*
  Flare *72*
  FrameMaker *72*
  GELLMU *71*
  GladTeX *71*
  Hevea *71*
  htlatex *71*
  InDesign *72*
  LaTeX2HTML *71*
  latexmk *169*
  LaTeXML *71*
  LibreOffice *72*
  Linux *114*, *213*
  lwarpmk *182*, *299*
  lwarpmk epstopdf *152*, *544*
  lwarpmk pdftosvg *152*, *544*
  Mac OS *114*, *213*
  Madcap *72*
  make *171*
  makeindex *142*
  MathJax *146*, *147*
  MS-Windows *114*, *213*
  OpenOffice *72*
  Pandoc *72*
  pdftocairo *152*, *544*
  Plastex *71*
  TeX2page *71*
  TeX4ht *71*
  TeXMaths *176*
  TtH *71*
  Unix *114*, *213*
  Windows *114*, *213*
  Word *72*
  xindy *143*
project.css (file) *113*
project.lwarpmkconf (file) *256*
psfrag (package) *154*, *912*
psfragx (package) *912*
pst-eps (package) *913*
pstool (package) *155*, *913*
pstricks (package) *154*, *914*
pxatbegshi (package) *914*
pxeveryshi (package) *914*
pxftnright (package) *915*
pxjahyper (package) *915*
pythontex (package) *170*

Q

\qquad *578*, 11823
\quad *578*, 11818
quotation (environment) 7087
quotchap (package) *915*
quote (environment) 7081
quoting (package) *916*

R

ragged2e (package) *917*
\raggedbottom 5323
\raggedleft 10728
\raggedright 10734
\raisebox 11317
realscripts (package) *917*
\ref 9166
refcheck (package) *918*
refcount (package) *227*
\reflectbox 391
register (package) *919*
relsize (package) *120*, *920*
repeatindex (package) *921*
[requirement]:
  LuaLaTeX (program) *75*
  MathJax (program) *75*
  pdfcrop (program) *75*
  pdfLaTeX (program) *75*
  pdfseparate (program) *75*, *80*
  pdftocairo (program) *75*, *80*

pdftotext (program) 75, 80
perl (program) 80
XeLaTeX (program) 75
\RequirePackage 1174
\resizebox 400
resizegather (package) 922
\ResumeTabular 8720
\reversemarginpar 6007
\rightline 10742
\rmfamily 11681
rmpage (package) 922
romanbar (package) 922
romanbarpagenumber (package) 922
\rotatebox 349
rotating (package) 923
rotfloat (package) 923
\rowcolor 7923
\rownum 7915
rterface (package) 170
\rule 11882
rviewport (package) 924

S

sagetex (package) 170
sample_project.css (file) 113, 294
savetrees (package) 924
\sb 11711
\scalebox 372
scalefnt (package) 925
schemata (package) 925
scrextend (package) 925
scrhack (package) 928
scrlayer (package) 928
scrlayer-notecolumn (package) 930
scrlayer-scrpage (package) 930
scrpage2 (package) 931
\scshape 11691
\section 6613
section (package) 932
sectionbreak (package) 933
sectsty (package) 934
semantic-markup (package) 934
\SetHTMLFileNumber 5186
setspace (package) 935
\sffamily 11683
\sfrac 1044
shadow (package) 936
shapepar (package) 937
showidx (package) 937
showkeys (package) 937
showtags (package) 938
sidecap (package) 938
sidenotes (package) 939
SideTOCDepth (counter) 108, 494
\sidetocname 108, 9589
\simplechapterdelim 6393
\sishape 11698
SIunits (package) 940
siunitx (package) 150, 542, 941
slantsc (package) 951
slashed (package) 952
\sloppy 5325
\slshape 11706
soul (package) 952
soulpos (package) 954
soulutf8 (package) 954
\sp 11710
splitidx (package) 955
srcltx (package) 956
srctex (package) 957
stabular (package) 957
stackengine (package) 957
\StartDefiningMath 5144
\StartDefiningTabulars 5134
steinmetz (package) 959
stfloats (package) 960
\StopDefiningMath 5148
\StopDefiningTabulars 5138
struktex (package) 960
subcaption (package) 161, 961
subfig (package) 162, 961
subfigure (package) 966
\subparagraph 6625
\subsection 6616
\subsubsection 6619
supertabular (package) 160, 967
svg (package) 968
sympytex (package) 170
syntonly (package) 969

T

tabbing (environment) 7164
tabfigures (package) 969
tablefootnote (package) 969
\tableofcontents 110, 9519
tabls (package) 970
tabular (environment) 8865
\TabularMacro 8718
tabularx (package) 970
tabulary (package) 970
tascmac (package) 971
\TeX 11977
TeX2page (program) 71
TeX4ht (program) 71
TeXMaths (program) 176
textarea (package) 972
\textbf 11344
\textcolor 63
textcomp (package) 100, 120, 973
\texteb 11354
textfit (package) 976
\textgreater 5179

\textit 11414
\textless 5177
\textlg 11364
\textmd 11334
\textnormal 11473
textpos (package) 977
\textrm 11374
\textsc 11424
\textsf 11384
\textsi 11444
\textsl 11463
\textsubscript 11714
\textsuperscript 11712
\texttt 11394
\textulc 11434
\textup 11404
\tfl@chapter@fix 1012
\thanks 116
\thanksmarkseries 91
\theauthor 387
thebibliography (environment) 9785
\thedate 387
\theHTMLSection 6659
\theHTMLTitleSection 6658
\theHTMLTitleSeparator 6635
theindex (environment) 9685
\thempfootnote 5939
theorem (package) 977
\thetitle 387
thinsp (package) 981
\ThisAltText 111, 9860
\thispagestyle 5320
threadcol (package) 981
threeparttable (package) 982
threeparttablex (package) 160, 983
thumb (package) 984
thumbs (package) 984
tikz (package) 153, 984
\title 116, 5836
titlepage (environment) 116, 6885
titleps (package) 986
titleref (package) 989
titles (option) [tocloft] 130
titlesec (package) 989
titletoc (package) 991
titling (package) 130, 992
titlingpage (environment) 14, 116
tocbasic (package) 997
[tocbibind]:
numindex (option) 145, 998
tocbibind (package) . . . 144, 145, 997, 998
tocdata (package) 999
tocdepth (counter) 108
tocenter (package) 1000
[tocloft]:
titles (option) 130
tocloft (package) 130, 145, 616, 1001
tocstyle (package) 1007
todo (package) 1007
todonotes (package) 166, 1008
topcapt (package) 1010
\tracinglwarp 190, 1336
tram (package) 1010
transparent (package) 1010
trimclip (package) 1011
trivfloat (package) 161, 1012
truncate (package) 1013
\ttfamily 11685
TtH (program) 71
turnthepage (package) 1013
tutorial.tex (file) 81
twoup (package) 1013
typearea (package) 1013
typicons (package) 1014

U

\ulcshape 11693
ulem (package) 1015
umoline (package) 1016
\underline 11722
underscore (package) 1017
unicode-math (package) 1017
units (package) 150, 1018
unitsdef (package) 1019
Unix (program) 114, 213
\up 11720
upref (package) 1020
\upshape 11687
\url 9268
url (package) 127, 1020
\UseMinipageWidths 121, 553, 11029
usingOSWindows (boolean) 215
uspace (package) 1020

V

varioref (package) 126
varwidth (package) 228
verbatim (environment) 7154
verbatim (package) 227
\VerbatimHTMLWidth (length) 398
\verbatiminput 7146
verse (environment) 2, 7095
verse (package) 164, 1021
versonotes (package) 1022
vertbars (package) 1023
\vleftmargini (length) 164, 397, 1021
\vleftskip (length) 164, 397, 1021
vmargin (package) 1023
vowel (package) 1024
vpe (package) 1024

vwcol (package) 1024

W

wallpaper (package) 1027
warpall (environment) 115, 921
warpHTML (environment) 112, 114, 922
warpHTML (option) 104, 216
\warpHTMLonly 112, 115, 912
warpingHTML (boolean) 216
warpingprint (boolean) 216
warpMathJax (environment) 115, 928
warpprint (environment) . . . 111, 115, 925
warpprint (option) 104, 216
\warpprintonly 112, 115, 911
watermark (package) 1027
widetable (package) 1027
widows-and-orphans (package) 1028
Windows (program) 114, 213
witharrows (package) 1028
Word (program) 72
WPMarkFloats (boolean) 175, 243
WPMarkLOFT (boolean) 176, 243
WPMarkMath (boolean) 176, 244
WPMarkMinipages (boolean) 176, 243
WPMarkTOC (boolean) 176, 243
WPTitleHeading (boolean) 177, 244
wrapfig (package) 1029

X

xbmks (package) 1031
xcolor (package) 153, 545, 1031
xechangebar (package) 1042
\XeLaTeX 12024
XeLaTeX (program) [requirement] 75
xellipsis (package) 1042
\XeTeX 12024
xetexko-vertical (package) 1043
xfakebold (package) 1043
xfrac (package) 1044
\xfracHTMLfontsize 3
xifthen (package) 227
xindy (option) 103, 106, 219
xindy (program) 143
xindyCodepage (option) 103, 217
xindyLanguage (option) 103, 217
xindyStyle (option) 103, 143, 217
xltabular (package) 1046
xltxtra (package) 1046
xmpincl (package) 1047
xparse (package) 166, 226
xpatch (package) 195
xpiano (package) 1047
xpinyin (package) 1048
xr (package) 1048
xr-hyper (package) 1049
xstring (package) 228
xtab (package) 160, 1049
xunicode (package) 1051
xurl (package) 1052
xy (package) 1052

Z

zhlineskip (package) 1053
zwpagelayout (package) 1054

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

\, . . . 119
~ . . . 119

A

accents
  in section & file names . . . 367
accessibility . . . 97
adapting
  class . . . 181
  document . . . 96
  package . . . 180
affiliation
  multiple authors . . . 129
algorithmicx
  with newfloat, trivfloat . . . 1012
alt text . . . 97
array
  mhchem . . . 839
audio . . . 155
author
  HTML meta tag . . . 117, 347
  multiple . . . 129

B

baseline
  tabular . . . 428
bibliography
  HTML page and TOC . . . 129
bitmapped fonts . . . 98
bugs . . . 184

C

Calibre . . . 172
chemistry
  Greek symbols . . . 664
class
  modifying for lwarp . . . 181
code listings . . . 119
compiling
  custom . . . 168
Computer Modern . . . 98
converting
  class . . . 181
  document . . . 96
  package . . . 180
CSS
  file selection . . . 113
  lwarp.css . . . 113
  per HTML page . . . 113
  project-specific changes . . . 113
ctable . . . 160

D

danger icon . . . 194
debugging . . . 184
  HTML debug comments . . . 237
  tracing log . . . 237
Deja Vu . . . 99
description
  HTML meta tag . . . 117, 347
display math
  complicated objects . . . 148
document
  convert existing . . . 96
DVI LaTeX . . . 91, 98
dynamic math . . . 148
dynamic math expressions . . . 321

E

endnotes
  HTML page and TOC . . . 129
EPS image
  converting . . . 93
  using . . . 152, 543
EPUB
  conversion software . . . 172
  HTML conversion settings . . . 172, 242
error messages . . . 184
export
  to word processor . . . 174

F

FAQ . . . 184
filename
  accent in . . . 367
  hashed . . . 513, 533
  international languages . . . 164
  length . . . 109
  simplify . . . 125
  underscore in . . . 106, 120
  unique . . . 109

font
Computer Modern 98
Deja Vu 99
ligatures 99
packages 99
selection 98
size
lateximage 146, 530
math, SVG 146, 530
xfrac 1044
type 1 vector 98
type 3 bitmapped 98
footnotes 348
numbering 127
foreign
section names 164
framed objects 122
Frequently Asked Questions 184

G

generator
HTML meta tag 380
GIF images 152, 544
gloss 131
glossaries
HTML page and TOC 129
language 131
options 131
processing 92
graphics
file formats 152, 543
Greek
chemistry symbols 664

H

hash
SVG image filename 513, 533
heading, word processor 177
horizontal space 119
between minipages 578
HTML
alt text 97
conversion settings 106
debug comments 237
EPUB 172, 242
word processor 174, 242, 243
conversion suggestions 118
entities, conversion 119
filename generation 112
headings 193
meta tag
author 117, 347
description 117, 347
generator 380
title 109, 110, 117, 347
viewport 380
selecting print/HTML definitions . . 238
tabular column conversion 429
verbatim, in 119
hyperref
and xindy 133
title text 97

I

icon
warning 194
image
alt text 97
file formats 152, 543
GIF 152, 544
graphicx package 543
hashed filename 513, 533
PDF or EPS
converting 93, 152, 544
using 152, 543
PNG and JPG 152, 544
processing 299
\includegraphics
using 152, 543
index
custom makeindex style file 142
custom xindy style file 143
custom display styles 144
formatting 501
HTML page and TOC 129
letter headings 779
placement and TOC options . . . 144, 997
processing 85, 86, 133
see, seealso, ranges 133
source code 133
table of contents 144, 997
UTF-8 100
xindy and hyperref 133
inline math
complicated objects 148
international
section names 164
item
empty 402

J

JavaScript
MathJax 145
JPG images 152, 544

L

label
in HTML 383
math environment 525
language
glossaries 131
localization 97
language HTML metadata 378

- lateximage
  - font size . . . 146, 530
  - processing . . . 299
- Latin Modern font . . . 98
- LibreOffice
  - conversion recommendations . . . 177
  - import into . . . 174
  - section headings . . . 177
- ligatures . . . 99, 223
- line numbers . . . 194
- link
  - `title` text . . . 97
- list
  - empty item . . . 402
  - filename . . . 120
- listings, program code . . . 119
- locale . . . 171
- localization . . . 97
- LuaLaTeX
  - detection . . . 195
  - file & section names . . . 367
- lwarp
  - loading . . . 101
  - options . . . 101
- `lwarp.ist`
  - customizing . . . 142
- `lwarp.xdy`
  - customizing . . . 143
- *lwarpmk*
  - customizing . . . 182

M

- make utility . . . 171
- makeindex
  - customizing . . . 142
- margin
  - numbers . . . 194
  - tags . . . 194
- markup languages . . . 72
- math
  - `alt` text . . . 97
  - display with complicated objects . . . 148
  - dynamic . . . 148, 321
  - font size — SVG . . . 146, 530
  - inline with complicated objects . . . 148
  - MathJax custom functions . . . 147
  - MathJax summary . . . 146
  - `mathjax` option . . . 217
  - `mathsvg` option . . . 216
  - mhchem . . . 839
  - SVG summary . . . 146
  - word processor conversion . . . 176
- MathJax
  - accessibility . . . 97
  - custom functions . . . 147
  - custom script . . . 297
  - `mathjax` option . . . 217
  - summary . . . 146
- `\mcolrowcell` . . . 158
- MD5 hash
  - SVG image filename . . . 513, 533
- memoir
  - verse . . . 1021
- meta tag, HTML
  - author . . . 117, 347
  - description . . . 117, 347
  - generator . . . 380
  - title . . . 117, 347
  - viewport . . . 380
- minipage
  - framed . . . 122
  - horizontal space between . . . 578
- modifying
  - class . . . 181
  - document . . . 96
  - package . . . 180
- `\mrowcell` . . . 158
- `\multicolumn`
  - with `\multirow` . . . 158
- multimedia . . . 155
- multiple projects in a directory . . . 93
- `\multirow`
  - with `\multicolumn` . . . 158
- multirow
  - `\mrowcell` and `\mcolrowcell` . . . 158

N

- newfloat
  - with trivfloat, algorithmicx . . . 1012
- nomencl . . . 132
- numbers
  - left margin . . . 194

P

- package
  - modifying for lwarp . . . 180
  - required . . . 223
- `\par`
  - hooks . . . 343
- PDF image
  - converting . . . 93
  - using . . . 152, 543
- Perl . . . 80
- PNG images . . . 152, 544
- Poppler . . . 75, 80
- print
  - selecting print/HTML definitions . . . 238
- problems . . . 184
- program listings . . . 119
- programs
  - utility . . . 74
- projects . . . 93

\published 394

S

section
depths 193
file names 125
heading, word processor 177
international languages 164
settings
accessibility 97
CSS project-specific 113
CSS selection 113
filenames 106
HTML conversion 106
language 97
lwarp package options 101
selecting output 114
title page 116
shell escape 98
sidetoc
name 97
sidetoc
depth 106
siunitx
with *TeXMaths* 176
space
horizontal 119
between minipages 578
stack depths 193
\subtitle 394
SVG
converting from PDF or EPS 93
dynamic math 321
image processing 299
math summary 146
mathsvg option 216

T

tabular
baseline 428
HTML columnn conversion 429
macros inside 157, 409
\multicolumn with \multirow 158
multirow \mrowcell and \mcolrowcell 158
tikz
catcodes 985
dollar redefined 985
title
HTML meta tag 117, 347
titlepage
\subtitle and \published 394
tracing log 237
trivfloat
with newfloat, algorithmicx 1012
troubleshooting 184
HTML debug comments 237
tracing log 237
type 1 vector fonts 98
type 3 bitmapped fonts 98

U

underscore
filename 106, 120
Unicode
enhanced coverage 99
file & section names 367
input characters 209
selection 98
UTF-8
enhanced coverage 99
file & section names 367
index 100
locale 171
selection 98
utility
programs 74

V

vector fonts 98
verbatim
code and HTML tags 119
video 155
viewport
HTML meta tag 380

W

warning icon 194
word processor
conversion recommendations 177
HTML conversion settings 174, 242, 243
section headings 177

X

xcite 93
XeLaTeX
detection 195
file & section names 367
xindy
and hyperref 133
customizing 143
xparse
warnings 167
xr 93
xr-hyper 93

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 130, 595
acronym
  multiply-defined labels 601
\AddSubtitlePublished 394
affiliation 387
alt tags 148, 187
AMSmath
  ntheorem
    numbering 149, 873
appendix
  incorrect TOC link 130, 617
array
  chemformula 166
  MATHJAX 147
arydshln 620
audio 155
authblk
  \theauthor 386, 387
  titling 130, 386, 626, 992
author
  affiliation 387
  formatting 116
autonum 627

B

babel
  French 165
backref 630
backref 127
bibliography
  HTML page and TOC 129
bibtex
  \etalchar 131
  Improper \prevdepth 131
bigdelim 161, 637
bigfoot 128, 817
booktabs 641
boxes 121
breqn
  darray 645

C

Calibre
  EPUB conversion 172
caption
  numbering 162
  options 118
chemfig 166
chemformula
  MATHJAX 166, 659
chemgreek
  fontspec mapping 664
  text-mode symbols 664
chemmacros
  \makepolymerdelims 665
  redox reactions 665
colortbl 160, 686
cross reference
  incorrect link 188
  MATHJAX 147
  missing 188
CSS 189
ctable 160

D

ditaa 171
duplcate filename 125

E

\endhead, etc. 159
endnotes
  HTML page and TOC 129
\ensuremath 187
epstopdf 154, 706
EPUB
  encoding 172
  page order 172
  search order 172
  section breaks 173
error messages 184
\etalchar 131

F

fancybox
  \VerbatimFootnotes . . . 128, 713, 719
fancyvrb
  \VerbatimFootnotes . . . 128, 713, 719
figure
  macro in name . . . 187
file
  inaccessible . . . 108
  multiple projects in directory . . . 93
filename
  accents . . . 367
  corrupted . . . 125, 164
  duplicate . . . 125
  international, UTF-8 . . . 164
  Korean . . . 167
  macro in name . . . 125
  math in . . . 96, 126, 145
  Missing $ inserted . . . 186
fixme . . . 166, 727
float
  alignment . . . 119, 161
  numbering . . . 162
  out of sequence . . . 188
floatrow
  \FBwidth and \FBheight . . . 162, 732
  with subfig . . . 162, 732
floatrow
  \ttabbox . . . 157, 409
font
  ligatures . . . 99, 100
  missing symbols . . . 120, 973
  monospace . . . 100
  package conflicts . . . 99
  selection . . . 98
  UTF-8 . . . 98
fontspec . . . 99
footmisc . . . 127, 348
footnote
  displaymath . . . 128, 713, 719
  in math . . . 128
  in sectioning command . . . 127
  memoir . . . 128, 348
  numbering . . . 127
  paragraph tags . . . 128, 713, 719
  sectioning, footmisc . . . 127, 348
  sectioning, verbatim . . . 128, 713, 719
  title . . . 387
  verbatim . . . 128
  \VerbatimFootnotes . . . 128, 713, 719
forest . . . 746
formatting
  \bfseries etc. . . . 119, 563
frames . . . 122

G

gloss . . . 131
glossaries
  HTML page and TOC . . . 129
  makeglossaries not found . . . 131, 752
  page numbers . . . 132, 753
  style . . . 132, 752
graphics
  \rotatebox, \scalebox, \reflectbox . . . 153, 545
  image format priorities . . . 152, 544
  images out of order . . . 87, 188
  multimedia . . . 155
  optional arguments . . . 152, 544
  scale option . . . 96, 151, 543
  scaled . . . 755
  viewport . . . 152, 544
Greek packages . . . 664
grffile . . . 153, 767

H

horizontal spacing . . . 162, 961
HTML
  &, <, > . . . 119, 563
  alt tags . . . 148
  author . . . 347
  corrupted . . . 116, 121, 185, 186, 189
  entities . . . 119, 563
  image appear as HTML code . . . 87, 189
  inaccessible pages . . . 108, 110
    starred section . . . 129
  invalid . . . 119
  missing pages
    filename not unique . . . 109
    recompile . . . 189
    SideTOCDepth and FileDepth . . . 108
    SideTOCDepth and tocdepth . . . 108
  page did not update . . . 108, 189
  settings
    changed . . . 106
    undefined . . . 186
  validation . . . 119
\HTMLAuthor . . . 347
HTMLIndexCmd
  filenames . . . 103
hyperref
  backref . . . 127
  comments between arguments . . . 127
  incorrect link . . . 188

I

image
  appears as HTML code . . . 87, 189
  format priorities . . . 152, 544
  incorrect . . . 87, 188, 189
  viewport . . . 152, 544

Improper \prevdepth
  bibtex 131
  boxes 187
\includegraphics
  optional arguments 152, 544
  scale 96, 151, 543
index
  empty 143
  empty link 126, 188
  formatting 501
  HTML page and TOC 129
  missing entries 133
  numbers, not links 228
  reference ranges 143
  see and seealso 143
  styling references 143
  xindy and hyperref 143
  xstring bug 228

K

keyfloat 163, 788

L

label
  \nameref empty 126, 188
  characters 126, 188
Label(s) may have changed 187
\LateximageFontSizeName 530
Leaders not followed by proper glue . . 187
LibreOffice
  import 174
ligatures 100
LINUX 114, 213
list
  empty item 120, 402
listings 188
lists
  label formatting 402
locale 171
longtable
  \endhead, etc. 159
lrbox 121
ltxtable
  numbering 811
lualaTeX 99
\LWRbackslash 168
\LWRdollar 168
\LWRhash 168
\LWRopquote 168
\LWRopseq 168
\LWRpercent 168

M

MAC OS 114, 213
makeglossaries
  not found 131, 752
\makelabel 402
manyfoot 128, 817
\marginpar 120, 354
math
  alt tags 187
  appears as HTML code 87, 189
  chemformula 166
  custom macros 187
  dynamic 321
  equation numbering
    ntheorem 149, 873
  file name 96, 126, 145
  footnote 128
  in TeX boxes 146
  incorrect 87, 189, 321
  non-math contents 187
  section name 96, 126, 145
  slow or failed compile 187
  subequations 147
  tabbing 156, 400
  Ti*k*z 148
MATHJAX
  chemformula 166, 659
  custom script 297
  errors 148, 187, 321
  mathtools 149, 821
  mhchem 838
  \multicolumn 147
  \multirow 147, 159, 850
  physics 151
  references 147
  siunitx 150, 542, 941
  subequations 147
  unsupported packages 148, 187
MATHJAX
  arydshln 620
  booktabs 641
mathtools 149, 821
\mcolrowcell 158
media9 155
memoir
  footmisc 128, 348
  options clash 163, 1057
  version clash 163, 1057
mhchem
  MATHJAX 838
  nested dollar signs 839
minipage
  alignment 122, 553
  horizontal space between 578
  in a span 121, 552
  inline 121, 552
  multicols, width in 121, 553
  size 121, 553
  tabular, width in 121, 553

Misplaced \noalign 159, 807
tabular
rules 158, 410
Misplaced \omit
tabular 240
Misplaced alignment tab character &
ctable 160, 689
floatrow 162, 732
frames 123
supertabular 160, 967, 1049
tabular
macros 156, 320, 408
Missing $ inserted
filename or URL 186
Missing \begin{document}
package options 118
morewrites 143, 186
movie15 155
\mrowcell 158
MS-Windows 114, 213
multicol
\linewidth 122, 553
\multicolumn
MathJax 147
multimedia 155
multiple projects in a directory 93
multiply-defined labels
acronym 601
\multirow and \multicolumn 159, 850
multirow
MathJax 147, 159, 850
\mrowcell and \mcolrowcell 158

N

newclude 165
newtxmath 151
nicefrac 150
No room for a new \write 186
nomencl 132
ntheorem 874
font 149, 873
numbering 149, 873

O

operating system 114, 213
options
clash with memoir 163, 1057
with braces 118
overpic 155, 886

P

package
options with braces 118
version numbers with memoir 163, 1057
page
inacessible 108
page counter
references 126
SVG images 87, 189
page numbers 126, 188, 549
pdftricks 154, 900
perltex 170
pfnote
numbering 128, 902
physics
MathJax 151
polyglossia
Undefined control sequence . . 165, 591
poppler
Syntax Warning (ligature) 740
PrintIndexCmd 103
projects
multiple 93
psfrag 154
pstool 155, 913
pstricks 154, 914
pythontex 170

R

reference
% character between arguments . . 188
empty link 126, 188
incorrect link 188
label characters 126, 188
MathJax 147
missing or incorrect 188
page number 126, 188, 549
repeatindex 921
rterface 170

S

sample_projects.css
overwritten 113
\savebox 121
\sbox 121
scale (\includegraphics option) 96, 151, 543
sectioning
accents 367
duplicate name 125
international language 164
macro in name 125, 187
math in name 96, 126, 145
missing 108
starred section 129
word processor import 177
siunitx 150, 542, 941
MathJax 150, 542, 941
S column 158, 410
tabular S and s columns . 150, 542, 941
with *TeXMaths* 176
splitidx
\thepage and \AtWriteToIndex 134, 955

subcaption
numbering . . . 162
subfig
options . . . 162, 961
with floatrow . . . 162, 732
SVG image
appears as HTML code . . . 87, 189
incorrect . . . 87, 188, 189
math incorrect . . . 321
out of order . . . 87, 188
sympytex . . . 170
Syntax Warning (ligature) . . . 740

T

tabbing
math . . . 156, 400
table
macro in name . . . 187
numbering . . . 811
Table of Contents
missing . . . 110
tabular
baselines . . . 428
corrupt rows . . . 157, 409, 460
\endhead, \endfoot, \endlastfoot . . . 159, 806
inside an environment . . . 157, 409, 460
\kill . . . 159, 807
macro inside . . . 157, 409
Misplaced \noalign . . . 158, 410
Misplaced alignment tab character &
macros . . . 156, 408
rules . . . 320
\multicolumn with \multirow . . . 158
multirow \mrowcell and \mcolrowcell . . . 158
\newcolumntype . . . 157, 409
numbering . . . 1046
row corruption . . . 157, 409
rules . . . 158, 410
S column . . . 158, 410
TeXMaths . . . 176
text
&, <, > . . . 119, 563
\bfseries etc. . . . 119, 563
corrupted . . . 98
textcomp
missing symbols . . . 120, 973
\theauthor and authblk . . . 386, 387
threeparttablex . . . 160
tikz
in math . . . 148
matrices, & . . . 153, 984
title
affiliation . . . 387
newlines . . . 116
\thanks . . . 387
titling
authblk . . . 130, 386, 626, 992
hooks . . . 394
tocloft
chapter titles . . . 130, 145, 616, 1001
todonotes . . . 166, 812, 1008
tracing lwarp . . . 190
tram . . . 1010
transparent . . . 1011

U

Undefined control sequence
polyglossia . . . 165, 591
Unicode
fonts . . . 98
UTF-8 locale . . . 171
units . . . 150
UNIX . . . 114, 213
URL
Missing $ inserted . . . 186
\usebox . . . 121
UTF-8
locale . . . 171

V

verbatim
footnote . . . 128
framed . . . 124, 712
VerbatimFootnotes . . . 128, 713, 719
verse
spacing . . . 164, 397, 1021
version numbers
with memoir . . . 163, 1057
video . . . 155
viewport . . . 152, 544

W

warning messages . . . 184
warpall . . . 186
warpHTML . . . 114, 186
warpprint . . . 115, 186
WINDOWS . . . 114, 213
word processor
import . . . 174
sectioning headings . . . 177

X

xcite . . . 93
xeLaTeX . . . 99
xfakebold . . . 1043
xfrac . . . 1044
xindy
hyperref . . . 143
options
HTMLIndexCmd . . . 104

LatexmkIndexCmd 104
PrintIndexCmd 103
xltabular
numbering 1046
xparse 166
xr 93
xr-hyper 93
xstring 228

Index of Indexes

C

Change History 1088

G

General Index 1140

I

Index of Objects 1119

T

Troubleshooting Index 1144