The minitoc package

Jean-Pierre F. Drucbert, et al.

July 13, 2018
Summary

Table of contents
List of figures
List of tables
About this document

I User’s Manual
1 The minitoc package
2 Frequently Asked Questions
3 Memento
4 Examples of documents
5 Messages
6 Jargon
7 Installation
8 Postface

II Implementation
9 Commented code of the minitoc package
10 Commented code of the mtcoff package
11 Commented code of the mtcmess package
12 Patch for the memoir class
13 Language definition (.mld) and object (.mlo) files
# Table of contents

## Summary
2

## Table of contents
4

## List of figures
17

## List of tables
18

## About this document
20

## User’s Manual
22

### I The \texttt{minitoc} package
24

#### 1 Introduction
25
- 1.1 Important restrictions
26
- 1.2 Version
26

#### 1.2 License
26

#### 1.3 Using the \texttt{minitoc} package
27
- 1.3.1 Loading the package and creating the mini-tables
27
- 1.3.2 Preparing the mini-tables
27
- 1.3.3 Placing the mini-tables
29
- 1.3.4 Starred chapters, parts and sections
33

#### 1.4 Typesetting of the mini-tables
34
- 1.4.1 Chapter-level mini-tables
35
- 1.4.2 Titles for chapter-level mini-tables
35
- 1.4.3 Part-level mini-tables
35
- 1.4.4 Titles for part-level mini-tables
37
- 1.4.5 Section-level mini-tables
39
- 1.4.6 Titles for section-level mini-tables
39
- 1.4.7 Position of the titles
39
- 1.4.7.1 For mini-tables at the part level
39
- 1.4.7.2 For mini-tables at the chapter level
40
- 1.4.7.3 For mini-tables at the section level
40
- 1.4.7.4 Summary of the positioning of titles
40
- 1.4.8 Line spacing in the mini-tables
41
- 1.4.9 Simplified commands for fonts
41
- 1.4.10 Simplified command for mini-table titles
42
- 1.4.11 Simplified command for mini-table depths
42
Table of contents

1.4.12 Simplified command for mini-table offsets ............................... 42
1.4.13 Polymorphic entries in the mini-tables ................................ 43
1.4.14 Languages for the titles .................................................. 44
1.4.15 Altering the layout of the mini-tables .................................. 44
1.5 Special Features ............................................................... 46
   1.5.1 Horizontal Rules ....................................................... 46
   1.5.2 Page Numbers, Leaders ................................................ 47
   1.5.3 Features for parttocns and other mini-tables ......................... 47
   1.5.3.1 Remark about page styles ......................................... 49
   1.5.4 The “Chapter 0” Problem (solved) .................................. 49
   1.5.5 Special Entries in the TOC ............................................ 50
1.6 The notocite option .......................................................... 52
1.7 The listfiles and nolistfiles options ...................................... 52
1.8 The hints option .................................................................. 52
1.9 Usage with MS-DOS ............................................................ 54
1.10 Why several \LaTeX{} runs are required? .................................. 55
1.11 The mtcoff package .......................................................... 55

2 Frequently Asked Questions ..................................................... 56
   2.0 Introduction .................................................................... 57
   2.1 Avoiding a page break near the rules before and after a mini-table .... 58
   2.2 Implementing others layouts for a mini-table ............................ 58
   2.3 A “\" command in a contents line makes an error ......................... 58
   2.4 Reordering chapters makes havoc ........................................ 58
   2.5 Extensions for the names of auxiliary files ............................... 58
   2.6 Playing with the chapter number ........................................ 59
   2.7 Supported document classes .............................................. 60
   2.8 Compatibility with \LaTeX{} versions ...................................... 60
   2.9 Other mini-tables ............................................................ 60
   2.10 Why so many auxiliary files? ............................................. 61
   2.11 Mini-tables at levels other than chapter ................................ 61
   2.12 Incompatibility with \LaTeX{}2.09 .......................................... 62
   2.13 Documents resetting the chapter number at each part ................ 62
   2.14 The mini-tables have too much spaced lines ........................... 62
   2.15 The secttoc is wrong ..................................................... 62
   2.16 Removing the lines of dots ............................................... 62
   2.17 Using the hyperref package with minitoc .............................. 62
   2.18 Problem while upgrading minitoc ....................................... 63
   2.19 A local TOC for the set of appendices .................................. 63
   2.20 Use with the appendix package ......................................... 64
   2.21 Use with the tocloft package .......................................... 64
   2.22 Use with the memoir class .............................................. 65
   2.23 There are too many commands for fonts, titles, and depths ........ 66
   2.24 Compatibility with the AMSTeX document classes ..................... 66
   2.25 Hiding some entries from the main table of contents ................ 67
   2.26 Defining your own .mld file ............................................ 70
   2.27 Use with the abstract package ......................................... 70
   2.28 Use with the sectsty package ........................................... 70
   2.29 Strange alignment in the minitocs ..................................... 71
Table of contents

2.30 Useful precautions with starred sectioning commands .......................... 72
2.31 Use with packages for captions .................................................. 72
2.32 Bad interaction minitoc/hyperref/memoir ........................................ 72
2.33 Use with the varsects package ..................................................... 73
2.34 Initial font settings ...................................................................... 73
2.35 Use with the KOMA-Script classes .................................................. 75
2.36 Use with the jura class or the alphabet package ................................. 75
2.37 The .mld files and the babel package .............................................. 75
2.38 Use with the fncychap package ..................................................... 75
2.39 Use with the quotchap package .................................................... 76
2.40 Use with the romannum package .................................................... 76
2.41 Use with the sfheaders package ................................................... 76
2.42 Use with the alphanum package .................................................... 76
2.43 Use with the captcont package ..................................................... 76
2.44 Vertical spaces (gaps) for parttocs, partlofs, and partlots titles .............. 76
2.45 Vertical spacing before the bottom rule of a minitable ......................... 77
2.46 Another interaction between the tocloft and minitoc packages .......... 78
2.47 Use with the hangcaption package ................................................. 79
2.48 Use with the flowfram package .................................................... 79

3 Memento ................................................................................. 80

4 Examples of documents ................................................................. 90
4.1 The mtc-2c.tex document file ....................................................... 91
4.2 The mtc-2nd.tex document file ..................................................... 92
4.3 The mtc-3co.tex document file ..................................................... 93
4.4 The mtc-add.tex document file .................................................... 96
4.5 The mtc-ads.tex document file ................................................... 100
4.6 The mtc-amm.tex document file .................................................. 105
4.7 The mtc-apx.tex document file ................................................... 105
4.8 The mtc-art.tex document file .................................................... 105
4.9 The mtc-bk.tex document file ..................................................... 110
4.10 The mtc-bo.tex document file .................................................... 115
4.11 The mtc-ch0.tex document file .................................................... 119
4.12 The mtc-cri.tex document file .................................................... 121
4.13 The mtc-fko.tex document file .................................................... 121
4.14 The mtc-fol.tex document file .................................................... 122
4.15 The mtc-fo2.tex document file .................................................... 123
4.16 The mtc-gap.tex document file ................................................... 125
4.17 The mtc-hi1.tex document file .................................................... 125
4.18 The mtc-hi2.tex document file .................................................... 125
4.19 The mtc-hia.tex document file .................................................... 125
4.20 The mtc-hir.tex document file .................................................... 126
4.21 The mtc-hop.tex document file ................................................... 127
4.22 The mtc-liv.tex document file .................................................... 128
4.23 The mtc-mem.tex document file .................................................. 132
4.24 The mtc-mm1.tex document file .................................................. 133
4.25 The mtc-mu.tex document file .................................................... 134
4.26 The mtc-nom.tex document file .................................................. 136
## Implementation

### 9 Commented code of the minitoc package

1. **Introduction**
2. **Identification code**
3. **A file descriptor to write**
4. **Indentation and skip**
5. **Tests and flags**
   1. Flags for the `hints` option
   2. Use of section-level mini-lists of floats
   3. Presence of some packages and classes
   4. Flags for packages dealing with floats
   5. Insertion of empty mini-tables
   6. Presence or absence of some sectioning commands
   7. Flags to check if some commands are used
   8. Check if the document has exactly 2 parts
6. **Preparation for the `notoccite` option**
7. **Preparation for the `tight` and `k-tight` options**
8. **Preparation to work with `hyperref`**
9. **Checking the presence of some packages**
   1. Check if the `sectsty` package is loaded, and when
   2. Check if the `varsects` package is loaded, and when
   3. Check if the `fncychap` package is loaded, and when
   4. Check if the `hangcaption` package is loaded, and when
   5. Check if the `quotchap` package is loaded, and when
   6. Check if the `romannum` package is loaded, and when
   7. Check if the `sfheaders` package is loaded, and when
   8. Check if the `alnumsec` package is loaded, and when
   9. Check if the `captcont` package is loaded, and when
   10. Check if the `caption` package is loaded, and when
   11. Check if the `caption2` package is loaded, and when
   12. Check if the `ccaption` package is loaded, and when
   13. Check if the `mcaption` package is loaded, and when
   14. Check if the `float` package is loaded
   15. Check if the `floatrow` package is loaded
   16. Check if the `rotfloat` package is loaded
   17. Check if the `trivfloat` package is loaded
9. **Is the `memoir` class loaded?**
10. **Testing the emptiness of a file**
11. **Internal macros to decrement minitoc counters**
12. **Patching the `\part` command**
13. **Adding an entry in the TOC for a starred part**
14. **Section level macros**
15. **Corrections for numbering**
16. **Patching the `\section` command**
17. **Adding an entry in the TOC for a starred section**
18. **Chapter level macros**
19. **Patching the `\chapter` command**
20. **Adding an entry in the TOC for a starred chapter**
# Table of contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.22</td>
<td>Miscellaneous declarations</td>
<td>285</td>
</tr>
<tr>
<td>9.23</td>
<td>Autoconfiguration of extensions</td>
<td>286</td>
</tr>
<tr>
<td>9.24</td>
<td>Detecting obsolete versions of \LaTeXX</td>
<td>287</td>
</tr>
<tr>
<td>9.25</td>
<td>Adding a TOC entry without leaders nor page numbers</td>
<td>288</td>
</tr>
<tr>
<td>9.26</td>
<td>Default values for the page-number customizations</td>
<td>288</td>
</tr>
<tr>
<td>9.27</td>
<td>“Features” for the mini-tables</td>
<td>291</td>
</tr>
<tr>
<td>9.28</td>
<td>Fake tables of contents</td>
<td>294</td>
</tr>
<tr>
<td>9.29</td>
<td>Depth counters for minilofs and minilots</td>
<td>294</td>
</tr>
<tr>
<td>9.30</td>
<td>Chapter level commands</td>
<td>294</td>
</tr>
<tr>
<td>9.31</td>
<td>Starred parts, chapters or sections</td>
<td>295</td>
</tr>
<tr>
<td>9.32</td>
<td>Font commands for the mini-tables</td>
<td>297</td>
</tr>
<tr>
<td>9.33</td>
<td>Internal commands to position the mini-table titles</td>
<td>297</td>
</tr>
<tr>
<td>9.34</td>
<td>The mtcverse environment</td>
<td>298</td>
</tr>
<tr>
<td>9.35</td>
<td>The \minitoc, \minilof, and \minilot commands</td>
<td>298</td>
</tr>
<tr>
<td>9.35.1</td>
<td>The \minitoc command</td>
<td>298</td>
</tr>
<tr>
<td>9.35.2</td>
<td>The \minilof command</td>
<td>302</td>
</tr>
<tr>
<td>9.35.3</td>
<td>The \minilot command</td>
<td>305</td>
</tr>
<tr>
<td>9.36</td>
<td>Patching the \chapter command, continued</td>
<td>308</td>
</tr>
<tr>
<td>9.37</td>
<td>The \addstarred... commands</td>
<td>309</td>
</tr>
<tr>
<td>9.38</td>
<td>TOC entries without leaders</td>
<td>309</td>
</tr>
<tr>
<td>9.39</td>
<td>Mini-tables with or without leaders</td>
<td>310</td>
</tr>
<tr>
<td>9.40</td>
<td>The dominitoc command and its siblings</td>
<td>311</td>
</tr>
<tr>
<td>9.40.1</td>
<td>Analysis and splitting of the TOC file</td>
<td>314</td>
</tr>
<tr>
<td>9.41</td>
<td>Mini-lists of figures</td>
<td>318</td>
</tr>
<tr>
<td>9.41.1</td>
<td>Analysis and splitting of the list of figures file</td>
<td>318</td>
</tr>
<tr>
<td>9.42</td>
<td>Mini-lists of tables</td>
<td>320</td>
</tr>
<tr>
<td>9.42.1</td>
<td>Analysis and splitting of the list of tables file</td>
<td>321</td>
</tr>
<tr>
<td>9.43</td>
<td>Macro to write a contents line</td>
<td>323</td>
</tr>
<tr>
<td>9.44</td>
<td>Depth counters for partlofs and partlots</td>
<td>324</td>
</tr>
<tr>
<td>9.45</td>
<td>Part level commands</td>
<td>325</td>
</tr>
<tr>
<td>9.46</td>
<td>Fonts for the parttocs</td>
<td>326</td>
</tr>
<tr>
<td>9.47</td>
<td>Default titles for part-level mini-tables</td>
<td>326</td>
</tr>
<tr>
<td>9.48</td>
<td>The ptcverse environment</td>
<td>328</td>
</tr>
<tr>
<td>9.49</td>
<td>The part level mini-tables: \parttoc, \partlof, and \partlot</td>
<td>328</td>
</tr>
<tr>
<td>9.49.1</td>
<td>The \parttoc command</td>
<td>329</td>
</tr>
<tr>
<td>9.49.2</td>
<td>The \partlof command</td>
<td>331</td>
</tr>
<tr>
<td>9.49.3</td>
<td>The \partlot command</td>
<td>334</td>
</tr>
<tr>
<td>9.50</td>
<td>Auxiliary commands for printing parttocs</td>
<td>337</td>
</tr>
<tr>
<td>9.51</td>
<td>Patching the \part command, continued</td>
<td>338</td>
</tr>
<tr>
<td>9.52</td>
<td>The \doparttoc command and its siblings</td>
<td>338</td>
</tr>
<tr>
<td>9.52.1</td>
<td>Processing macros for the parttocs</td>
<td>340</td>
</tr>
<tr>
<td>9.52.2</td>
<td>Processing macros for the partlofs</td>
<td>344</td>
</tr>
<tr>
<td>9.52.3</td>
<td>Processing macros for the partlots</td>
<td>346</td>
</tr>
<tr>
<td>9.53</td>
<td>Depth counters for sectlofs and sectlots</td>
<td>349</td>
</tr>
<tr>
<td>9.54</td>
<td>Section-level commands</td>
<td>349</td>
</tr>
<tr>
<td>9.55</td>
<td>Fonts commands for secttocs and co.</td>
<td>350</td>
</tr>
<tr>
<td>9.56</td>
<td>Internal macros for title positionning</td>
<td>350</td>
</tr>
<tr>
<td>9.57</td>
<td>The stcverse environment</td>
<td>351</td>
</tr>
<tr>
<td>9.58</td>
<td>The \secttoc, \sectlof, and \sectlot commands</td>
<td>351</td>
</tr>
<tr>
<td>9.58.1</td>
<td>The \secttoc command</td>
<td>351</td>
</tr>
</tbody>
</table>
9.58.2 The \sectlof command ........................................ 354
9.58.3 The \sectlot command ...................................... 357
9.59 Auxiliary internal commands, section level .................... 359
9.60 Patching the \section command (continued) ...................... 360
9.61 The \dosecttoc command and siblings .......................... 361
9.62 End of section-level commands .................................. 371
9.63 The \mtcprepare command ...................................... 360
9.64 Use with \nofiles ............................................. 372
9.65 Necessary \@... commands ..................................... 373
9.66 The horizontal rules and their default values .................. 374
9.67 The \mtcset... commands ..................................... 375
9.67.1 Keywords for the \mtcset... commands ....................... 375
9.67.2 The \mtcsetfont command ................................... 377
9.67.3 The \mtcsettitlefont command .............................. 381
9.67.4 The \mtcsettitle command .................................. 382
9.67.5 The \mtcsetformat command .................................. 384
9.67.6 The \mtcsetpagenumbers command ............................ 388
9.67.7 The \mtcsetrules command ................................... 391
9.67.8 The \mtcsetfeature command ................................ 393
9.67.9 The \mtcsetdepth command ................................... 395
9.67.10 The \mtcsetoffset command ................................ 397
9.68 Polymorphic entries ........................................... 398
9.69 The \mtchideinmaintoc environment and siblings ................. 399
9.70 Fixing the “Glossary” entry in the TOC .......................... 401
9.71 Fixing the “Index” entry in the TOC ............................ 403
9.72 Fixing the “Nomenclature” entry in the TOC ...................... 404
9.73 The \mtcselectlanguage command ................................ 405
9.74 The \mtcloadmlo internal command ................................ 406
9.75 The “coffee breaks” ............................................ 407
9.76 Initialization of counters ...................................... 407
9.77 Declarations for simple options ................................ 407
9.77.1 Options tight and loose, k-tight and k-loose ................. 407
9.77.2 Options checkfiles and nocheckfiles ........................ 408
9.77.3 Options dotted and undotted ................................ 408
9.77.4 Option notoccite ............................................ 408
9.77.5 Option shortext ............................................. 408
9.78 The insection option .......................................... 408
9.79 The listfiles and nolistfiles options ............................ 409
9.80 Language options ............................................... 409
9.81 The hints option ................................................ 414
9.81.1 First part: \mtc@hints@begindoc ............................. 414
9.81.1.1 Hint about the alphanum package ........................ 415
9.81.1.2 Hint about the appendix package ......................... 415
9.81.1.3 Hint about the tocbibind package ......................... 415
9.81.1.4 Hint about the KOMA-Script classes ...................... 415
9.81.1.5 Hint about the tocloft package ........................... 416
9.81.1.6 Hint about the titlesec package ............................ 416
9.81.1.7 Hint about the titletoc package ........................... 417
9.81.1.8 Hint about the placeins package ......................... 417
9.81.1.9 Hint about the memoir class .............................. 418
Table of contents

9.81.1.10 Hint about the amsart and amsproc classes .......................... 419
9.81.1.11 Hint about the amsbook class ........................................ 419
9.81.1.12 Hint about the abstract package ...................................... 419
9.81.1.13 Hint about the jura class .............................................. 420
9.81.1.14 Hint about the flowfram package ................................. 420
9.81.1.15 Hint about the alteration of the sectionning commands .......... 420
  9.81.1.15.1 Alteration of \part ........................................... 421
  9.81.1.15.2 Alteration of \chapter ........................................ 421
  9.81.1.15.3 Alteration of \section ...................................... 422
9.81.1.16 Hint about the consistency of the calling sequences of the commands .......................................................... 422
9.81.2 Final part: \mtc@hints@enddoc ........................................... 423
  9.81.2.1 Hint about \sect-lof|lot and the insection option ................ 423
  9.81.2.2 Final part of the consistency tests ............................. 424
  9.81.2.3 Check if the main tables have been prepared (first part) ...... 426
  9.81.2.4 Check if the main tables have been prepared (second part) ... 428
  9.81.2.5 Check the number of mini-tables, in case of short extensions .. 430
  9.81.2.6 Final part of the hint about the sectsty package ............ 431
  9.81.2.7 Final part of the hint about the varsects package ........... 431
  9.81.2.8 Final part of the hint about the fncychap package .......... 432
  9.81.2.9 Final part of the hint about the hangcaption package ....... 432
  9.81.2.10 Final part of the hint about the quotchap package .......... 433
  9.81.2.11 Final part of the hint about the romannum package .......... 433
  9.81.2.12 Final part of the hint about the sfheaders package ........ 433
  9.81.2.13 Final part of the hint about the alnumsec package .......... 434
  9.81.2.14 Final part of the hint about the captecont package ......... 434
  9.81.2.15 Final part of the hint about the caption package ........... 434
  9.81.2.16 Final part of the hint about the caption2 package .......... 435
  9.81.2.17 Final part of the hint about the ecaption package ........ 435
  9.81.2.18 Final part of the hint about the mcaption package ........ 435
  9.81.2.19 Final part of the hint about the float package ............. 436
  9.81.2.20 Final part of the hint about the floatrow package ......... 436
  9.81.2.21 Final part of the hint about the trivfloat package ......... 436
  9.81.2.22 Final part of the hint about the rotfloat package ......... 437
  9.81.2.23 Check if empty mini-tables have been detected .......... 437
  9.81.2.24 Check if obsolete commands have been used .......... 439
  9.81.2.25 Check if some hints have been written ................ 440
9.82 Processing of options .................................................... 440
  9.82.1 Processing the insection option .................................. 441
  9.82.2 Processing the notoccite option .................................. 442
  9.82.3 Processing the listfiles option .................................. 442
  9.82.4 Processing the hints option ...................................... 443
  9.82.5 Saving the sectionning commands ................................ 443
9.83 Trapping the undefined preparation and insertion commands ......... 444
9.84 The minitoc-fr.dtx file ................................................. 445

10 Commented code of the mtoff package .................................. 447
  10.1 Why mtoff? ............................................................. 448
  10.2 Identification of the package ..................................... 448
Table of contents

10.3 Faking counters and dimensions .............................................. 449
10.4 Faking simple commands ..................................................... 452
10.5 Faking commands with one optional argument ............................ 453
10.6 Faking flags ......................................................................... 454
10.7 Disabling the internal commands ............................................. 454
10.8 Disabling the font commands ................................................... 455
10.9 Disabling the \mtcset... commands .......................................... 456
10.10 Disabling the \mtcpoly... commands ........................................ 456
10.11 Disabling the new \l@... commands ........................................ 456
10.12 Ignore the obsolete commands ................................................. 457
10.13 Disabling the \mtcselectlanguage and \mtcloadmlo commands .. 457
10.14 Disabling the commands for the horizontal rules ....................... 457
10.15 Disabling the commands for the page numbers ......................... 458
10.16 Disabling the mini-table features commands ......................... 458
10.17 Disabling miscellaneous flags and commands ......................... 459
10.18 Caution for some commands .................................................. 460
10.19 Disabling commands for “coffee” ........................................... 461
10.20 Disabling the mtchideinmain... environments ......................... 461
10.21 Inhibition of the \mtc@[save|restore]XXXdepth internal commands 462
10.22 Disabling the \mtcfixglossary command ................................ 462
10.23 Disabling the \mtcfixindex command ...................................... 462
10.24 Disabling the \mtcfixnomenclature command ......................... 462
10.25 Disabling the \addstarred... commands ................................ 463

11 Commented code of the mtcmess package ................................. 464

12 Patch for the memoir class ..................................................... 465

13 Language definition (.mld) and object (.mlo) files ...................... 467
13.1 Overview ............................................................................ 471
13.2 “Acadian” language: acadian.mld ........................................ 472
13.3 “Acadien” language: acadien.mld ......................................... 472
13.4 “Afrikaan” language: afrikaan.mld ....................................... 473
13.5 “Afrikaans” language: afrikaans.mld .................................... 473
13.6 “Albanian” language: albanian.mld ...................................... 473
13.7 “American” language: american.mld .................................... 474
13.8 “Arab” language: arab.mld .................................................. 474
13.9 “Arab2” language: arab2.mld .............................................. 475
13.10 “Arabi” language: arab1.mld .............................................. 475
13.11 “Arabic” language: arabic.mld .......................................... 476
13.12 “Armenian” language: armenian.mld .................................. 476
13.13 “Australian” language: australian.mld ................................. 476
13.14 “Austrian” language: austrian.mld ...................................... 477
13.15 “Bahasa” language: bahasa.mld ......................................... 477
13.16 “Bahasai” language: bahasai.mld ....................................... 477
13.17 “Bahasam” language: bahasam.mld .................................... 478
13.18 “Bangla” language: bangla1.mld ....................................... 478
13.19 “Basque” language: basque.mld ........................................ 479
13.20 “Bengali” language: bengali.mld ....................................... 479
13.21 “Bicig” language: bicig.mld ............................................. 479
<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>13.22</td>
<td>&quot;Bicig2&quot; language: bicig2.mld</td>
<td>480</td>
</tr>
<tr>
<td>13.23</td>
<td>&quot;Bicig3&quot; language: bicig3.mld</td>
<td>480</td>
</tr>
<tr>
<td>13.24</td>
<td>&quot;Bithe&quot; language: bithe.mld</td>
<td>480</td>
</tr>
<tr>
<td>13.25</td>
<td>&quot;Brazil&quot; language: brazil.mld</td>
<td>481</td>
</tr>
<tr>
<td>13.26</td>
<td>&quot;Brazilian&quot; language: brazilian.mld</td>
<td>482</td>
</tr>
<tr>
<td>13.27</td>
<td>&quot;Breton&quot; language: breton.mld</td>
<td>482</td>
</tr>
<tr>
<td>13.28</td>
<td>&quot;British&quot; language: british.mld</td>
<td>483</td>
</tr>
<tr>
<td>13.29</td>
<td>&quot;Bulgarian&quot; language: bulgarian.mld</td>
<td>483</td>
</tr>
<tr>
<td>13.30</td>
<td>&quot;Bulgarianb&quot; language: bulgarianb.mld</td>
<td>483</td>
</tr>
<tr>
<td>13.31</td>
<td>&quot;Buryat&quot; language: buryat.mld</td>
<td>484</td>
</tr>
<tr>
<td>13.32</td>
<td>&quot;Buryat2&quot; language: buryat2.mld</td>
<td>485</td>
</tr>
<tr>
<td>13.33</td>
<td>&quot;Canadian&quot; language: canadian.mld</td>
<td>485</td>
</tr>
<tr>
<td>13.34</td>
<td>&quot;Canadien&quot; language: canadien.mld</td>
<td>485</td>
</tr>
<tr>
<td>13.35</td>
<td>&quot;Castillian&quot; language: castillian.mld</td>
<td>486</td>
</tr>
<tr>
<td>13.36</td>
<td>&quot;Castillan&quot; language: castillan.mld</td>
<td>486</td>
</tr>
<tr>
<td>13.37</td>
<td>&quot;Catalan&quot; language: catalan.mld</td>
<td>486</td>
</tr>
<tr>
<td>13.38</td>
<td>&quot;Chinese1&quot; language: chinese1.mld[d,o]</td>
<td>487</td>
</tr>
<tr>
<td>13.40</td>
<td>&quot;Croatian&quot; language: croatian.mld</td>
<td>487</td>
</tr>
<tr>
<td>13.41</td>
<td>&quot;Czech&quot; language: czech.mld</td>
<td>488</td>
</tr>
<tr>
<td>13.42</td>
<td>&quot;Danish&quot; language: danish.mld</td>
<td>488</td>
</tr>
<tr>
<td>13.43</td>
<td>&quot;Devanagari&quot; language: devanagari.mld</td>
<td>489</td>
</tr>
<tr>
<td>13.44</td>
<td>&quot;Dutch&quot; language: dutch.mld</td>
<td>489</td>
</tr>
<tr>
<td>13.45</td>
<td>&quot;English&quot; language: english.mld</td>
<td>490</td>
</tr>
<tr>
<td>13.46</td>
<td>&quot;English1&quot; language: english1.mld</td>
<td>490</td>
</tr>
<tr>
<td>13.47</td>
<td>&quot;English2&quot; language: english2.mld</td>
<td>491</td>
</tr>
<tr>
<td>13.48</td>
<td>&quot;Esperant&quot; language: esperant.mld</td>
<td>492</td>
</tr>
<tr>
<td>13.49</td>
<td>&quot;Esperanto&quot; language: esperanto.mld</td>
<td>492</td>
</tr>
<tr>
<td>13.50</td>
<td>&quot;Estonian&quot; language: estonian.mld</td>
<td>492</td>
</tr>
<tr>
<td>13.51</td>
<td>&quot;Ethiopia&quot; language: ethiopia.mld</td>
<td>493</td>
</tr>
<tr>
<td>13.52</td>
<td>&quot;Ethiopian&quot; language: ethiopian.mld</td>
<td>493</td>
</tr>
<tr>
<td>13.53</td>
<td>&quot;Ethiopian2&quot; language: ethiopian2.mld</td>
<td>494</td>
</tr>
<tr>
<td>13.54</td>
<td>&quot;Farsi1&quot; language: farsi1.mld[d,o]</td>
<td>494</td>
</tr>
<tr>
<td>13.55</td>
<td>&quot;Farsi2&quot; language: farsi2.mld[d,o]</td>
<td>495</td>
</tr>
<tr>
<td>13.56</td>
<td>&quot;Farsi3&quot; language: farsi3.mld</td>
<td>495</td>
</tr>
<tr>
<td>13.57</td>
<td>&quot;Finnish&quot; language: finnish.mld</td>
<td>495</td>
</tr>
<tr>
<td>13.58</td>
<td>&quot;Finnish2&quot; language: finnish2.mld</td>
<td>496</td>
</tr>
<tr>
<td>13.59</td>
<td>&quot;Francais&quot; language: francais.mld</td>
<td>496</td>
</tr>
<tr>
<td>13.60</td>
<td>&quot;French&quot; language: french.mld</td>
<td>497</td>
</tr>
<tr>
<td>13.61</td>
<td>&quot;French1&quot; language: french1.mld</td>
<td>497</td>
</tr>
<tr>
<td>13.62</td>
<td>&quot;French2&quot; language: french2.mld</td>
<td>498</td>
</tr>
<tr>
<td>13.63</td>
<td>&quot;Frenchb&quot; language: frenchb.mld</td>
<td>498</td>
</tr>
<tr>
<td>13.64</td>
<td>&quot;Frenchle&quot; language: frenchle.mld</td>
<td>499</td>
</tr>
<tr>
<td>13.65</td>
<td>&quot;Frenchpro&quot; language: frenchpro.mld</td>
<td>499</td>
</tr>
<tr>
<td>13.66</td>
<td>&quot;Galician&quot; language: galician.mld</td>
<td>499</td>
</tr>
<tr>
<td>13.67</td>
<td>&quot;German&quot; language: german.mld</td>
<td>500</td>
</tr>
<tr>
<td>13.68</td>
<td>&quot;Germanb&quot; language: germanb.mld</td>
<td>500</td>
</tr>
<tr>
<td>13.69</td>
<td>&quot;Germanb2&quot; language: germanb2.mld</td>
<td>501</td>
</tr>
<tr>
<td>13.70</td>
<td>&quot;Greek&quot; language: greek.mld</td>
<td>501</td>
</tr>
<tr>
<td>Section</td>
<td>Language</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>----------</td>
<td>-------------</td>
</tr>
<tr>
<td>13.71</td>
<td>“Greek-mono”</td>
<td>greek-mono.mld</td>
</tr>
<tr>
<td>13.72</td>
<td>“Greek-polydemo”</td>
<td>greek-polydemo.mld</td>
</tr>
<tr>
<td>13.73</td>
<td>“Greek-polykatha”</td>
<td>greek-polykatha.mld</td>
</tr>
<tr>
<td>13.74</td>
<td>“Guarani”</td>
<td>guarani.mld</td>
</tr>
<tr>
<td>13.75</td>
<td>“Hangul1”</td>
<td>hangul1.ml[d</td>
</tr>
<tr>
<td>13.76</td>
<td>“Hangul2”</td>
<td>hangul2.ml[d</td>
</tr>
<tr>
<td>13.77</td>
<td>“Hangul3”</td>
<td>hangul3.ml[d</td>
</tr>
<tr>
<td>13.78</td>
<td>“Hangul4”</td>
<td>hangul4.ml[d</td>
</tr>
<tr>
<td>13.79</td>
<td>“Hangul-u8”</td>
<td>hangul-u8.ml[d</td>
</tr>
<tr>
<td>13.80</td>
<td>“Hanja1”</td>
<td>hanja1.mld.ml[d</td>
</tr>
<tr>
<td>13.81</td>
<td>“Hanja2”</td>
<td>hanja2.ml[d</td>
</tr>
<tr>
<td>13.82</td>
<td>“Hanja-u8”</td>
<td>hanja-u8.ml[d</td>
</tr>
<tr>
<td>13.83</td>
<td>“Hebrew”</td>
<td>hebrew.mld</td>
</tr>
<tr>
<td>13.84</td>
<td>“Hebrew2”</td>
<td>hebrew2.mld</td>
</tr>
<tr>
<td>13.85</td>
<td>“Hindi”</td>
<td>hindi.mld</td>
</tr>
<tr>
<td>13.86</td>
<td>“Hindi-modern”</td>
<td>hindi-modern.mld</td>
</tr>
<tr>
<td>13.87</td>
<td>“Hungarian”</td>
<td>hungarian.mld</td>
</tr>
<tr>
<td>13.88</td>
<td>“Icelandic”</td>
<td>icelandic.mld</td>
</tr>
<tr>
<td>13.89</td>
<td>“Indon”</td>
<td>indon.mld</td>
</tr>
<tr>
<td>13.90</td>
<td>“Indonesian”</td>
<td>indonesian.mld</td>
</tr>
<tr>
<td>13.91</td>
<td>“Interlingua”</td>
<td>interlingua.mld</td>
</tr>
<tr>
<td>13.92</td>
<td>“Irish”</td>
<td>irish.mld</td>
</tr>
<tr>
<td>13.93</td>
<td>“Italian”</td>
<td>italian.mld</td>
</tr>
<tr>
<td>13.94</td>
<td>“Italian2”</td>
<td>italian2.mld</td>
</tr>
<tr>
<td>13.95</td>
<td>“Japanese”</td>
<td>japanese.ml[d</td>
</tr>
<tr>
<td>13.96</td>
<td>“Japanese2”</td>
<td>japanese2.ml[d</td>
</tr>
<tr>
<td>13.97</td>
<td>“Japanese3”</td>
<td>japanese3.ml[d</td>
</tr>
<tr>
<td>13.98</td>
<td>“Japanese4”</td>
<td>japanese4.ml[d</td>
</tr>
<tr>
<td>13.99</td>
<td>“Japanese5”</td>
<td>japanese5.ml[d</td>
</tr>
<tr>
<td>13.100</td>
<td>“Japanese6”</td>
<td>japanese6.ml[d</td>
</tr>
<tr>
<td>13.101</td>
<td>“Kannada”</td>
<td>kannada.mld</td>
</tr>
<tr>
<td>13.102</td>
<td>“Khalkha”</td>
<td>khalkha.mld</td>
</tr>
<tr>
<td>13.103</td>
<td>“Latin”</td>
<td>latin.mld</td>
</tr>
<tr>
<td>13.104</td>
<td>“Latin2”</td>
<td>latin2.mld</td>
</tr>
<tr>
<td>13.105</td>
<td>“Latinc”</td>
<td>latinc.mld</td>
</tr>
<tr>
<td>13.106</td>
<td>“Latinc2”</td>
<td>latinc2.mld</td>
</tr>
<tr>
<td>13.107</td>
<td>“Latvian”</td>
<td>latvian.mld</td>
</tr>
<tr>
<td>13.108</td>
<td>“Latvian2”</td>
<td>latvian2.mld</td>
</tr>
<tr>
<td>13.109</td>
<td>“Letton”</td>
<td>letton.mld</td>
</tr>
<tr>
<td>13.110</td>
<td>“Letton2”</td>
<td>letton2.mld</td>
</tr>
<tr>
<td>13.111</td>
<td>“Lithuanian”</td>
<td>lithuanian.mld</td>
</tr>
<tr>
<td>13.112</td>
<td>“Lithuanian2”</td>
<td>lithuanian2.mld</td>
</tr>
<tr>
<td>13.113</td>
<td>“Lowersorbian”</td>
<td>lowersorbian.mld</td>
</tr>
<tr>
<td>13.114</td>
<td>“Lsorbian”</td>
<td>lsorbian.mld</td>
</tr>
<tr>
<td>13.115</td>
<td>“Magyar”</td>
<td>magyar.mld</td>
</tr>
<tr>
<td>13.116</td>
<td>“Magyar2”</td>
<td>magyar2.mld</td>
</tr>
<tr>
<td>13.117</td>
<td>“Magyar3”</td>
<td>magyar3.mld</td>
</tr>
<tr>
<td>13.118</td>
<td>“Malay”</td>
<td>malay.mld</td>
</tr>
<tr>
<td>13.119</td>
<td>“Malayalam-b”</td>
<td>malayalam-b.mld</td>
</tr>
<tr>
<td>Section</td>
<td>Language/Library</td>
<td>Page</td>
</tr>
<tr>
<td>-----------</td>
<td>--------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>13.120</td>
<td>“Malayalam-keli” language: malayalam-keli.mld</td>
<td>525</td>
</tr>
<tr>
<td>13.121</td>
<td>“Malayalam-keli2” language: malayalam-keli2.mld</td>
<td>525</td>
</tr>
<tr>
<td>13.122</td>
<td>“Malayalam-mr” language: malayalam-mr.mld</td>
<td>526</td>
</tr>
<tr>
<td>13.124</td>
<td>“Malayalam-rachana” language: malayalam-rachana.mld</td>
<td>527</td>
</tr>
<tr>
<td>13.125</td>
<td>“Malayalam-rachana2” language: malayalam-rachana2.mld</td>
<td>527</td>
</tr>
<tr>
<td>13.126</td>
<td>“Malayalam-rachana3” language: malayalam-rachana3.mld</td>
<td>528</td>
</tr>
<tr>
<td>13.127</td>
<td>“Manju” language: manju.mld</td>
<td>528</td>
</tr>
<tr>
<td>13.128</td>
<td>“Mexican” language: mexican.mld</td>
<td>528</td>
</tr>
<tr>
<td>13.129</td>
<td>“Meyalu” language: meyalu.mld</td>
<td>529</td>
</tr>
<tr>
<td>13.130</td>
<td>“Mongol” language: mongol.mld</td>
<td>529</td>
</tr>
<tr>
<td>13.131</td>
<td>“Mongolb” language: mongolb.mld</td>
<td>530</td>
</tr>
<tr>
<td>13.132</td>
<td>“Mongolian” language: mongolian.mld</td>
<td>531</td>
</tr>
<tr>
<td>13.133</td>
<td>“Naustrian” language: naustrian.mld</td>
<td>531</td>
</tr>
<tr>
<td>13.134</td>
<td>“Newzealand” language: newzealand.mld</td>
<td>531</td>
</tr>
<tr>
<td>13.135</td>
<td>“Ngerman” language: ngerman.mld</td>
<td>531</td>
</tr>
<tr>
<td>13.136</td>
<td>“Ngermanb” language: ngermanb.mld</td>
<td>532</td>
</tr>
<tr>
<td>13.137</td>
<td>“Ngermanb2” language: ngermanb2.mld</td>
<td>532</td>
</tr>
<tr>
<td>13.138</td>
<td>“Norsk” language: norsk.mld</td>
<td>533</td>
</tr>
<tr>
<td>13.139</td>
<td>“Norsk2” language: norsk2.mld</td>
<td>533</td>
</tr>
<tr>
<td>13.140</td>
<td>“Nynorsk” language: nynorsk.mld</td>
<td>534</td>
</tr>
<tr>
<td>13.141</td>
<td>“Nynorsk2” language: nynorsk2.mld</td>
<td>534</td>
</tr>
<tr>
<td>13.142</td>
<td>“Occitan” language: occitan.mld</td>
<td>535</td>
</tr>
<tr>
<td>13.143</td>
<td>“Occitan2” language: occitan2.mld</td>
<td>535</td>
</tr>
<tr>
<td>13.144</td>
<td>“Polish” language: polish.mld</td>
<td>536</td>
</tr>
<tr>
<td>13.145</td>
<td>“Polish2” language: polish2.mld</td>
<td>536</td>
</tr>
<tr>
<td>13.146</td>
<td>“Polski” language: Polski.mld</td>
<td>537</td>
</tr>
<tr>
<td>13.147</td>
<td>“Portuges” language: portuges.mld</td>
<td>537</td>
</tr>
<tr>
<td>13.148</td>
<td>“Portuguese” language: portuguese.mld</td>
<td>537</td>
</tr>
<tr>
<td>13.149</td>
<td>“Romanian” language: romanian.mld</td>
<td>538</td>
</tr>
<tr>
<td>13.150</td>
<td>“Romanian2” language: romanian2.mld</td>
<td>538</td>
</tr>
<tr>
<td>13.151</td>
<td>“Romanian3” language: romanian3.mld</td>
<td>539</td>
</tr>
<tr>
<td>13.152</td>
<td>“Russian” language: russian.mld</td>
<td>539</td>
</tr>
<tr>
<td>13.153</td>
<td>“Russian2m” language: russian2m.mld</td>
<td>540</td>
</tr>
<tr>
<td>13.154</td>
<td>“Russian2o” language: russian2o.mld</td>
<td>541</td>
</tr>
<tr>
<td>13.155</td>
<td>“Russianb” language: russianb.mld</td>
<td>541</td>
</tr>
<tr>
<td>13.156</td>
<td>“Russian” language: russian.mld</td>
<td>542</td>
</tr>
<tr>
<td>13.158</td>
<td>“Russian-cca1” language: russian-cca1.ml[d</td>
<td>o]</td>
</tr>
<tr>
<td>13.159</td>
<td>“Russian-lh” language: russian-lh.ml[d</td>
<td>o]</td>
</tr>
<tr>
<td>13.161</td>
<td>“Russian-lhcyrko” language: russian-lhcyrko.ml[d</td>
<td>o]</td>
</tr>
<tr>
<td>13.162</td>
<td>“Russian-lhyrw” language: russian-lhyrw.mld[d</td>
<td>o]</td>
</tr>
<tr>
<td>13.163</td>
<td>“Samin” language: samin.mld</td>
<td>545</td>
</tr>
<tr>
<td>13.164</td>
<td>“Scottish” language: scottish.mld</td>
<td>546</td>
</tr>
<tr>
<td>13.165</td>
<td>“Serbian” language: serbian.mld</td>
<td>546</td>
</tr>
<tr>
<td>13.166</td>
<td>“Serbian” language: serbian.mld</td>
<td>547</td>
</tr>
<tr>
<td>13.167</td>
<td>“Slovak” language: slovak.mld</td>
<td>547</td>
</tr>
<tr>
<td>13.168</td>
<td>“Slovene” language: slovene.mld</td>
<td>548</td>
</tr>
</tbody>
</table>
Table of contents

13.169 “Spanish” language: spanish.mld ................................. 548
13.170 “Spanish2” language: spanish2.mld ............................. 549
13.171 “Spanish3” language: spanish3.mld ............................. 549
13.172 “Spanish4” language: spanish4.mld ............................. 550
13.173 “Swahili” language: swahili.mld ................................. 550
13.174 “Swedish” language: swedish.mld ............................... 551
13.175 “Swedish2” language: swedish2.mld ............................ 551
13.176 “Thai” language: thai.ml[d,o] ..................................... 552
13.177 “Turkish” language: turkish.mld ................................. 552
13.178 “Uighur” language: uighur.mld .................................. 552
13.179 “Uighur2” language: uighur2.mld ............................... 553
13.180 “Uighur3” language: uighur3.mld ............................... 553
13.181 “UKenglish” language: UKenglish.mld ......................... 553
13.182 “Ukraineb” language: ukraineb.mld .................. 553
13.183 “Ukrainian” language: ukrainian.mld ............................ 554
13.184 “Uppersorbian” language: uppersorbian.mld .................. 554
13.185 “USenglish” language: USenglish.mld ........................... 555
13.186 “Usorbian” language: usorbian.mld ............................. 555
13.188 “Vietnamese” language: vietnamese.mld ...................... 556
13.189 “Welsh” language: welsh.mld ................................... 556
13.190 “Xalx” language: xalx.mld ...................................... 557
13.191 “Xalx2” language: xalx2.mld .................................... 557
13.192 “Xalx3” language: xalx3.mld .................................... 558

III Complements 559

Bibliography 561

Changes history 597

Acknowledgments 634
List of figures

1.1 Float barriers ......................................................... 31
1.2 Layout of a ToC (LoF, LoT) entry .................................. 45
2.1 Three compilations for minitoc ..................................... 59
## List of tables

<table>
<thead>
<tr>
<th>Section</th>
<th>Table Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Commands for a minitoc</td>
<td>28</td>
</tr>
<tr>
<td>1.2</td>
<td>Commands for a secttoc</td>
<td>30</td>
</tr>
<tr>
<td>1.3</td>
<td>Commands for a parttoc</td>
<td>32</td>
</tr>
<tr>
<td>1.4</td>
<td>Adding an entry in the ToC for a starred part, chapter, or section</td>
<td>34</td>
</tr>
<tr>
<td>1.5</td>
<td>Fonts and titles for the mini-table commands</td>
<td>36</td>
</tr>
<tr>
<td>1.6</td>
<td>Fonts for the mini-table entries</td>
<td>37</td>
</tr>
<tr>
<td>1.7</td>
<td>Available languages</td>
<td>38</td>
</tr>
<tr>
<td>1.8</td>
<td>Horizontal rules</td>
<td>46</td>
</tr>
<tr>
<td>1.9</td>
<td>Page numbers</td>
<td>46</td>
</tr>
<tr>
<td>1.10</td>
<td>Features for mini-tables</td>
<td>48</td>
</tr>
<tr>
<td>1.11</td>
<td>Extensions of the auxiliary files</td>
<td>55</td>
</tr>
<tr>
<td>2.1</td>
<td>Kernings before minitable bottom rules</td>
<td>78</td>
</tr>
<tr>
<td>3.1</td>
<td>Package options</td>
<td>80</td>
</tr>
<tr>
<td>3.2</td>
<td>General commands</td>
<td>81</td>
</tr>
<tr>
<td>3.3</td>
<td>Part level commands</td>
<td>82</td>
</tr>
<tr>
<td>3.4</td>
<td>Chapter level commands</td>
<td>83</td>
</tr>
<tr>
<td>3.5</td>
<td>Section level commands</td>
<td>84</td>
</tr>
<tr>
<td>3.6</td>
<td>Commands for horizontal rules</td>
<td>85</td>
</tr>
<tr>
<td>3.7</td>
<td>Commands for page numbers</td>
<td>85</td>
</tr>
<tr>
<td>3.8</td>
<td>Commands for mini-tables features</td>
<td>86</td>
</tr>
<tr>
<td>3.9</td>
<td>Preparation and insertion commands</td>
<td>87</td>
</tr>
<tr>
<td>3.10</td>
<td>Adjustment commands</td>
<td>87</td>
</tr>
<tr>
<td>3.11</td>
<td>Classes and packages needing some precautions with minitoc</td>
<td>88</td>
</tr>
<tr>
<td>3.12</td>
<td>Checking if inside a minitable</td>
<td>89</td>
</tr>
<tr>
<td>3.13</td>
<td>Commands for polymorphic entries</td>
<td>89</td>
</tr>
<tr>
<td>3.14</td>
<td>Obsolete commands</td>
<td>89</td>
</tr>
<tr>
<td>5.1</td>
<td>Message identifiers</td>
<td>152</td>
</tr>
<tr>
<td>6.1</td>
<td>Category codes</td>
<td>211</td>
</tr>
<tr>
<td>6.2</td>
<td>Encoding schemes implemented in CJK</td>
<td>212</td>
</tr>
<tr>
<td>6.3</td>
<td>Standard document classes</td>
<td>213</td>
</tr>
<tr>
<td>6.4</td>
<td>Depths for sectionning commands</td>
<td>214</td>
</tr>
<tr>
<td>6.5</td>
<td>Various encodings</td>
<td>216</td>
</tr>
<tr>
<td>6.6</td>
<td>Most common font encodings</td>
<td>230</td>
</tr>
<tr>
<td>6.7</td>
<td>Most common font families</td>
<td>230</td>
</tr>
</tbody>
</table>
## List of tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.8</td>
<td>Most common font series</td>
<td>230</td>
</tr>
<tr>
<td>6.9</td>
<td>Most common font shapes</td>
<td>231</td>
</tr>
<tr>
<td>6.10</td>
<td>Most common font widths</td>
<td>231</td>
</tr>
<tr>
<td>6.11</td>
<td>The five font parameters of some fonts</td>
<td>231</td>
</tr>
<tr>
<td>6.12</td>
<td>Author commands for fonts</td>
<td>232</td>
</tr>
<tr>
<td>6.13</td>
<td>Some systems derived from \TeX{} and \LaTeX{}</td>
<td>233</td>
</tr>
<tr>
<td>7.1</td>
<td>List of files (minitoc.1), first part</td>
<td>243</td>
</tr>
<tr>
<td>7.2</td>
<td>List of files (minitoc.1), second part</td>
<td>244</td>
</tr>
<tr>
<td>7.3</td>
<td>List of the graphic files</td>
<td>244</td>
</tr>
<tr>
<td>7.4</td>
<td>List of the flag files</td>
<td>244</td>
</tr>
<tr>
<td>7.5</td>
<td>A TDS-compliant hierarchy for the minitoc files</td>
<td>247</td>
</tr>
<tr>
<td>9.1</td>
<td>Trick to detect the limitation to short extensions</td>
<td>286</td>
</tr>
</tbody>
</table>
About this document

This document is rather thick, but please, be not afraid: you do not need to read every page.

• The most useful chapters are in the first part (“User’s Manual”, page 23):
  – the chapter “The minitoc package”, page 24, describes the essential commands to use the package;
  – the chapter “Frequently Asked Questions”, page 56, may help you to solve some specific problems;
  – the chapter “Memento”, page 80, is a set of tables to be used as a remainder of the commands of this package;
  – the chapter “Examples of documents”, page 90, gives the code of some documents showing the basic usage of the minitoc package and some interesting situations;
  – the chapter “Messages”, page 151, is certainly boring, but it should be searched if you get some warning or error messages from the minitoc package, because it explains them and also the informative messages (table 5.1 on page 152 will help you to find the meaning of a message);
  – the chapter “Jargon”, page 205, attempts to explain most of the technical terms used here;
  – the chapter “Installation”, page 242, describes all the files included in the distribution of the package;
  – the chapter “Postface”, page 248, gives an abbreviated history of the package.

• The second part, “Implementation”, page 261, is much more technical; you can read it if you are interested in the details of the coding of the package. The chapter “Language definition (.mld) and object (.mlo) files”, page 467, may be useful if you are interested by some language. This chapter contains many maps and illustrations.

• The third part, “Complements”, page 560, contains a bibliography, a detailed history of the package, a list of acknowledgments, and an index.

For this document, I have used:

• a short table of contents (summary), with the \shorttoc command from my shorttoc package [155], displaying only parts and chapters;
About this document

- a main table of contents (\tableofcontents), with a maximum depth (6);
- a main list of figures (\listoffigures) and a main list of tables (\listoftables);
- for each part, a table of contents displaying only the chapters (\parttoc with parttocdepth equal to 1);
- for each chapter, a complete table of contents (\minitoc with minitocdepth equal to 6);
- for each chapter, a list of figures (\minilof) and a list of tables (\minilot) when useful;
- customized parameters for the layout of the mini-tables; as the PDF version of the documentation uses hyperlinks (with the help of the hyperref package [390]), these mini-tables should help you to navigate within the document;
- some hyperlinks, placed in the right margin, contain a message identifier; the link points to the description of the message in the “Messages” chapter; try this one: \hhref{\textfile{I0001}};
- some flags, with hyperlinks to articles (mainly in Wikipedia) about countries or languages;
- the calc package [441] to make some computations with comfort;
- the booktabs package [165] to format the tables;
- the doc [327] and docstrip [287] packages to document the code;
- many other packages to improve the presentation of the documentation.
Part I

User’s Manual
# Contents of the First Part

<table>
<thead>
<tr>
<th></th>
<th>The minitoc package</th>
<th>24</th>
<th>5 Messages</th>
<th>151</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Frequently Asked Questions</td>
<td>56</td>
<td>6 Jargon</td>
<td>205</td>
</tr>
<tr>
<td>3</td>
<td>Memento</td>
<td>80</td>
<td>7 Installation</td>
<td>242</td>
</tr>
<tr>
<td>4</td>
<td>Examples of documents</td>
<td>90</td>
<td>8 Postface</td>
<td>248</td>
</tr>
</tbody>
</table>
Chapter 1

The minitoc package

Contents

1.1 Introduction ............................................. 25
  1.1.1 Important restrictions ............................... 26
  1.1.2 Version ............................................ 26

1.2 License ................................................ 26

1.3 Using the minitoc package ............................... 27
  1.3.1 Loading the package and creating the mini-tables .... 27
  1.3.2 Preparing the mini-tables .......................... 27
  1.3.3 Placing the mini-tables ............................. 29
  1.3.4 Starred chapters, parts and sections ............... 33

1.4 Typesetting of the mini-tables ......................... 34
  1.4.1 Chapter-level mini-tables .......................... 35
  1.4.2 Titles for chapter-level mini-tables ............... 35
  1.4.3 Part-level mini-tables ............................. 35
  1.4.4 Titles for part-level mini-tables .................. 37
  1.4.5 Section-level mini-tables .......................... 39
  1.4.6 Titles for section-level mini-tables ............... 39
  1.4.7 Position of the titles ............................. 39
    1.4.7.1 For mini-tables at the part level ............. 39
    1.4.7.2 For mini-tables at the chapter level .......... 40
    1.4.7.3 For mini-tables at the section level .......... 40
    1.4.7.4 Summary of the positionning of titles ......... 40
  1.4.8 Line spacing in the mini-tables .................... 41
  1.4.9 Simplified commands for fonts ..................... 41
  1.4.10 Simplified command for mini-table titles .......... 42
  1.4.11 Simplified command for mini-table depths ........ 42
  1.4.12 Simplified command for mini-table offsets ........ 42
  1.4.13 Polymorphic entries in the mini-tables .......... 43
  1.4.14 Languages for the titles ........................ 44
  1.4.15 Altering the layout of the mini-tables ........... 44

1.5 Special Features ....................................... 46
  1.5.1 Horizontal Rules .................................. 46
  1.5.2 Page Numbers, Leaders ............................ 47
  1.5.3 Features for parttoc's and other mini-tables ....... 47
The minitoc package, initially written by Nigel Ward and Dan Jurafsky, has been almost completely redesigned by Jean-Pierre F. Ducrèbert (ONERA/Centre de Toulouse). A summary of the evolution of this package is given in the chapter 8 on page 248. This package creates a mini-table of contents (a “minitoc”) at the beginning of each chapter of a document. It is also possible to have a mini-list of figures (a “minilof”) and a mini-list of tables (a “minilot”). The document class should of course define chapters (classes like book or report [282]) or sections (classes like article 2) [282]. Thus, this package should not be used with document classes without standard sectioning commands (like letter). When the document class defines a “part” sectioning level (i.e., classes like book, report and article), you can create a “partial” table of contents (a “parttoc”) at the beginning of each part of a document. It is also possible to have a partial list of figures (a “partlof”) and a partial list of tables (a “partlot”). When the document class has no \chapter command but has a \section command, you may use

---

1 The minitoc package introduces its own jargon, explained in this document. It should not be too difficult, however, to learn and use; it will be used here, of course.

2 As the standard proc class [281], and the ltxdoc [116] and ltnews [248] classes, load the standard article class, these classes will be just considered as variants of the article class.
section level tables of contents ("secttocs") at the beginning of each section; and you can also have section level lists of figures ("sectlofs") or of tables ("sectlots").

All these tables ("minitocs", "partlots", "sectlofs", etc.) are collectively referenced as "mini-tables" (or sometimes "mini-lists").

1.1.1 Important restrictions

Note: you cannot use chapter level and section level mini-tables in the same document. This restriction is intended to avoid documents with full of local tables of contents, lists of figures and tables at every sectioning level.

Note: the commands relative to the part level are defined only if the document class defines \part. The commands relative to the section level are defined only if the document class defines \section but does not define \chapter.

1.1.2 Version

The current version of this package is #61. You will find a resumed history of the package in the “Postface” chapter (chapter 8 on page 248) and a more detailed history in “Changes History”, page 597.

1.2 License

This package must be distributed and/or may be modified under the conditions of the \LaTeX Project Public License, either version 1.3 of this license or (as convenient) 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 2003/12/01 or later.

But please don’t bother me about hacked versions; they will not be supported. However, suggestions for corrections and reasoned improvements are welcome.
1.3 Using the minitoc package

1.3.1 Loading the package and creating the mini-tables

\usepackage[minitoc\chapter]{minitoc}

To use the minitoc package, you must insert a command:

\usepackage[...options...]{minitoc}

in the preamble of the document. The mini-table of contents will be in the chapter, after the \chapter command, at the point of the \minitoc command. The \minitoc command may occur almost anywhere inside a chapter.

Of course, it is better to put it at the beginning of the chapter, eventually after some introductory material. But you can also decide to put it at the end of the chapter. You should use the same conventions in all chapters. If you want to add the mini-table of contents for a chapter, you must use the sequence given in table 1.1 on the next page.

For each mini-table of contents, an auxiliary file will be created with a name of the form document.mtc\langle N\rangle, where \langle N\rangle is the absolute chapter number. “Absolute” means that this number is unique, and always increasing from the first chapter. The suffix is .mlf\langle N\rangle for mini-lists of figures and is .mlt\langle N\rangle for mini-lists of tables. (If under MS-DOS or any operating system with short extensions to filenames, see section 1.9 on page 54 and section 2.5 on page 58). There are similar commands for mini-tables at the part or section level, depending on the document class.

1.3.2 Preparing the mini-tables

\dominitoc, \dominilof, and \dominilot (for mini-tables at the chapter level), take respectively the document.toc, document.lof, and document.lot files, and cut slices from them to create the document.mtc\langle N\rangle, document.mlf\langle N\rangle, and document.mlt\langle N\rangle files.

---

3 This command must be placed after any modification done on the sectioning commands; if you modify some sectioning commands after loading the minitoc package, this one might not work properly.

4 “Almost anywhere” means “in a normal place”, like between two paragraphs of normal text, or in a (wide enough) minipage, but not in a too strange position (like a marginal note or a footnote). Even a multicolumn or a floating environment can be used, but with care. But note that a minitoc can be rather long, if the chapter is complex and if you are asking for details with a high value for \minitocdepth. As an example, I once used a \afterpage command (afterpage package [115]) to place the long minilof of chapter 13 on page 467 (so the minilof was forced to begin at the top of the next page).

5 The concept of an “absolute” counter for the mini-tables has solved some obscure problems, and also made obsolete some commands, like \firstpartis, \firstchapteris, and \firstsectionis.

6 The code of these \do... commands is directly derived from that of the \xr package [114], by David P. Carlisle, with his permission.
The commands \dosecttoc, \dosectlof, and \dosectlot (for mini-tables at the section level) and \doparttoc, \dopartlof, and \dopartlot (for mini-tables at the part level) are analog.

The \mtcprepare command invokes (and replaces) all these preparation commands when they are available with the document class and if the adequate contents file exists. This command accepts also an optional argument to set the default position of the title for all the mini-tables.

All the preparation commands are ignored if the \nofiles command is invoked in the preamble, to avoid to overwrite the mini-table auxiliary files.

To obtain a satisfactory result (i.e., non empty), please note that all these commands must imperatively be put before any command analog to the \tableofcontents, \listoffigures, and \listoftables commands, or their \fake... siblings.
It is also strongly recommended to put these commands *before* any sectionning command producing an entry in the table of contents (for the \do...toc commands), and *before* any \caption-like command producing an entry in the list of figures (for the \do...lof commands) or in the list of tables (for the \do...lot commands); else disorder in the mini-tables might result.

### 1.3.3 Placing the mini-tables

\mtcskip \mtcskipamount \bigskipamount

The \mtcskip command may be used to add a vertical skip between two mini-tables. Its height is \mtcskipamount (equal to \bigskipamount by default). \mtcskip eliminates any immediate previous vertical skip, to not accumulate vertical space when a mini-table is empty and skipped by the checkfiles option.

\secttoc \section

The section-level table of contents will be in the section, after the \section command, at the point of the \secttoc command. The \secttoc command may occur *almost anywhere* inside a section. It is often better to put it at the beginning of the section, or after some short introductory material. You should use the same conventions in all sections. If you want to add a section-level table of contents for a section, you must use the sequence given in Table 1.2 on the following page.

For each section-level table of contents, an auxiliary file will be created with a name of the form *document.stc(N)*, where *(N)* is the absolute section number. The suffix is *.sft(N)* for section-level lists of figures and is *.slt(N)* for section-level lists of tables. (If under MS-DOS or any operating system with short extensions to filenames, see section 1.9 on page 54 and section 2.5 on page 58).

\usepackage \FloatBarrier

As floats (figures and tables) could drift\(^7\) somewhere outside the printing area of the text of the section, the sectlofs and sectlots can be rather strange. In order to have a better behaviour of these mini-tables, it may be useful to add the insection option in the \usepackage command:

\usepackage[insection]{minitoc}

if you want more consistent sectlofs and sectlots. The insection option loads the placeins package [15] with its verbose and section options. Sometimes, it might be necessary to use the \FloatBarrier command of this package to correctly place the figure or table and have a correct mini-table. The options above or below options should not be used, because they allow floats to drift above or below a \FloatBarrier (or a section limit): the barrier\(^7\)

---

\(^7\) A float is like a ship in harbor. There is a place in the text which is the anchor location. The figure or “ship” can float around to various places relative to the anchor, but always downstream or downwind. A float with bad placement parameters is like a ship that slips its anchor and eventually crashes on the rocks at the end of a chapter.

Donald Arseneau
becomes “porous” upwards\(^8\) (↑) or downwards (↓), or both (↕). The section option makes a more “watertight” barrier (≡). This is illustrated by the figure 1.1 on the next page.

The placeins package, by Donald Arseneau, is available on CTAN archives; note that the file placeins.sty contains its own documentation, with a copy in placeins.txt. You need a version whose date is at least 2005/04/18.

Since version #45, this option also loads the flafter package (described in [288] and [330, page 286]) to force a float to appear after its reference. The above and below options of the placeins package are no more used, because they allowed the floats to move out of the section.

In all cases, it is strongly recommended to verify the position of the floats and, if necessary, to look at the messages of the placeins package in the document.log file. The placement of floats is a very complex problem, so some manual intervention may be necessary, like the use of the float package [302] or, better, of the floatrow package [285].

\(^8\) But a float can not drift upwards beyond the top of the current page.
With the *section* option

<table>
<thead>
<tr>
<th>Command</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>\section</td>
<td>\section Forbidden Area \section Allowed Area</td>
</tr>
<tr>
<td>\section</td>
<td>\section Floats inserted here \section Allowed Area</td>
</tr>
</tbody>
</table>

With the above option

<table>
<thead>
<tr>
<th>Command</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>\section</td>
<td>\section Allowed Area</td>
</tr>
<tr>
<td>\section</td>
<td>\section Floats inserted here \section Allowed Area</td>
</tr>
<tr>
<td>\section</td>
<td>\section Forbidden Area \section Allowed Area</td>
</tr>
</tbody>
</table>

With the below option

<table>
<thead>
<tr>
<th>Command</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>\section</td>
<td>\section Allowed Area</td>
</tr>
<tr>
<td>\section</td>
<td>\section Floats inserted here \section Allowed Area</td>
</tr>
<tr>
<td>\section</td>
<td>\section Allowed Area</td>
</tr>
</tbody>
</table>

With the above and below options

<table>
<thead>
<tr>
<th>Command</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>\section</td>
<td>\section Allowed Area</td>
</tr>
<tr>
<td>\section</td>
<td>\section Floats inserted here \section Allowed Area</td>
</tr>
<tr>
<td>\section</td>
<td>\section Allowed Area</td>
</tr>
</tbody>
</table>

Figure 1.1: Float barriers
If you want to add the partial table of contents for a part, you must use the sequence given in Table 1.3. For each partial table of contents, an auxiliary file will be created with a name of the form `document.ptc(N)`, where `(N)` is the absolute part number. The suffix is `.plf(N)` for partial lists of figures and is `.plt(N)` for partial lists of tables. (If under MS-DOS or any operating system with short extensions to filenames, see section 1.9 on page 54 and section 2.5 on page 58).

**Note:** the user is responsible of asking or not asking a mini-table (mini-toc, -lof or -lot) for some chapter. Asking a minilof for a chapter without any figure would result in an empty and ugly mini-list of figures (i.e., the title and two horizontal rules). He is also responsible of requiring or not requiring a partial toc (lof or lot) for some part. Asking a partlof for a part without any figure would result in an empty and ugly part list of figures (i.e., the title alone on a page). Analogous remarks apply to section-level mini-tables (secttoc, sectlof, and sectlot) and to the part-level mini-tables (parttoc, partlof, and partlot).

But since version #35, empty mini-tables are just ignored and this problem should disappear in normal circumstances. Nevertheless, it is recommended to put no \minitoc command

---

**Table 1.3: Commands for a parttoc**

```latex
\documentclass{book}
\usepackage{minitoc}
\setlength{\ptcindent}{0pt} \default \mtcsetoffset{parttoc}{0pt} \default
\setcounter{parttocdepth}{2} \default \renewcommand{\ptcfont}{\normalsize\rmfamily\upshape\mdseries} \default
\renewcommand{\ptcoffset}{0pt} \default \mtcsetoffset{parttoc}{0pt} \default
\mtcsetdepth{parttoc}{2} \default \mtcsetfont{parttoc}{\normalsize\rmfamily\upshape\mdseries} \default
\mtcsetfont{parttoc}{chapter}{\normalsize\rmfamily\upshape\bfseries} \default
\mtcsetfont{parttoc}{section}{\normalsize\rmfamily\upshape\mdseries} \default
\begin{document}
\doparttoc \dopartlof \dopartlot
\tableofcontents \or \faketableofcontents
\listoffigures \or \fakelistoffigures
\listoftables \or \fakelistoftables
\part...
\parttoc \if you want one
\partlof \if you want one
\partlot \if you want one
\part...
```
in a chapter without sections and no \minilof or \minilot command in a chapter without figures or tables. The checkfiles (see section 1.3.3 on page 29) package option (default) skips empty mini-tables (with a note in the document.log file); the nocheckfiles package option restores the old behaviour (empty mini-tables are displayed).

By default, the mini-tables and partial tables of contents contain only references higher and to sections and subsections. The counters parttocdepth, minitocdepth and secttocdepth, similar to tocdepth, allow the user to modify this behaviour. Mini or partial lists of figures or tables are not affected by the value of these counters, but if there are depth counters for these lists (lofdepth and lotdepth), as done by the subfigure and subfig packages [130, 132] from Steven Douglas Cochran, new depth counters are created if necessary, with obvious names like partlofdepth, partlotdepth, minilofdepth, minilotdepth, sectlofdepth, and sectlotdepth.

1.3.4 Starred chapters, parts and sections

\addcontentsline \addstarredpart \addstarredchapter \addstarredsection

NOTE: if using \chapter* and a
\addcontentsline{toc}{chapter}{...}

command to add something in the table of contents, the numbering of the minitoc auxiliary files would be altered. To avoid that problem, a first method is to say:

\addstarredpart{...}
\addstarredchapter{...}
\addstarredsection{...}

These commands apply only for the level of a part-, mini- or sect-toc; for lower levels, the usual command is sufficient:

\addcontentsline \addcontentsline{toc}{section}{...}

So, to add a section-level entry in the global toc and in the minitoc of a starred chapter:

\chapter*{Title of chapter}
\addstarredchapter{Title of chapter}
\minitoc
\section*{First section}
\addcontentsline{toc}{section}{First section}
\section*{Second section}
\addcontentsline{toc}{section}{Second section}
There is sometimes a problem with mini-tables when you use `\chapter*` (or `\section*`): the minitocs appear in the wrong chapter. You can add a `\adjustmtc` (or `\adjuststc` or `\adjustptc`) command at the end of the starred chapter (or section or part) to increment the corresponding counter. Do not use commands like `\stepcounter{mtc}` or `\addtocounter{mtc}{...}` (which should work, but it is cheating), because the mtcoff package (see section 1.11 on page 55) knows what to do about `\adjustmtc` (and others), but can do nothing about `\stepcounter` or `\addtocounter`, as they are a standard basic commands of \LaTeX, not minitoc specific commands. Syntax:

\begin{verbatim}
\adjustptc[n] \adjustmtc[n] \adjuststc[n]
\end{verbatim}

where \( n \) is the increment (default: 1).

There are similar commands to `decrement` or `increment` by 1 these counters: \begin{verbatim}
\decrementptc, \decrementmtc, \decrementsstc, \incrementptc, \incrementmtc, \incrementstc
\end{verbatim} the same remarks as above apply. These commands have no argument.

But a more clever way to solve this problem would be using commands similar to:

\begin{verbatim}
\mtcaddchapter[title]
\end{verbatim}

This command adds an entry in the table of contents (and adjusts the counter, because it calls `\adjustmtc`). The table 1.4 summarizes these commands, that you put after `\chapter*`, etc.

If the optional argument is omitted or empty or blank, no entry will be visible in the table of contents nor in the minitocs. If the optional argument is something invisible (like ~, `\space` or `\quad`), the result will be strange but still logically correct. See also section 2.30 on page 72 for the problems with `\mtcaddpart`.

\begin{table}[h]
\centering
\begin{tabular}{ll}
Level & With title \\
partment & `\mtcaddpart[title]` \\
chapter & `\mtcaddchapter[title]` \\
section & `\mtcaddsection[title]` \\
\end{tabular}
\caption{Adding an entry in the ToC for a starred part, chapter, or section}
\end{table}

1.4 Typesetting of the mini-tables

The mini-tables are typeset in a verse-like environment, and can be split over several pages.


1.4.1 Chapter-level mini-tables

The mini-table of contents is typeset in the \mtcfont font, which is \small\rmfamily by default. In fact, the font \mtcfont is selected at the beginning of a minitoc, minilof or minilot. More selective choices are made with the following fonts. Section entries are typeset in the \mtcSfont font, which is \small\bfseries by default.

For subsections, subsubsections, paragraphs and subparagraphs, the commands \mtcSSfont, \mtcSSSfont, \mtcPfont and \mtcSPfont are available (by default, \small\rmfamily) to enable the use of various fonts. Mini lists of figures and tables are typeset in the fonts \mlffont and \mltfont, which are \small\rmfamily by default. There are also \mlfSfont and \mltSfont for sub-figures and sub-tables entries. See tables 1.5 to 1.6 on pages 36–37.

Note that the default choice of fonts is certainly not perfect and hence it is not definitive. A symptom of this imperfection is the presence of poor alignments in the mini-tables, if bold and non-bold fonts are mixed (the true length of 1em is not the same for the fonts). This can often be adjusted by changing some fonts.

1.4.2 Titles for chapter-level mini-tables

Titles are typeset in the \mtifont (\large\bfseries by default) font and the text strings of the titles are defined by \mtctitle, \mlftitle and \mltitl, which are the strings “Contents”, “Figures” and “Tables” by default. These title commands should be redefined by \renewcommand or \mtcsettitle for languages other than english.

The language definition files like french.mld and english.mld (the suffix .mld means “minitoc language definition (file)”)(and many others, see the list in table 1.7 on page 38 and section 1.4.14 on page 44) are available. You can easily prepare a similar file for a preferred language (see section 2.26 on page 70). You can change the language of these titles by using the \mtcselectlanguage\{language\} macro.

1.4.3 Part-level mini-tables

The partial table of contents is typeset in the \ptcfont font, which is defined as \normalsize\rmfamily by default. In fact, the font \ptcfont is selected at the beginning of a parttoc, partlof or partlot. More selective choices are made with the following fonts. Chapter entries are typeset in the \ptcCfont font, which is
Table 1.5: Fonts and titles for the mini-table commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Font default setting</th>
<th>Title string default setting</th>
<th>Title font default setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>For the <code>\part</code>... commands:</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| `\parttoc` | `\ptcfont` 
`\normalsize\rmfamily`  
`\small\rmfamily` | `\ptctitle` Table of Contents† | `\ptifont` `\LARGE\bfseries` `\LARGE\bfseries` `\LARGE\bfseries` |
| `\partlof` | `\plffont`  
`\normalsize\rmfamily`  
`\small\rmfamily` | `\plftitle` List of Figures† | `\ptifont` `\LARGE\bfseries` `\LARGE\bfseries` |
| `\partlot` | `\pltfont`  
`\normalsize\rmfamily`  
`\small\rmfamily` | `\pltttitle` List of Tables† | `\ptifont` `\LARGE\bfseries` `\LARGE\bfseries` |
| | | | |
| For the `\mini`... commands: | | | |
| `\minitoc` | `\mtcfont`  
`\small\rmfamily` | `\mtctitle` Contents† | `\mtifont` `\large\bfseries` `\large\bfseries` |
| `\minilof` | `\mlffont`  
`\small\rmfamily` | `\mlftitle` Figures† | `\mtifont` `\large\bfseries` `\large\bfseries` |
| `\minilot` | `\mltfont`  
`\small\rmfamily` | `\mlttitle` Tables† | `\mtifont` `\large\bfseries` `\large\bfseries` |
| | | | |
| For the `\sect`... commands: | | | |
| `\secttoc` | `\stcfont`  
`\small\rmfamily` | `\stctitle` Contents† | `\stifont` `\large\bfseries` `\large\bfseries` |
| `\sectlof` | `\slffont`  
`\small\rmfamily` | `\slftitle` Figures† | `\stifont` `\large\bfseries` `\large\bfseries` |
| `\sectlot` | `\sltfont`  
`\small\rmfamily` | `\sltttitle` Tables† | `\stifont` `\large\bfseries` `\large\bfseries` |

†for document classes with `\chapter` level (e.g., `book`, `report`).
‡for document classes with no `\chapter` level (e.g., `article`).

default for english; changed by the language definition files or `\renewcommand`.

All these fonts use \rmfamily, \upshape, and \mdseries by default.

`\normalsize\bfseries` by default. Section entries are typeset in the \ptcSSfont font, which is \normalsize\rmfamily by default.

\ptcSSfont For subsections, subsubsections, paragraphs and subparagraphs, the commands \ptcSSfont, \ptcPfont and \ptcSPfont are available (by default, \normalsize\rmfamily) if you want to use various fonts.
Table 1.6: Fonts for the mini-table entries

<table>
<thead>
<tr>
<th>Level</th>
<th>Font</th>
<th>Default setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>For the <code>parttoc</code> entries:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chapter<code>'</code></td>
<td><code>ptcCfont</code></td>
<td><code>\normalsize\bfseries</code>'</td>
</tr>
<tr>
<td>Section</td>
<td><code>ptcSfont</code></td>
<td><code>\normalsize\rmfamily</code>'</td>
</tr>
<tr>
<td></td>
<td></td>
<td><code>\small\bfseries</code></td>
</tr>
<tr>
<td>Subsection</td>
<td><code>ptcSSfont</code></td>
<td>(like <code>ptcfont</code>)</td>
</tr>
<tr>
<td>Subsubsection</td>
<td><code>ptcSSSfont</code></td>
<td>(like <code>ptcfont</code>)</td>
</tr>
<tr>
<td>Paragraph</td>
<td><code>ptcPfont</code></td>
<td>(like <code>ptcfont</code>)</td>
</tr>
<tr>
<td>Subparagraph</td>
<td><code>ptcSPfont</code></td>
<td>(like <code>ptcfont</code>)</td>
</tr>
<tr>
<td>For the <code>minitoc</code> entries:<code>'</code></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Section</td>
<td><code>mtcSfont</code></td>
<td><code>\small\bfseries</code></td>
</tr>
<tr>
<td>Subsection</td>
<td><code>mtcSSfont</code></td>
<td>(like <code>mtcfont</code>)</td>
</tr>
<tr>
<td>Subsubsection</td>
<td><code>mtcSSSfont</code></td>
<td>(like <code>mtcfont</code>)</td>
</tr>
<tr>
<td>Paragraph</td>
<td><code>mtcPfont</code></td>
<td>(like <code>mtcfont</code>)</td>
</tr>
<tr>
<td>Subparagraph</td>
<td><code>mtcSPfont</code></td>
<td>(like <code>mtcfont</code>)</td>
</tr>
<tr>
<td>For the <code>secttoc</code> entries:<code>''</code></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Subsection</td>
<td><code>stcSSfont</code></td>
<td><code>\normalsize\bfseries</code></td>
</tr>
<tr>
<td>Subsubsection</td>
<td><code>stcSSSfont</code></td>
<td>(like <code>stcfont</code>)</td>
</tr>
<tr>
<td>Paragraph</td>
<td><code>stcPfont</code></td>
<td>(like <code>stcfont</code>)</td>
</tr>
<tr>
<td>Subparagraph</td>
<td><code>stcSPfont</code></td>
<td>(like <code>stcfont</code>)</td>
</tr>
</tbody>
</table>

' for document classes with `\chapter` level (e.g., book, report).
`` for document classes with no `\chapter` level (e.g., article).

`plffont`, `pltfont`, `plfSfont`, `pltSfont`
Partial lists of figures and tables are typeset in the fonts `plffont` and `pltfont`, which are `\normalsize\rmfamily` by default. There are also `plfSfont` and `pltSfont` for sub-figures and sub-tables entries.

1.4.4 Titles for part-level mini-tables

`ptifont`, `ptctitle`, `plftitle`, `plltitle`, `mtcsettitle`, `mtcselectlanguage`
Titles are typeset in the `ptifont` (`\LARGE\bfseries` by default) font and the text strings of the titles are defined by `ptctitle`, `plftitle` and `plltitle`, which are the strings “Table of Contents”, “List of Figures” and “List of Tables” by default. These title commands should be redefined by `\renewcommand` or `\mtcsettitle` for languages other than English.

The language definition files like `french.mld` and `english.mld` (and many others; for a complete list, see table 1.7 on the next page) are available. Read also section 1.4.14 on page 44. You can easily prepare a similar file for a preferred language (see section 2.26 on page 70). You can change the language of these titles by using the `\mtcselectlanguage{language}` macro.
Table 1.7: Available languages

- afrikaans (afrikaans)
- albanian
- arab (arabic)\(^a\)
- arab2\(^a\)
- arabi\(^a\)
- armenian\(^c\)
- bahasa (bahasa, indon, indonesian)\(^a\)
- bahasa (bahasa, indon, indonesian)\(^a\)
- bangla (bengali)\(^c\)
- bahasam (malay, ethiopia (ethiopian)
- estonian
- esperanto (esperanto)
- english2
- english1
- dutch
- english1
- english2
- esperanto (esperanto)
- espaniol (espaniol)
- ethiopia (ethiopian)
- estonian
- esperanto (esperanto)
- english2
- english1
- dutch
- english1
- english2
- esperanto (esperanto)
- espaniol (espaniol)
- ethiopia (ethiopian)
- estonian
- esperanto (esperanto)
- english2
- english1
- dutch
- english1
- english2
- esperanto (esperanto)
- espaniol (espaniol)
- ethiopia (ethiopian)
- estonian
- esperanto (esperanto)
- english2
- english1
- dutch
- english1
- english2
- esperanto (esperanto)
- espaniol (espaniol)
- ethiopia (ethiopian)
- estonian
- esperanto (esperanto)
- english2
- english1
- dutch
- english1
- english2
- esperanto (esperanto)
- espaniol (espaniol)
- ethiopia (ethiopian)
- estonian
- esperanto (esperanto)
- english2
- english1
- dutch
- english1
- english2
- esperanto (esperanto)
- espaniol (espaniol)
- ethiopia (ethiopian)
- estonian
- esperanto (esperanto)
- english2
- english1
- dutch
- english1
- english2
- esperanto (esperanto)
- espaniol (espaniol)
- ethiopia (ethiopian)
- estonian
- esperanto (esperanto)
- english2
- english1
- dutch
- english1
- english2
- esperanto (esperanto)
- espaniol (espaniol)
- ethiopia (ethiopian)
- estonian
- esperanto (esperanto)
- english2
- english1
- dutch
- english1
- english2
- esperanto (esperanto)
- espaniol (espaniol)
- ethiopia (ethiopian)
- estonian
- esperanto (esperanto)
- english2
- english1
- dutch
- english1
- english2
- esperanto (esperanto)
- espaniol (espaniol)
- ethiopia (ethiopian)
- estonian
- esperanto (esperanto)
- english2
- english1
- dutch
- english1
- english2
- esperanto (esperanto)
- espaniol (espaniol)
- ethiopia (ethiopian)
- estonian
- esperanto (esperanto)
- english2
- english1
- dutch
- english1
- english2
- esperanto (esperanto)
- espaniol (espaniol)
- ethiopia (ethiopian)
- estonian
- esperanto (esperanto)
- english2
- english1
- dutch
- english1
- english2
- esperanto (esperanto)
- espaniol (espaniol)
- ethiopia (ethiopian)
- estonian
- esperanto (esperanto)
- english2
- english1
- dutch
- english1
- english2
- esperanto (esperanto)
- espaniol (espaniol)
- ethiopia (ethiopian)
- estonian
- esperanto (esperanto)
- english2
- english1
- dutch
- english1
- english2
- esperanto (esperanto)
- espaniol (espaniol)
- ethiopia (ethiopian)
- estonian
- esperanto (esperanto)
- english2
- english1
- dutch
- english1
- english2
- esperanto (esperanto)
- espaniol (espaniol)
- ethiopia (ethiopian)
- estonian
- esperanto (esperanto)
- english2
- english1
- dutch
- english1
- english2
- esperanto (esperanto)
- espaniol (espaniol)
- ethiopia (ethiopian)
- estonian
- esperanto (esperanto)
- english2
- english1
- dutch
- english1
- english2
- esperanto (esperanto)
- espaniol (espaniol)
- ethiopia (ethiopian)
- estonian
- esperanto (esperanto)
- english2
- english1
- dutch
- english1
- english2
- esperanto (esperanto)
- espaniol (espaniol)
- ethiopia (ethiopian)
- estonian
- esperanto (esperanto)
- english2
- english1
- dutch
- english1
- english2
- esperanto (esperanto)
- espaniol (espaniol)
- ethiopia (ethiopian)
- estonian
- esperanto (esperanto)
- english2
- english1
- dutch
- english1
- english2
- esperanto (esperant
1.4.5 Section-level mini-tables

The section-level table of contents is typeset in the \textfont font, which is defined as \normalsize\rmfamily by default. In fact, the font \textfont is selected at the beginning of a secttoc, sectlof or sectlot.

More selective choices are made with the following fonts. Subsection entries are typeset in the \SSfont font, which is \normalsize\bfseries by default. Subsubsection entries are typeset in the \SSSfont font, which is \normalsize\rmfamily by default.

\Pfont and \SPfont are available (by default, \normalsize\rmfamily) if you want to use various fonts. Section-level lists of figures and tables are typeset in the fonts \slffont and \slttfont, which are defined as \normalsize\rmfamily by default. There are also \slfSSfont and \slttSSfont for sub-figures and sub-tables entries.

1.4.6 Titles for section-level mini-tables

Titles are typeset in the \textfont (\normalsize\bfseries by default) font and the text strings of the titles are defined by \stctitle, \slftitle and \sltttitle, which are the strings “Contents”, “Figures” and “Tables” by default. These title commands should be redefined by \renewcommand or \mtcsettitle for languages other than English.

The language definition files like \texttt{french.mld} and \texttt{english.mld} (and also many others, as listed in table 1.7 on the preceding page and explained in section 1.4.14 on page 44) are available. You can easily prepare a similar file for your preferred language (see section 2.26 on page 70). You can change the language of these titles by using the \mtcselectlanguage{language} macro.

1.4.7 Position of the titles

1.4.7.1 For mini-tables at the part level

By default, titles are on the left. The preparation commands \doparttoc, \dopartlof and \dopartlot accept an optional argument to change the default position of the corresponding title: \texttt{[l]} for left (default), \texttt{[c]} for center, \texttt{[r]} for right, or \texttt{[e]} (or \texttt{n}) for empty (no title).

The change is global for all the document. If you want to change the position of the title for only one \parttoc (or \partlof or \partlot), just use such an optional argument with the command \parttoc (or \partlof or \partlot).
The minitoc package

1.4.7.2 For mini-tables at the chapter level

\dominitoc By default, titles are on the left. The preparation commands \dominitoc, \dominilof and \dominilot accept an optional argument to change the default position of the corresponding title: [l] for left (default), [c] for center, [r] for right, or [e] (or [n]) for “empty” (“no” title). The change is global for all the document. If you want to change the position of the title for only one minitoc (or minilof or minilot), just use such an optional argument with the command \minitoc (or \minilof or \minilot).

1.4.7.3 For mini-tables at the section level

\dosecttoc By default, titles are on the left. The preparation commands \dosecttoc, \dosectlof and \dosectlot accept an optional argument to change the default position of the corresponding title: [l] for left (default), [c] for center, [r] for right, or [e] (or [n]) for empty (no title). The change is global for all the document. If you want to change the position of the title for only one secttoc (or sectlof or sectlot), just use such an optional argument with the command \secttoc (or \sectlof or \sectlot).

1.4.7.4 Summary of the positioning of titles

\doparttoc To summarize: by default, all titles are on the left. However, each one of the following preparation commands: \doparttoc, \dopartlof, \dopartlot, \dominitoc, \dominilof, \dominilot, \dosecttoc, \dosectlof, \dosectlot, \mtcprepare accepts an optional argument to change the positioning of the title: [l] for left (default), [c] for center, [r] for right, [e] or [n] for empty (no title), for all the corresponding mini-tables (for all mini-tables in the case of \mtcprepare).

\parttoc The following insertion commands: \parttoc, \partlof, \partlot, \minitoc, \minilof, \minilot, \secttoc, \sectlof, \sectlot \minilot \secttoc accept the same optional arguments, but these options change the positioning only for the title of the current mini-table. \sectlof \sectlot
1.4.8 Line spacing in the mini-tables

With the commands \tightmtctrue (or the tight package option) and \tightmtcfalse (or the loose package option, which is the default), the mini-tables will have less (tight) or more (loose) space between contents lines.

But with the KOMA-Script classes [343, 344, 399] (scrartcl, scrbook and scrreprt), it may sometimes be necessary to use the following options or commands, because we need to set \parskip to zero in place of \parsep to tighten the mini-table. The efficiency of the following options depends on the options given to these KOMA-Script classes (parindent option, \parskip option and variants).

For the KOMA-Script classes, with the commands \ktightmtctrue (or the k-tight package option) and \ktightmtcfalse (or the k-loose package option, which is the default), the mini-tables will have less (tight) or more (loose) space between contents lines.

1.4.9 Simplified commands for fonts

To simplify the redefinition of the fonts for mini-tables, there are two useful commands:

\mtcsetfont{mini-table}{sectionning-level}{commands}
\mtcsettitlefont{mini-table}{commands}

For instance,

\mtcsetfont{minitoc}{subsection}{\small\rmfamily\upshape\bfseries}
\mtcsettitlefont{minilof}{\small\rmfamily\upshape\bfseries}

will redefine \mtcSSfont and \mlfSfont with the given font commands.

Note that \mtcsetfont{parttoc}{^*}{...} allows also to redefine \ptcfont, etc.

Moreover,

\mtcsettitlefont{parttoc}{\Large\rmfamily\itshape\mdseries}

will redefine \ptifont (for titles in the parttocs, partlofs and partlofs) with the given font commands.
1.4.10 Simplified command for mini-table titles

\mtcsettitle To simplify the redefinition of the titles for mini-tables, the \mtcsettitle command is also available:

\mtcsettitle{mini-table}{title string}

For instance,

\mtcsettitle{minitoc}{Description of contents}

will redefine \mtctitle with the given string. This command checks that you redefine a title for a mini-table type available in your document class.

1.4.11 Simplified command for mini-table depths

\mtcsetdepth To simplify the redefinition of the depths for mini-tables, the \mtcsetdepth command is also available:

\mtcsetdepth{mini-table}{depth}

For instance,

\mtcsetdepth{minitoc}{4}

will set the counter minitocdepth with the given value. This command checks that you set a depth for a mini-table type available in your document class (and that it is possible to change its depth).

1.4.12 Simplified command for mini-table offsets

\mtcsetoffset To simplify the redefinition of the offsets for mini-tables, the \mtcsetoffset command is also available:

\mtcsetoffset{mini-table}{offset}
For instance,

\texttt{mtcsetoffset\{minitoc\}\{-4em\}}

will set the macro \texttt{mtcoffset} to the given value. This command checks that you set a offset for a mini-table type available in your document class (and that it is possible to change its offset).

\textbf{NOTE:} the argument of \texttt{mtcsetoffset} is \textit{not} verified. It must be a length value, without shrink nor stretch part. A positive offset is towards the right, a negative one towards the left.

### 1.4.13 Polymorphic entries in the mini-tables

The title of a sectionning command can appear in several places: a) at the beginning of the section, of the chapter or of the part; b) in the page header; c) in the main TOC; d) in the minitoc of the chapter (for a section title or lower); e) in the parttoc of the part (for a chapter title or lower). A sectionning command has two arguments: an optionnal one, \texttt{OA}, and a mandatory one, \texttt{MA}, like in:

\texttt{\section\{OA\}\{MA\}}

\texttt{OA} is taken as \texttt{MA} if omitted. Normally, \texttt{OA} is used in the TOC and in the minitables, as in the page headers when necessary. \texttt{MA} is used as title for the sectionning unit and is the default for \texttt{OA}. But, some times, you may need to have a different version (a variant) for a sectionning unit title in a minitable. So, it is now possible to define such variants by detecting if that title is used inside some minitable: the following flags are defined (when meaningful):

<table>
<thead>
<tr>
<th>Level</th>
<th>Flag:</th>
<th>for tocs,</th>
<th>for lofs,</th>
<th>for lots,</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part</td>
<td>\texttt{\ifinparttoc} \texttt{\ifinpartlof} \texttt{\ifinpartlot}</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chapter</td>
<td>\texttt{\ifinminitoc} \texttt{\ifinminilof} \texttt{\ifinminilot}</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Section</td>
<td>\texttt{\ifinsecttoc} \texttt{\ifinsectlof} \texttt{\ifinsectlot}</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

But these flags are used to build three new commands, to be called from inside the optionnal argument (\texttt{OA}) of a sectionning command or that of a caption:

<table>
<thead>
<tr>
<th>From \texttt{OA} of:</th>
<th>Command</th>
<th>Arg. 1</th>
<th>Arg. 2</th>
<th>Arg. 3</th>
<th>Arg. 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>sect. command</td>
<td>\texttt{\mtcpolymtoc}</td>
<td>\rightarrow\texttt{\parttoc}</td>
<td>\rightarrow\texttt{\minitoc}</td>
<td>\rightarrow\texttt{\secttoc}</td>
<td>\rightarrow\texttt{\maintoc}</td>
</tr>
<tr>
<td>figure caption</td>
<td>\texttt{\mtcpolymlof}</td>
<td>\rightarrow\texttt{\partlof}</td>
<td>\rightarrow\texttt{\minilof}</td>
<td>\rightarrow\texttt{\sectlof}</td>
<td>\rightarrow\texttt{\mainlof}</td>
</tr>
<tr>
<td>table caption</td>
<td>\texttt{\mtcpolymlot}</td>
<td>\rightarrow\texttt{\partlot}</td>
<td>\rightarrow\texttt{\minilot}</td>
<td>\rightarrow\texttt{\sectlot}</td>
<td>\rightarrow\texttt{\mainlot}</td>
</tr>
</tbody>
</table>
Such entries are “polymorphic”. See the example mtc-vti.tex, section 4.36 on page 148, for a short demonstration.

### 1.4.14 Languages for the titles

Most of the strings defined in the language definition files (.mld) were taken from the superb babel package [60, 61] of Johannes L. Braams, some were adapted, others were made available by gentle users or taken from specific packages, like ArabTeX [276, 277], Aebb [243], ArmTeX (armenian) [142], BangTeX (bangla, bengali) [362], CervanTeX (spanish) [47]. Devanagari for TeX [364], ethiop [44], guarani [45], malayalam [4] and omal [5], MongolTeX (mongol) [137, 140], Gujar (chinese, korean-hangul/hanja, japanese, thai) [127, 297, 298], polski [357, 463] (polish), SETeX [318] (swedish), FarsiTeX [162] (farsi or iranian), or vietnam — latvian (letton), greek-mono, greek-polydemo, greek-polykatha, polish2, russian2m, russian2o and spanish3 need Lambda (A), i.e., the Omega (Ω) version of LEP, (see [272]), or even found by searching on the Web (bulgarianb.mld for upper cyrillic bulgarian, japanese.mld for japanese, serbianc.mld for cyrillic serbian). Other languages are welcome 11. See table 1.7 on page 38.

But for some oriental languages 12, the sources of the titles use some exotic encodings which are difficult to manipulate in a .dtx file, hence the .mld file is then just a wrapper which loads a special file, nicknamed a .mlo file 13, not generated by the .dtx file in the current version of minitoc package, but via filecontents environments in the minitoc.ins file, and playing with the “catcode” of the “delete” character.

### 1.4.15 Altering the layout of the mini-tables

The layout of a mini-table is described in the figure 1.2 on the next page (this figure is adapted from [469]), which defines some internal commands (these are not dimensions, but \LaTeX commands, created by \newcommand, modifiable via \renewcommand).

- \@dotsep, which is the separation between the dots in the dotted line. It is a pure number expressing math units; 18 math units make 1em (one quad), which is about the width of a “m” in the current font. As the real size of 1em is font dependent, the separation between the dots may vary if you use different fonts for different types of entries in the mini-tables.

- \@pnumwidth, is the width of the space reserved for the page number. It is a \LaTeX command containing the representation of a length (e.g., 1.55em).

- \@tocrmarg, is the distance (margin) between the right border of the table and the end of the dotted line. It should be larger than \@pnumwidth, and can be a rubber length (i.e., contain some glue, like 2.55em plus 1fil); if you specify the

---

11 I am searching for the titles in corsican, in particular.

12 Mainly for chinese, farsi, hangul (korean), hanja (korean), japanese, malayalam-omega, thai and some variants of russian.

13 The extension .mlo means minitoc language object.
The minitoc package

\linewidth-2\mtcindent

\indent \numwidth \@tocrmarg

3.5 Heading... ... title
continue... ... title
title end . . . . . . . . . . . . . . . 487

\@dotsep \@pnumwidth

\mtcsetformat

As these commands are internal (their names contain the “@” character) and must have a local effect only on the specified kinds of mini-tables, you should alter them indirectly via the \mtcsetformat command:

\mtcsetformat{mini-table}{parameter}{value}

where mini-table is one of the parttoc, partlof, partlot, minitoc, minilof, minilot, secttoc, sectlof or sectlot keywords; parameter is one of the dotinterval (for \@dotsep), pagenumwidth (for \@pnumwidth), or tocrrightmargin (for \@tocrmarg) keywords; so:

\mtcsetformat{partlof}{tocrrightmargin}{2.55em plus 1fil}

will set the right margin to 2.55em plus 1fil in the lists of tables at the part level. The elasticity (plus 1fil) is useful if the table captions are long (it prevents most hyphenations).

Note that the tocrrightmargin (for \@tocrmarg) parameter should obviously be greater than the pagenumwidth parameter (this appears in the figure 1.2).

If the dotinterval parameter (for \@dotsep) is large enough (try 450, then increase or decrease), the dots of leaders will be so much spread out that they will disappear.

Figure 1.2: Layout of a ToC (LoF, LoT) entry

“... plus 1fil” portion, the text of the entry will be ragged on right; it is useful if you have long entries, and it can avoid most hyphenations.
Table 1.8: Horizontal rules

| \ptcrule | parttoc | \noptcrule | parttoc | \plfrule | partlofs | \noplfrule | partlofs | \pltrule | partlots | \nopltrule | partlots | \mtcrule | minitocs | \nomtcrule | minitocs | \mlfrule | minilofs | \nomlfrule | minilofs | \mltrule | minilots | \nomltrule | minilots | \stcrule | secttocs | \nostcrule | secttocs | \slfrule | sectlofs | \noslfrule | sectlofs | \sltrule | sectlots | \nosltrule | sectlots |
|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| \ptcrule | parttoc | \noptcrule | parttoc | N | N | Y | \plfrule | partlofs | \noplfrule | partlofs | N | N | Y | \pltrule | partlots | \nopltrule | partlots | N | N | Y | \mtcrule | minitocs | \nomtcrule | minitocs | Y | Y | (NA) | \mlfrule | minilofs | \nomlfrule | minilofs | Y | Y | (NA) | \mltrule | minilots | \nomltrule | minilots | Y | Y | (NA) | \stcrule | secttocs | \nostcrule | secttocs | (NA) | (NA) | Y | \slfrule | sectlofs | \noslfrule | sectlofs | (NA) | (NA) | Y | \sltrule | sectlots | \nosltrule | sectlots | (NA) | (NA) | Y |

(NA) = not available.

Table 1.9: Page numbers

<table>
<thead>
<tr>
<th>Type</th>
<th>Page numbers (Default)</th>
<th>No page numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>parttoc</td>
<td>\ptcpagenumbers</td>
<td>\nptcpagenumbers</td>
</tr>
<tr>
<td>minitoc</td>
<td>\mtcpagenumbers</td>
<td>\nmtcpagenumbers</td>
</tr>
<tr>
<td>secttoc</td>
<td>\stcpagenumbers</td>
<td>\nstcpagenumbers</td>
</tr>
<tr>
<td>partlofs</td>
<td>\plfpagenumbers</td>
<td>\nplfpagenumbers</td>
</tr>
<tr>
<td>minilofs</td>
<td>\mlfpagenumbers</td>
<td>\nmlfpagenumbers</td>
</tr>
<tr>
<td>sectlofs</td>
<td>\slfpagenumbers</td>
<td>\nslfpagenumbers</td>
</tr>
<tr>
<td>partlots</td>
<td>\pltpagenumbers</td>
<td>\npltpagenumbers</td>
</tr>
<tr>
<td>minilot</td>
<td>\mltpagenumbers</td>
<td>\nmltpagenumbers</td>
</tr>
<tr>
<td>sectlot</td>
<td>\sltpagenumbers</td>
<td>\nsltpagenumbers</td>
</tr>
</tbody>
</table>

1.5 Special Features

1.5.1 Horizontal Rules

By default, most of mini-tables have horizontal rules after their titles and at their ends. The exception is the “parttoc” in a book- or report-like document (i.e., when \chapter is defined). To activate or deactivate these rules, the commands of the table 1.8 are available. But you can also use the following command, which is simpler:

\mtcsetrules{mini-table}{on|off}
where \textit{mini-table} is one of the \texttt{parttoc}, \texttt{partlof}, \texttt{partlot}, \texttt{minitoc}, \texttt{minilof}, \texttt{minilot}, \texttt{secttoc}, \texttt{sectlof}, or \texttt{sectlot} keywords; if the first argument is a star (*), all mini-tables are affected; the keywords \texttt{on} and \texttt{off} have the following synonyms \cite{14}:

- \texttt{on}, \texttt{ON}, \texttt{yes}, \texttt{Y}, \texttt{true}, \texttt{TRUE}, \texttt{t}, \texttt{T}, \texttt{vrai}, \texttt{VRAI}, \texttt{v}, \texttt{V}, \texttt{oui}, \texttt{OUI}, \texttt{o}, \texttt{O}, +, and 1;
- \texttt{off}, \texttt{OFF}, \texttt{no}, \texttt{NO}, \texttt{n}, \texttt{false}, \texttt{FALSE}, \texttt{faux}, \texttt{FAUX}, \texttt{f}, \texttt{F}, \texttt{non}, \texttt{NON}, -, and 0.

\section*{1.5.2 Page Numbers, Leaders}

\texttt{mtcsetpagenumbers} By default, the page numbers are listed in each \texttt{minitoc}, \texttt{minilof}, etc. Some authors want only the section titles (with the section numbers), but without page numbers. Hence the obvious declarations of table \ref{table:19} on the preceding page are available. But you can also use the following command:

\begin{verbatim}
\mtcsetpagenumbers{\texttt{mini-table}|*}{\texttt{on}|\texttt{off}}
\end{verbatim}

where \texttt{mini-table} is one of the \texttt{parttoc}, \texttt{partlof}, \texttt{partlot}, \texttt{minitoc}, \texttt{minilof}, \texttt{minilot}, \texttt{secttoc}, \texttt{sectlof}, or \texttt{sectlot} keywords; the keywords \texttt{on} and \texttt{off} have the following synonyms \cite{14}:

- \texttt{on}, \texttt{ON}, \texttt{yes}, \texttt{Y}, \texttt{true}, \texttt{TRUE}, \texttt{t}, \texttt{T}, \texttt{vrai}, \texttt{VRAI}, \texttt{v}, \texttt{V}, \texttt{oui}, \texttt{OUI}, \texttt{o}, \texttt{O}, +, and 1;
- \texttt{off}, \texttt{OFF}, \texttt{no}, \texttt{NO}, \texttt{n}, \texttt{false}, \texttt{FALSE}, \texttt{faux}, \texttt{FAUX}, \texttt{f}, \texttt{F}, \texttt{non}, \texttt{NON}, -, and 0.

If the first argument is a star (*), all mini-tables are affected.

In the mini-tables, they are leaders of dots between the section titles and the page numbers. The \texttt{undotted} package option removes these dots. The \texttt{dotted} package option is the default. See also section \ref{section:19.15} on page \pageref{section:19.15}.

\section*{1.5.3 Features for parttocs and other mini-tables}

By default, a parttoc (or a partlof or a partlot), in a \texttt{book}- or \texttt{report}-class document, is preceded and followed by a \texttt{cleardoublepage} (which acts like \texttt{clearpage} in a one-side document), and has a page style of \texttt{empty}. Since version \#32, you can modify this behaviour by redefining the commands of table \ref{table:10} on the following page, whose meaning is often obvious. A feature defined as \texttt{empty} does nothing.

\cite{14} 0 and o are the letter O, 0 is the zero digit.
Table 1.10: Features for mini-tables

<table>
<thead>
<tr>
<th>Type</th>
<th>Command</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>parttoc</td>
<td>\beforeparttoc</td>
<td>\cleardoublepage</td>
</tr>
<tr>
<td>parttoc</td>
<td>\afterparttoc</td>
<td>\cleardoublepage</td>
</tr>
<tr>
<td>parttoc</td>
<td>\openparttoc</td>
<td>\empty</td>
</tr>
<tr>
<td>parttoc</td>
<td>\closeparttoc</td>
<td>\empty</td>
</tr>
<tr>
<td>parttoc</td>
<td>\thistoparttocstyle</td>
<td>\thispagestyle{empty}</td>
</tr>
<tr>
<td>partlof</td>
<td>\beforepartlof</td>
<td>\cleardoublepage</td>
</tr>
<tr>
<td>partlof</td>
<td>\afterpartlof</td>
<td>\cleardoublepage</td>
</tr>
<tr>
<td>partlof</td>
<td>\openpartlof</td>
<td>\empty</td>
</tr>
<tr>
<td>partlof</td>
<td>\closepartlof</td>
<td>\empty</td>
</tr>
<tr>
<td>partlof</td>
<td>\thistopartlofstyle</td>
<td>\thispagestyle{empty}</td>
</tr>
<tr>
<td>partlot</td>
<td>\beforepartlot</td>
<td>\cleardoublepage</td>
</tr>
<tr>
<td>partlot</td>
<td>\afterpartlot</td>
<td>\cleardoublepage</td>
</tr>
<tr>
<td>partlot</td>
<td>\openpartlot</td>
<td>\empty</td>
</tr>
<tr>
<td>partlot</td>
<td>\closepartlot</td>
<td>\empty</td>
</tr>
<tr>
<td>partlot</td>
<td>\thistopartlotstyle</td>
<td>\thispagestyle{empty}</td>
</tr>
<tr>
<td>minitoc</td>
<td>\beforeminitoc</td>
<td>\empty</td>
</tr>
<tr>
<td>minitoc</td>
<td>\afterminitoc</td>
<td>\empty</td>
</tr>
<tr>
<td>minitoc</td>
<td>\openminitoc</td>
<td>\empty</td>
</tr>
<tr>
<td>minitoc</td>
<td>\closeminitoc</td>
<td>\empty</td>
</tr>
<tr>
<td>minitoc</td>
<td>\thistopminitocstyle</td>
<td>\empty</td>
</tr>
<tr>
<td>minilof</td>
<td>\beforeminilof</td>
<td>\empty</td>
</tr>
<tr>
<td>minilof</td>
<td>\afterminilof</td>
<td>\empty</td>
</tr>
<tr>
<td>minilof</td>
<td>\openminilof</td>
<td>\empty</td>
</tr>
<tr>
<td>minilof</td>
<td>\closeminilof</td>
<td>\empty</td>
</tr>
<tr>
<td>minilof</td>
<td>\thistopminilofstyle</td>
<td>\empty</td>
</tr>
<tr>
<td>minilot</td>
<td>\beforeminilot</td>
<td>\empty</td>
</tr>
<tr>
<td>minilot</td>
<td>\afterminilot</td>
<td>\empty</td>
</tr>
<tr>
<td>minilot</td>
<td>\openminilot</td>
<td>\empty</td>
</tr>
<tr>
<td>minilot</td>
<td>\closeminilot</td>
<td>\empty</td>
</tr>
<tr>
<td>minilot</td>
<td>\thistopminilotstyle</td>
<td>\empty</td>
</tr>
<tr>
<td>secttoc</td>
<td>\beforesecttoc</td>
<td>\empty</td>
</tr>
<tr>
<td>secttoc</td>
<td>\aftersecttoc</td>
<td>\empty</td>
</tr>
<tr>
<td>secttoc</td>
<td>\opensecttoc</td>
<td>\empty</td>
</tr>
<tr>
<td>secttoc</td>
<td>\closesecttoc</td>
<td>\empty</td>
</tr>
<tr>
<td>secttoc</td>
<td>\thistopsecttocstyle</td>
<td>\empty</td>
</tr>
<tr>
<td>sectlof</td>
<td>\beforesectlof</td>
<td>\empty</td>
</tr>
<tr>
<td>sectlof</td>
<td>\aftersectlof</td>
<td>\empty</td>
</tr>
<tr>
<td>sectlof</td>
<td>\opensectlof</td>
<td>\empty</td>
</tr>
<tr>
<td>sectlof</td>
<td>\closesectlof</td>
<td>\empty</td>
</tr>
<tr>
<td>sectlof</td>
<td>\thistopsectlofstyle</td>
<td>\empty</td>
</tr>
<tr>
<td>sectlot</td>
<td>\beforesectlot</td>
<td>\empty</td>
</tr>
<tr>
<td>sectlot</td>
<td>\aftersectlot</td>
<td>\empty</td>
</tr>
<tr>
<td>sectlot</td>
<td>\opensectlot</td>
<td>\empty</td>
</tr>
<tr>
<td>sectlot</td>
<td>\closesectlot</td>
<td>\empty</td>
</tr>
<tr>
<td>sectlot</td>
<td>\thistopsectlotstyle</td>
<td>\empty</td>
</tr>
</tbody>
</table>

\mtcsetfeature{mini-table}{before|after|open|close|pagestyle}{command}

Modifies the features for a mini-table.
The command:
\mtcsetfeature{mini-table}{keyword}{commands}
allows you to redefine any of these commands. mini-table is one of the mini-table names: parttoc... sectlot. keyword is one of the followings: before, after, open, close or pagestyle. commands is either a sequence of commands like \clearpage, \cleardoublepage, \thispagestyle{...}, etc., either \empty (does nothing).

### 1.5.3.1 Remark about page styles

The default commands for part-level mini-tables page styles are defined as being simply a standard \thispagestyle{empty} command, because in document classes defining the \chapter command (like book or report), the part-level mini-tables are on their own pages. If the document is printed recto-verso, the first page is recto. Usually, these pages are not numbered and have no header and no footer. This behaviour is a consequence from the default definitions of the commands of table 1.10 on the page before. If you want an other behaviour, you can change these definitions. Note that, by default, only the first page of these mini-tables are in the empty page style. You can set the style of this first page by using \thispagestyle and set the style of the following pages by using \pagestyle, but you must not forget to reset the normal style after the mini-table. Look at this short theoretical example:

```
\mtcsetfeature{parttoc}{before}\
{\cleardoublepage}
\mtcsetfeature{parttoc}{pagestyle}\
{\thispagestyle{empty}\pagestyle{myheadings}}
\mtcsetfeature{parttoc}{after}\
{\cleardoublepage\pagestyle{headings}}
```

where we add a \cleardoublepage before each parttoc, then we set the empty page style for the first page of the parttoccs, the myheadings page style for the following pages of the parttoccs, and set headings page style for the pages after the mini-table, after a \cleardoublepage.

### 1.5.4 The “Chapter 0” Problem (solved)

Some documents do not begin with chapter number one, but with chapter number zero (or even a weirder number).

---

15 This remark is taken and adapted from a draft of the second edition of the JMPL [29], by Benjamin Baxart, where he comments the minitoc package.

16 This example shows that the third argument can be a sequence of commands: we set the style of the current page and the style of the following pages.
Before version #23 (1994/11/08) To make the minitoc package work with such documents, you must insert the command:
\firstchapteris{(N)}
before the \dominitoc and analogous commands. \textit{(N)} is the number of the first chapter. This command \textit{does not} modify the numbering of chapters, you must use a \addtocounter{chapter}{-1} command to get a first chapter numbered 0. The \firstpartis and \firstsectionis commands are similar for parts and sections with a non standard numbering.

Since version #23 (1994/11/08) These commands are now obsolete, as this problem has been solved (via the “absolute” numbering of the mini-table auxiliary files). Thus now they just produce harmless warnings.

1.5.5 Special Entries in the TOC

If you want to add entries in the Table of Contents for objects like the Table of Contents itself, the List of Figures, the List of Tables, the Bibliography or the Index, you should use the tocbibind package \cite{472} by Peter R. Wilson (package available from the CTAN archives).

\dominitoc But these entries are considered as chapters (or sections in an article class document) when the .toc file is scanned to prepare the minitocs (the \dominitoc phase).

Note that the same problems appear if you use one of the \texttt{scrbook}, \texttt{scrreprt} or \texttt{scrartcl} KOMA-Script classes \cite{343, 344, 399} with some options (\texttt{liststotoc}, \texttt{liststotocnumbered}, \texttt{bibtotoc}, \texttt{bibtotocnumbered}, and \texttt{idxtotoc}). The solutions are the same ones.

\mtcaddchapter \tableofcontents \listoffigures \listoftables \adjustmtc \bibliography So you must add a \mtcaddchapter command, \textit{without argument}, after each of the involved commands \texttt{tableofcontents}, \texttt{listoffigures}, and \texttt{listoftables}.

For the bibliography, you should add a \adjustmtc command after the \texttt{bibliography} command.

\printglossary \addcontentsline \mtcaddchapter \mtcfixglossary For the glossary, it is a bit more complicated, you should add the following commands just after the \texttt{printglossary} command:
\addcontentsline{lof}{xchapter}{1}
\addcontentsline{lot}{xchapter}{1}
But this can be done by:

\mtcfixglossary[\text{chapter}|\text{section}|\text{part}]

where the optional argument is the level for the glossary entry in the TOC. By default, if \text{chapter} is defined, the chapter level is used, else the section level. If neither \text{chapter} or \text{section} are defined, the part level will be used if \text{part} is defined; else an error is reported. You must check the result and, if necessary, adjust the optional argument.

\printindex \addcontentsline \mtcaddchapter \mtcfixindex

For the index, it is like for the glossary, you should add the following commands just after the \printindex command:

\addcontentsline{lof}{xchapter}{}
\addcontentsline{lot}{xchapter}{}
\mtcaddchapter

But this can be done by:

\mtcfixindex[\text{chapter}|\text{section}|\text{part}]

where the optional argument is the level for the index entry in the TOC. By default, if \text{chapter} is defined, the chapter level is used, else the section level. If neither \text{chapter} or \text{section} are defined, the part level will be used if \text{part} is defined; else an error is reported. You must check the result and, if necessary, adjust the optional argument.

\printnomenclature \addcontentsline \mtcaddchapter \mtcfixnomenclature

For the nomenclature, it is like for the glossary, you should add the following commands just after the \printnomenclature command:

\addcontentsline{lof}{xchapter}{}
\addcontentsline{lot}{xchapter}{}
\mtcaddchapter

But this can be done by:

\mtcfixnomenclature[\text{chapter}|\text{section}|\text{part}]

where the optional argument is the level for the nomenclature entry in the TOC. By default, if \text{chapter} is defined, the chapter level is used, else the section level. If neither \text{chapter} or \text{section} are defined, the part level will be used if \text{part} is defined; else an error is reported. You must check the result and, if necessary, adjust the optional argument.

\footnote{If you are using the \text{nomencl} package \cite{456} or the \text{nomentbl} package \cite{161} (nomentbl calls nomencl).

}
Of course, in documents were the TOC, LOF, LOT, bibliography and/or glossary (or index or nomenclature) are processed as starred sections, you must modify these additions to use section level commands.

And proceed with extreme care, tracking in the document.log file the insertion of .mtc(N) files (and siblings). They are some examples in the mtc-add.tex (see section 4.4 on page 96), mtc-ads.tex (see section 4.5 on page 100), and mtc-nom.tex (see section 4.26 on page 136) files distributed with minitoc. The mtc-ads.tex example shows how much that problem is difficult.

### 1.6 The notoccite option

\cite This option loads the notoccite package [14] (by Donald Arseneau). It avoids problems with \cite commands in sectionning commands or captions: if you then run \LaTeX{} using the unsrt (unsorted) style, or a similar style, these citations get numbered starting from the page in the table of contents where is the parasite citation, not the number they should have in the main text. The notoccite package prevents this. As minitoc prints TOCs, it is subject to the same problem. See also http://www.tex.ac.uk/cgi-bin/texfaq2html?label=bibtocorder.

### 1.7 The listfiles and nolistfiles options

The listfiles package option creates a list of the minitoc auxiliary files into the file document.maf. This feature can help you to remove these auxiliary files which are no more necessary after the \LaTeX{} run. Under Unix or Linux, you can try:

```bash
    cat document.maf | xargs -i -t \rm {}  
```

### 1.8 The hints option

This package option detects some actions and the loading of some packages and classes known as interacting with minitoc, and also some frequent misuses and errors. This list of interacting packages and classes is, of course, not closed. If a known package is loaded, this option writes some hints in the document.log file and emits a warning. The hints written in the document.log file may suggest you to consult the present document or the minitoc.bug file. Your advice about this option will be welcome. This option is activated by default, but you can inhibit it via the nohints option. The following (potential) problems are currently detected:

---

[1] — The minitoc package

This package option is now (since version #48) the default (list created).
The commands containing the "@" character in their names are internal commands of LaTeX, of a package or of a class; they are sometimes altered by another packages; reconsider then the loading order of the packages.

This package has its own system for minitocs.

The titlesec package redefines the sectioning commands in a way completely alien to the standard LaTeX way; hence minitoc and titlesec-titeloc are fundamentally incompatible, and it is very sad.
[1] — The minitoc package

- If the \texttt{romannum} package \cite{romannum} (by Peter R. Wilson) is used, it must be loaded \textit{before} the \texttt{minitoc} package.
- If the \texttt{sheaders} package \cite{sheaders} (by Maurizio Loreti) is used, it must be loaded \textit{before} the \texttt{minitoc} package.
- If the \texttt{alnumsec} package \cite{alnumsec} (by Frank Küster) is used, it must be loaded \textit{before} the \texttt{minitoc} package.
- If the \texttt{capcont} package \cite{capcont} (by Steven Douglas Cochran) is used, it must be loaded \textit{before} the \texttt{minitoc} package.
- If one of the \texttt{caption} \cite{caption}, \texttt{caption2} \cite{caption2}, (both written by Axel Sommerfeldt), \texttt{ccaption} \cite{ccaption} (written by Peter R. Wilson), or \texttt{mcaption} \cite{mcaption} (written by Stephan Hennig), packages is used, it must be loaded \textit{before} the \texttt{minitoc} package.
- If one of the \texttt{float} \cite{float}, \texttt{floatrow} \cite{floatrow}, \texttt{trivfloat} \cite{trivfloat}, or \texttt{rotfloat} \cite{rotfloat} packages is used, you must remember that you can not use the \texttt{minitoc} facilities for preparing mini-tables of floats of the new defined types.
- If you try to insert empty mini-tables, the \texttt{hints} option gives a global warning (except if you used also the \texttt{nocheckfiles} option, see section 1.3.3 on page 29).
- If you use one of the obsolete commands (\texttt{\firstpart}, \texttt{\firstchapter}, or \texttt{\firstsection}), a warning is issued for each use, of course, but also a global hint as reminder.
- If you invoke a same preparation command more than once, an informative hint is issued for each spurious invocation.

1.9 Usage with MS-DOS

Under MS-DOS (and other PC oriented old operating systems), the filename extensions are limited to 3 characters. The \texttt{minitoc} package determines dynamically the type of extensions available and will use it. All other modifications will be done automatically. The \texttt{.mtc(N)} extensions will become \texttt{.M(N)}, where \texttt{(N)} is the absolute chapter number. The extensions \texttt{.mlf(N)} and \texttt{.mlt(N)} become \texttt{.F(N)} and \texttt{.T(N)}. The \texttt{.ptc(N)} extensions become \texttt{.P(N)}, where \texttt{(N)} is the absolute part number. The extensions \texttt{.plf(N)} and \texttt{.plt(N)} become \texttt{.G(N)} and \texttt{.U(N)}. The \texttt{.stc(N)} extensions become \texttt{.S(N)}, where \texttt{(N)} is the absolute section number. The extensions \texttt{.slf(N)} and \texttt{.slt(N)} become \texttt{.H(N)} and \texttt{.V(N)}. All these extensions are listed in table 1.11 on the following page. Of course, this implies a limit of 99 chapters in a document, but do you really need so many chapters (or sections in an article)? The limit of 99 parts does not seem too serious for most documents, but for sections, it could be tragic. The \texttt{hints} option (section 1.8 on page 52) will report such situations. See also section 2.5 on page 58.

\footnote{This package is obsolete; now use the \texttt{caption} package.}
1.10 Why several \LaTeX\ runs are required?

The mini-tables, at part, chapter and section levels, are using some space on the first pages on each chapter, part or section, thus the page numbers are altered. After the first \LaTeX\ run, the mini-tables and lists, partial tables and lists and section-level tables and lists will be empty (in fact skipped since version #35); after the second run, they appear (if not empty), but because they modify the page numbering, page numbers are wrong; after the third \LaTeX\ run, the mini, part- and section-level tables and lists should be correct (see figure 2.1 on page 59).

1.11 The mtcoff package

If a document has been prepared with the minitoc package, it contains many minitoc specific commands, most of them being \dominitoc, \faketableofcontents, and \minitoc commands (and their equivalents for lists of figures and tables). If you want to typeset this document without any mini-table, you have just to replace the minitoc package by the mtcoff package (without option), and all these commands will be ignored, eventually writing warning messages in the document.log file. At least two \LaTeX\ runs will be necessary to get a correct page numbering and cross references. It also sanitizes the .aux, .toc, .lof, and .lot files from minitoc specific commands which are now spurious.
Chapter 2

Frequently Asked Questions

Contents

2.0 Introduction ........................................................................................................... 57
2.1 Avoiding a page break near the rules before and after a mini-table .................. 58
2.2 Implementing others layouts for a mini-table ..................................................... 58
2.3 A “\"” command in a contents line makes an error ........................................... 58
2.4 Reordering chapters makes havoc ...................................................................... 58
2.5 Extensions for the names of auxiliary files ....................................................... 58
2.6 Playing with the chapter number ..................................................................... 59
2.7 Supported document classes .......................................................................... 60
2.8 Compatibility with \LaTeX versions ..................................................................... 60
2.9 Other mini-tables ............................................................................................. 60
2.10 Why so many auxiliary files? .......................................................................... 61
2.11 Mini-tables at levels other than chapter ........................................................ 61
2.12 Incompatibility with \LaTeX2.09 ....................................................................... 62
2.13 Documents resetting the chapter number at each part ................................... 62
2.14 The mini-tables have too much spaced lines .................................................. 62
2.15 The secttocs are wrong .................................................................................... 62
2.16 Removing the lines of dots ............................................................................. 62
2.17 Using the hyperref package with minitoc ....................................................... 63
2.18 Problem while upgrading minitoc .................................................................... 63
2.19 A local TOC for the set of appendices ............................................................ 63
2.20 Use with the appendix package .................................................................... 64
2.21 Use with the tocloft package ......................................................................... 64
2.22 Use with the memoir class .......................................................................... 65
2.23 There are too many commands for fonts, titles, and depths ............................ 66
2.24 Compatibility with the \AMS document classes ............................................. 66
2.25 Hiding some entries from the main table of contents ..................................... 67
2.26 Defining your own .mld file ............................................................................ 70
2.27 Use with the abstract package ....................................................................... 70
2.28 Use with the sectsty package ........................................................................ 70
2.29 Strange alignment in the minitocs .................................................................. 71
2.30 Useful precautions with starred sectioning commands ................................... 72
2.31 Use with packages for captions ..................................................................... 72
2.32 Bad interaction minitoc/hyperref/memoir ....................................................... 72
Frequently Asked Questions

2.0 Introduction

Here is a list of problems and frequently asked questions about the `minitoc.sty` package. If the version has a number less than 61, please upgrade to version #61. This list is also given in the `minitoc.bug` file, in pure text form. The numbering of this list is done by date of the first occurrence of the question.

If a problem arises, it is often wise to: a) use the `hints` option (see section 1.8 on page 52), which is activated by default, and b) read the `document.log` file, which may contain pertinent messages. If you do not find a solution, ask a question on an adequate news group, like `fr.comp.text.tex` (in french) or `comp.text.tex` (in english) preferably, groups which I try to follow, or send me a mail in last ressort (please join a minimal but complete example [384, 432]\(^1\) (or “MCE”) reproducing the problem; this example should use the `hints` option).

\(^1\) See also: http://www.tex.ac.uk/cgi-bin/texfaq2html?label=minxampl and http://www.tex.ac.uk/cgi-bin/ texfaq2html?label=askquestion for good advices.
2.1 Avoiding a page break near the rules before and after a mini-table

This problem seemed solved since version #8, but version #12 added better fixes. You may have to make some final tuning with \enlargethispage. See the \LaTeX{} manual [279]. The \needspace{} package [468] may also be useful.

2.2 Implementing others layouts for a mini-table

Suggestions are welcome, but look at the section 1.4.15 on page 44. There are yet some examples in chapter 4 on page 90, for some layouts, like mini-tables on two or three columns.

2.3 A “\"" command in a contents line makes an error

Use \protect\linebreak. The \"" command should be used only in tabular material (tabular environment and similar, or in the tabbing environment) and in math arrays and equations, or in the quote-like environments.

2.4 Reordering chapters makes havoc

If you reorder chapters, havoc follows... mini-tables going in wrong chapters.

The best way seems to make one run with the \mtcoff{} package replacing the \minitoc{} package, then restore the \minitoc{} package and re-execute \LaTeX{} at least three times (yes, it is time-consuming...). See figure 2.1 on the next page\footnote{I used the pict2e package [178], by Hubert Gässlein, Rolf Nieprasch and Josef Tkadlec, to prepare this figure.}. Running with the \mtcoff{} package ensures that the standard auxiliary files are cleared from “spurious” commands introduced by \minitoc{}. A more radical solution is to delete the .aux, .toc, .lof and .lot files relative to the document, then re-execute \LaTeX{} at least three times.

2.5 Extensions for the names of auxiliary files

This package creates auxiliary files with extensions like .\mtc{}\(N\). Some operating systems allow only 3 characters extensions. What to do?
No modification is needed: all became automatic since version #28! If you insist to use 3 characters extensions, even on operating systems allowing more, just use the package option `shortext` Then you will get first the autoconfiguration messages, then a message saying that you will use short extensions. But then be careful to not have more than 99 mini-tables of the same kind (even empty)!

### 2.6 Playing with the chapter number

Do not cheat with the “chapter” counter, i.e., do not write ugly things like:

```latex
\setcounter{chapter}{6}
```

The mechanism would break. It is better to add \texttt{chapter} commands, to create empty (but numbered in a legal way) chapters. Since version #10, the \texttt{minitoc} package works with appendices. Version #19 allows to begin with a chapter other that number 1. And look at “Special Entries in the TOC”, section 1.5.5 on page 50.

Since version #23 (1994/11/08), the numbering of chapters and that of minitocs are independent, so that problem just vanished.

The same remarks apply to the \texttt{part} and \texttt{section} counters.
2.7 Supported document classes

The minitoc package is restricted to document classes which define chapters in the standard way, like “book” and “report”, or sections in the standard way, like “article” [282]. There are “parttoc” if the document class defines the \part command. Note that classes like “letter” [283], which have not the classical sectioning structure, cannot be supported. Classes using sectioning commands with other names are not supported. See also section 2.24 on page 66.

2.8 Compatibility with \LaTeX\ versions

Some users have failed to make minitoc to work. They got a message like:

Package minitoc Warning: W0021
Undefined command ... \@inputcheck ...
Your version of latex.tex is obsolete. Trying to continue...

or:

Package minitoc Warning: W0022
Undefined command ... \reset@font ...
Your version of latex.tex is very obsolete.
Trying to continue... crossing fingers.

The \reset@font command has been added to latex.tex on September 29th, 1991 and the \@inputcheck command on March 18th, 1992 and this version of latex.tex has been released on March 25th, 1992. If you get this message, you have an old version of latex.tex. Get a recent one from the archives (or a recent distribution) and regenerate a latex.fmt format via initex (or your configuration tool).

2.9 Other mini-tables

Some demanding users want to have minilof, minilot and minibbl (mini-bibliographies per part, chapter or section). First, “minibbl” is another problem, strongly related to the \BibTeX’s dealing with .aux files. Look at the chapterbib [19], bibunits [210], multibib [211], bibtopic [25], and splitbib [314] packages. Version #13 has implemented basic minilofs and minilots. Minibbls are not the aim of this package.

3 This would be very difficult: any user can create new sectioning commands (often with the help from some packages) with standard or new names; this is only limited by the imagination. The minitoc package relies on the names of the standard sectioning commands and on the syntax of these commands.

4 See http://www.tex.ac.uk/cgi-bin/texlaq2html?label=multibib
2.10 Why so many auxiliary files?

This package creates a lot of auxiliary files and some users have argued that it is too many. A deep redesign would be necessary to avoid that. Using only one big auxiliary file (or one for all minitocs, one for all minilofs, ...) would make the reading of such file very slow, as it would be read for each \miniXXX macro! Moreover, this would make the checkfiles (see section 1.3.3 on page 29) package option impractical to implement. Note that the many files *.mtc*, etc., may be deleted after the LATEX run. They are rebuilt by the preparation commands (like \dominitoc and siblings). But, since version #35, minitoc is able to detect and skip empty *.mtc* files (and siblings) to avoid ugly titles with just two thin rules. It would not be easy to do with only one big auxiliary file. Since version #44, the listfiles package option is available to create a list of these auxiliary files; see section 1.7 on page 52.

These files contain the mini-tables extracted from the .toc, .lof, and .lot files. They are no more useful after the LATEX run. If you run LATEX via a script or a “makefile”, it may be useful to add to it a cleaning feature (which should be optional, to allow debugging). The table 1.11 on page 55 gives the list of the extensions for these files (note that a document.mtc auxiliary file is also created as a scratch file).

As an example, you can look at the rubber tool [34] (written in Python) provided by Emmanuel Beffara:

http://iml.univ-mrs.fr/~beffara/soft/rubber/

2.11 Mini-tables at levels other than chapter

Here also, some redesign was needed. From version #15, there are parttocs, partlofs and partlots for the part level in book/report-like and article-like documents, secttocs, sectlofs and sectlots for the section level in article-like documents. Note that you can not have minitocs features at chapter and section level in the same document, because doing so would make an almost unreadable monster. The user must choose the main class of the document according to the size of it (e.g., do not write an article of more than 100 sections: this is a report, or even a book!).

<table>
<thead>
<tr>
<th>part</th>
<th>chapter</th>
<th>section</th>
</tr>
</thead>
<tbody>
<tr>
<td>book</td>
<td>+</td>
<td>*</td>
</tr>
<tr>
<td>report</td>
<td>+</td>
<td>*</td>
</tr>
<tr>
<td>article</td>
<td>+</td>
<td>+</td>
</tr>
</tbody>
</table>
2.12 Incompatibility with LaTeX2.09

The more recent version of \LaTeXe adds \protect before \contentsline in the .toc, .lof and .lot files. The version #17 of minitoc attempts to be compatible with \LaTeXe and \LaTeX2.09. This will be the last version usable with \LaTeX2.09. Versions #18 and later are \LaTeXe specific, and no more compatible with \LaTeX2.09, which is completely obsolete.

2.13 Documents resetting the chapter number at each part

Since version #23, minitoc works with document classes resetting chapter (or section) number at each part (or chapter). This is possible because the auxiliary files for the mini-tables have now an \textit{absolute} number.

2.14 The mini-tables have too much spaced lines

From version #29, you can have tight mini-tables with the \texttt{tight} option, and with the \texttt{k-tight} option for the KOMA-Script classes [343, 344, 399] (since version #43).

2.15 The secttos are wrong

Secttos did not work: corrected (version #38).

2.16 Removing the lines of dots

The lines of dots (leaders) between section titles and page numbers are removed by the \undotted option (#29). See also section 1.4.15 on page 44.

2.17 Using the hyperref package with minitoc

Since version #31, minitoc works correctly with the powerful hyperref package [390], thanks to Heiko Oberdiek, using the work of Bernd Jahnke, Didier Verny and A. J. “Tony” Roberts. \textit{Hence the minitoc-hyper package [454] is now obsolete and should no more be used. It is still present on the CTAN archives for compatibility with old documents.} If you add the loading of the hyperref package to a document yet using minitoc, you will get error message about spurious closing braces. Just let finish the \LaTeX run, then re-\LaTeX the document. There will
be no problem if you remove the loading of hyperref and add it again: the problem occurs only when upgrading from minitoc #30 to minitoc #31 (or higher) with a document already processed and adding hyperref at the same time! It seems better to process the document with minitoc #31 (or higher) without hyperref, then with hyperref, because some internal commands written into the auxiliary files have been modified. If used, the hyperref package must be loaded before minitoc. Note that the documents minitoc.dtx and minitoc-fr.dtx show (not so) basic examples of the use of the hyperref package with minitoc.

2.18 Problem while upgrading minitoc

If upgrading from version #30 or lower to version #31 or higher, you should delete the .aux, .toc, .lof, .lot files of the document, else the first \LaTeX run with version #31 or higher will produce a lot of errors (the next run should be ok). See also the section 2.17 on the preceding page.

2.19 A local TOC for the set of appendices

Some users need a table of contents for the appendices, but without putting the entries of it into the main table of contents. The solution is to put the appendices in a \part subdivision of the document and ask for a table of contents at the \part level:

\doparttoc % after \begin{document}
\tableofcontents
\appendix
\part
\parttoc
\addtocontents
\protect
\setcounter
\chapter
\partbegin
% To suppress the appendix part in the main toc
\addtocontents{toc}\protect\setcounter{tocdepth}{-1}
% Create a part level subdivision
\part{Appendices}
% Create a local table of contents
\parttoc
% Add this at the end of appendices if there is something
% after the appendices (like an index or a bibliography)
% to put a bound to the contents of \parttoc
\addtocontents{toc}\protect\partbegin

See also section 2.25 on page 67.
2.20 Use with the appendix package

If you use the appendix package [471] (by Peter R. Wilson), you will observe a serious problem with minitocs in the appendices environment (and after it): they do not match with their respective appendices. In fact, the environment opening \begin{appendices} hides a \addcontentsline command for a chapter or a section, putting trouble in the numbering of minitocs or secttocs. Several solutions are available. The first one is to add a \adjustmtc or \adjuststc command (depending on the level of the appendices, chapter or section) after each \begin{appendices} command. An other solution is to add the following commands in the preamble after the loading of the appendix package:

\let\oldappendices\appendices
\def\appendices{\oldappendices\adjustmtc}

if appendices are at the chapter level, OR:

\let\oldappendices\appendices
\def\appendices{\oldappendices\adjuststc}

if appendices are at the section level.

These two solutions may be modified by replacing \adjustmtc by the sequence:

\addtocontents{toc}{\chapterend}
OR
\addtocontents{toc}{\sectend}

when it is necessary to delimit the end of the preceding chapter or section.

A rather more elegant solution is to add an entry into the TOC via the \addappheadtotoc command offered by the appendix package. As this entry is a chapter-level (or section-level) entry, itdelimits correctly the end of the preceding chapter or section.

See also the mtc-amm.tex example file (section 4.6 on page 105), which uses the memoir class [479, 481, 482], which includes itself the appendix package functionnality (these packages and this class are from the same author).

2.21 Use with the tocloft package

(This answer is given in the documentation of the tocloft package [469].) The tocloft (by Peter R. Wilson) and minitoc packages have an unfortunate interaction, which fortunately

\footnote{In fact, the commands \partend, \chapterend and \sectend should not be used directly by the user, in normal circumstances.}

\footnote{Discovered by Lyndon Dudding.}
can be fixed. In the normal course of events, when minitoc is used in a chaptered document it will typeset section entries in the minitocs in bold font. If tocloft is used in conjunction with minitoc, then the minitoc section entries are typeset in the normal font, except for the page numbers which are in bold font, while the ToC section entries are all in normal font.

One cure, if you want the minitoc section entries to be all in normal small font, is to put:

\renewcommand{\mtcSfont}{\normalfont\small}

or:

\mtcsetfont{minitoc}{section}{\normalfont\small}

in the preamble.

Otherwise, the cure is the following incantation:

\renewcommand{\cftsecfont}{\bfseries}
\renewcommand{\cftsecleader}{\bfseries\cftdotfill{\cftdotsep}}
\renewcommand{\cftsecpagefont}{\bfseries}

To have the section entries in both the ToC and the minitocs in bold then put the incantation in the preamble. To have only the minitoc section entries in bold while the ToC entries are in the normal font, put the incantation between the \tableofcontents command and the first \chapter command.

As tocloft is a very powerful and useful package, these cures are worth to be added if you need the benefits of this package. See also section 2.22.

2.22 Use with the memoir class

The memoir class [479, 481, 482] offers basically the functionnalities of the appendix, tocbibind and tocloft packages (this class and these packages have the same author, Peter R. Wilson), hence it has the same problems; see above the available solutions (sections 2.20 on the preceding page, 1.5.5 on page 50, and 2.21 on the preceding page respectively). If your version of the memoir class is recent, the syntax of the \chapter command is different and the memoir class could be no more compatible with the minitoc package, but a patch is inserted to fix the problem. Hopefully, if your version of the memoir class is more recent than 2005/09/25, the patch is no more necessary.

If you are using the memoir class (or the tocloft package), the \mtcsetfont command has no effect (\mtcsettitlefont works); you should use the font commands which are specific of the memoir class (or of the tocloft package).
If you still want to use the \mtcsetfont commands while using the \texttt{memoir} class (or of the \texttt{tocloft} package), you must disable the \texttt{memoir/tocloft} font commands. This is done by the following commands:

\begin{verbatim}
\let\cftpartfont\relax
\let\cftchapterfont\relax
\let\cftsectionfont\relax
\let\cftsubsectionfont\relax
\let\cftsubsubsectionfont\relax
\let\cftparagraphfont\relax
\let\cftsubparagraphfont\relax
\let\cftfigurefont\relax
\let\cftsubfigurefont\relax
\let\cfttablefont\relax
\let\cftsubtablefont\relax
\end{verbatim}

2.23 There are too many commands for fonts, titles, and depths

\texttt{\mtcsetfont} Since version #41, the \texttt{\mtcsetfont} and \texttt{\mtcsettitlefont} commands are available. You do not need anymore to know \texttt{\mtcSSSfont}, \texttt{\ptifont}, etc.

\texttt{\mtcsettitle} Since version #42, the \texttt{\mtcsettitle} command is available. You do not need anymore to know \texttt{\mtctitle}, \texttt{\sltttitle}, etc.

\texttt{\mtcsetdepth} Since version #43, the \texttt{\mtcsetdepth} command is available. You do not need anymore to know the counters \texttt{\minitocdepth}, \texttt{\sectlotdepth}, etc.

2.24 Compatibility with the \texttt{AMS} document classes

This problem has been pointed out by Henri Massias.

\texttt{\mtcaddchapter} Unfortunately, the \texttt{amsart} and \texttt{amsproc} document classes are \textit{incompatible} with \texttt{minitoc}. The \texttt{amsbook} document class requires the insertion of commands if you want a list of figures and/or a list of tables:

\begin{verbatim}
\listoffigures
\mtcaddchapter % added
\listoftables
\mtcaddchapter % added
\end{verbatim}
2.25 Hiding some entries from the main table of contents

It is a problem similar to that of section 2.19 on page 63. An example is having a local table of contents for a chapter (\minitoc) whose entries should not appear in the main table of contents. Just use the \mtchideinmaintoc environment:

```
\chapter{Title}
\begin{mtchideinmaintoc}[level]
\minitoc
\section{sub-title}
...
\end{mtchideinmaintoc}
```

This environment accepts an optional numeric argument, which is the depth of hiding in the main toc (default: -1, complete hiding). You can look at the mtc-apx.tex example file:
Of course, the environments \texttt{mtchideinmainlof} and \texttt{mtchideinmainlot} are also available, to hide some entries in the main list of figures or of tables.

Note that the position of the end of these environments must be adjusted to include a page break (like the one done by a \texttt{chapter} command), else the restore command might be inserted too early into the \texttt{.toc}, \texttt{.lof} or \texttt{.lot} file. There is an example file (\texttt{mtc-hi1.tex}):

\begin{verbatim}
\documentclass{report}
\ProvidesFile{mtc-hi1.tex} \[2007/01/04\]
\usepackage{minitoc}
\begin{document}
\dominilof \listoffigures
\dominilot \listoftables
\chapter{First}
\begin{table}
\caption{TAAAA1}
\end{table}
\begin{table}
\caption{TAAAA2}
\end{table}
\chapter{Second}
\begin{table}
\caption{TBBBB1}
\end{table}
\begin{table}
\caption{TBBBB2}
\end{table}
\chapter{Third}
\end{document}
\end{verbatim}

We begin the hiding of figure entries in the list of figures and of table entries in the list of tables. In this document, we use the environment forms.

\begin{verbatim}
\begin{mtchideinmainlof}
\begin{mtchideinmainlot}
\begin{figure}
\caption{BBBB1}
\end{figure}
\begin{figure}
\caption{BBBB2}
\end{figure}
\begin{table}
\caption{TBBBB1}
\end{table}
\begin{table}
\caption{TBBBB2}
\end{table}
\end{mtchideinmainlot}
\end{mtchideinmainlof}
\end{verbatim}

We terminate the hiding of figure entries in the list of figures and of table entries in the list of tables. In this document, we use the environment forms.

\begin{verbatim}
\begin{mtchideinmainlot}
\begin{figure}
\caption{CCCC1}
\end{verbatim}

\begin{verbatim}
\begin{figure}
\caption{AAAA1}
\end{verbatim}

\begin{verbatim}
\begin{table}
\caption{TAAAA1}
\end{table}
\begin{table}
\caption{TAAAA2}
\end{table}
\begin{table}
\caption{TBBBB1}
\end{table}
\begin{table}
\caption{TBBBB2}
\end{table}
\begin{table}
\caption{TBBBB3}
\end{table}
\begin{table}
\caption{TBBBB4}
\end{table}
\end{document}
\end{verbatim}
But it is also possible to use commands in place of these environments: you place a \mtchideinmainlof (or \mtchideinmainlot) command in the first figure (or table) to hide, before its caption and a \endmtchideinmainlof (or \endmtchideinmainlot) command at the end of the last figure (or table) to hide, after its caption, like in this example file (mtc-hi2.tex):

\begin{figure}
\begin{table}
\caption{AAAA1}
\caption{TAAAA1}
\caption{AAAA2}
\caption{TAAAA2}
\caption{TBBBB1}
\caption{TBBBB2}
\caption{BBBB1}
\caption{BBBB2}
\caption{TBBBB1}
\caption{TBBBB2}
\caption{CCCC1}
\caption{CCCC2}
\caption{TCCCC1}
\caption{TCCCC2}
\end{figure}
\end{table}
\end{document}
This method, recommended while more delicate to apply, is much more reliable in delimiting the hiding domain: it solves the problem of the asynchronism between the writing of floats and the writing of the normal text.

### 2.26 Defining your own `.mld` file

First, you should not directly modify one of the distributed `.mld` and `.mlo` files. The simplest way to alter some title is to redefine the corresponding command via `\renewcommand` or better via `\mtcsettitle`. If you really want to have your own `.mld` file, you copy an existing `.mld` file into one with a new name (not the name of a distributed `.mld` file). Then you modify this new `.mld` file and you can use it via `\mtcselectlanguage`. You can always contact me to add this new `.mld` file to the distribution. These remarks apply also to the `language[.mld–.mlo]` pairs of language definition files.

### 2.27 Use with the `abstract` package

If the `abstract` package [470] (by Peter R. Wilson), is used with its `addtotoc` option, a “Abstract” entry is added to the table of contents, as a starred chapter if the document class defines `\chapter`, else as a starred section. This problem is detected by the `hints` option and you should add a `\mtcaddchapter[]` or a `\mtcaddsection[]` command after your `abstract` environment.

### 2.28 Use with the `sectsty` package

If the `sectsty` package [319] (by Rowland McDonnell) is used, it must be loaded *before* the `minitoc` package, because it alters (redefines) the sectioning commands. Of course, the `hints` option detects this problem.
2.29 Strange alignment in the minitocs

In minitocs, subsection titles are not aligned with sections, as they are in the main table of contents.

The entries of a table of contents are formatted via internal commands like \@part, \@chapter, \@section, etc.

The “part” and “chapter” levels (and “section” for an article) use specific commands which are somewhat complex for a more elaborated formatting. For the “section” (in the report and book classes) and lower levels, these commands are (book class, book.cls) by default:

\renewcommand\l@section{\@dottedtocline{1}{1.5em}{2.3em}}
\renewcommand\l@subsection{\@dottedtocline{2}{3.8em}{3.2em}}
\renewcommand\l@subsubsection{\@dottedtocline{3}{7.0em}{4.1em}}
\renewcommand\l@paragraph{\@dottedtocline{4}{10em}{5em}}
\renewcommand\l@subparagraph{\@dottedtocline{5}{12em}{6em}}

which will be applied in the main table of contents and in the minitocs. The arguments of \@dottedtocline are

1) the logical depth (which will be compared to tocdepth or minitocdepth).
2) the indentation.
3) the width reserved for the section/subsection/... number.

In the standard book, report and article classes [282], the dimensions (second and third arguments) are given in “em” units, and this unit depends on the current font. In the main table of contents, the section and subsection entries are written in the same font, hence usually the alignment is correct. But in the minitocs, the section entries are written in a bold font while the subsection entries are written in a non bold font (the default font choices are given in table 1.6 on page 37), hence one “em” has different sizes in these two fonts and the alignment is changed.

There are several solutions:

- Redefine the \l@section ... \l@subparagraph commands to use font independent units (pt, mm, pc, etc.). This redefinition must be performed in a package or via a command defined by a package or between \makeatletter and \makeatother, because these commands have a \@ in their names; you must use \renewcommand* to redefine these commands.

- Use the tocloft package [469] to change the indentation, with font independent units. But then see also section 2.21 on page 64.
• Use the same font for the section and subsection entries in the minitocs, using the \mtcsetfont command (see section 1.4.9 on page 41) or redefining the \mtcSfont, \mtcSSfont, \mtcSSSfont, \mtcPfont and \mtcSPfont commands (see table 1.6 on page 37), or similar.

2.30 Useful precautions with starred sectionning commands

• The headers are not modified by \part*, \chapter* or \section*: it is necessary to use \markboth or \markright to get correct page headers for the current and following pages.

• If you need an entry in the table of contents for a \chapter* or a \section* command, you must use \mtcaddchapter[title] or \mtcaddsection[title] after the starred sectionning command. If you need an entry in the table of contents for a \part* command, the page number in the table of contents would be wrong, because \part* implies a \clearpage or a \cleardoublepage before the first page of the part. Use the sequence

\cleardoublepage % \clearpage if openany option.
\mtcaddpart[title]
\part*[title]

2.31 Use with packages for captions

If one of the caption [421, 422, 424], caption2\textsuperscript{7} [423], (both written by Axel Sommerfeldt), ccaption [474] (by Peter R. Wilson), or mcaption [228] (by Stephan Hennig), packages is used, it must be loaded before the minitoc package, because such packages alter (redefine) the commands listing figures and tables. Of course, the hints option detects this problem.

2.32 Bad interaction minitoc/hyperref/memoir

When the minitoc and hyperref [390] packages are used in a document of class memoir [479, 481, 482], the chapter header “Chapter” does not appear on the first page of the chapter.

This problem is fixed in version #44 of minitoc.

\textsuperscript{7} This package is obsolete; now use a recent version of the caption package.
2.33 Use with the varsects package

If the varsects package [437] (by Daniel Taupin) is used, it must be loaded before the minitoc package, because it alters (redefines) the sectioning commands. Of course, the hints option detects this problem.

2.34 Initial font settings

The setting of the fonts in the mini-tables is a rather complex problem. If we take the parttocs as an example, there is a \ptcfont font-command which is used for two purposes:

- First, to be used as default value for some other font-commands (like \ptcSPfont). As its default value is used in the initialization of the minitoc package, the value of these other commands is not altered if you modify \ptcfont. You must modify these commands one at a time.

- Second, it is invoked at the beginning of each parttoc, partlof or partlot to set an initial font command. Then each entry of the mini-table calls its own font command (like \ptcSPfont). Thus, if you modify \ptcfont, you can obtain a global effect on the fonts in the parttocs, partlofs, and partlots. So you can play with the various parameters of the fonts (family, shape, series, size), if you want fancy mini-tables; but it is rather difficult.

In the initialization of the minitoc package, we have a sequence of commands:

\let\ptcSSfont\ptcfont % (subsections)
\let\ptcSSSfont\ptcfont % (subsubsections)
\let\ptcPfont\ptcfont % (paragraphs)
\let\ptcSPfont\ptcfont % (subparagraphs)
\let\plffont\ptcfont % (figures)
\let\plSfont\ptcfont % (subfigures)
\let\pltfont\ptcfont % (tables)
\let\pltSfont\ptcfont % (subtables)

8 The same remarks apply to the other mini-tables.
Note that if you say:
\let\ptcSSfont\ptcfont % (subsections)
\let\ptcSSSfont\ptcfont % (subsubsections)
\let\ptcPfont\ptcfont % (paragraphs)
\let\ptcSPfont\ptcfont % (subparagraphs)
\let\plffont\ptcfont % (figures)
\let\plfSfont\ptcfont % (subfigures)
\let\pltfont\ptcfont % (tables)
\let\pltSfont\ptcfont % (subtables)

after loading the minitoc package, these font commands will be “associated” to \ptcfont, hence if you modify \ptcfont (by via \mtcsetfont{parttoc}{*}{...} or \renewcommand), they will follow the modification. But if you modify one of these commands via \renewcommand or \mtcsetfont{parttoc}{subsection}{...} (subsection is an example), the association is broken. But you could be more clever by saying something like
\mtcsetfont{parttoc}{subsection}{\ptcfont\itshape}

to preserve the association and modify only some parameters of a minitoc font command.

For levels above subsection (part, chapter and section), the fonts a more specific in general, but you can, of course, say something like \def\ptcCfont{\ptcfont} to make a similar association. You can even make other associations, like this:

% for high sectionning levels:
\def\highlevelsfont{\rmfamily\bfseries\normalsize\upshape}
% for low sectionning levels:
\def\lowlevelsfont{\rmfamily\mdseries\smallsize\upshape}
% then for each level:
\def\ptcCfont{\highlevelsfont}
\def\ptcsfont{\highlevelsfont}
\def\ptcSfont{\lowlevelsfont}
\def\ptcSSfont{\lowlevelsfont}
\def\ptcPfont{\lowlevelsfont\itshape}
\def\ptcSPfont{\lowlevelsfont\itshape}

Then you can redefine \highlevelsfont or \lowlevelsfont to act on several fonts in one step, but you must use \renewcommand. You cannot act on \highlevelsfont or \lowlevelsfont with \mtcsetfont.

Note that only the fonts for parttoc are used in the examples above; but, of course, the situation is the same for minitocs and secttoc. \highlevelsfont and \lowlevelsfont are macro names that you can choice, they are not part of the minitoc package.
2.35 Use with the KOMA-Script classes

If a KOMA-Script class [343, 344, 399], compatible with minitoc (scrbook, scrreprt or scrartcl), is used, some class options may cause problems with the minitoc package, because these options add chapter or section entries in the table of contents. See section 1.5.5 on page 50. Of course, the hints option detects this problem.

2.36 Use with the jura class or the alphanum package

The jura class loads the alphanum package, which redefines the sectioning structure in a non-standard way, after the loading of the report class. This class and this package are incompatible with minitoc.

2.37 The .mld files and the babel package

If you are using the babel package [60, 61], you can automatize the loading of the .mld file by adding some code in the preamble of your document, like this:

\AtBeginDocument{%
  \addto\captions{\textit{language1}}{\mtcselectlanguage{language2}}}

where language1 is the language name for babel and language2 the language name for minitoc; there are often identical, but there are exceptions (when you use a locally customized .mld file, for instance).

2.38 Use with the fncychap package

If the fncychap package [301] (by Ulf A. Lindgren) is used, it must be loaded before the minitoc package, because it alters (redefines) the sectioning commands. Of course, the hints option detects this problem.

2.39 Use with the quotchap package

If the quotchap package [442] (by Karsten Tinnefeld) is used, it must be loaded before the minitoc package, because it alters (redefines) the sectioning commands. Of course, the hints option detects this problem.
2.40 Use with the \texttt{romannum} package

If the \texttt{romannum} package [480] (by Peter R. Wilson) is used, it must be loaded \texttt{before} the \texttt{minitoc} package, because it alters (redefines) the numbering of the sectionning commands. Of course, the \texttt{hints} option detects this problem.

2.41 Use with the \texttt{sfheaders} package

If the \texttt{sfheaders} package [304] (by Maurizio Loreti) is used, it must be loaded \texttt{before} the \texttt{minitoc} package, because it alters (redefines) the sectionning commands. Of course, the \texttt{hints} option detects this problem.

2.42 Use with the \texttt{alnumsec} package

If the \texttt{alnumsec} package [274] (by Frank Küster) is used, it must be loaded \texttt{before} the \texttt{minitoc} package, because it alters (redefines) the numbering of the sectionning commands. Of course, the \texttt{hints} option detects this problem.

2.43 Use with the \texttt{captcont} package

If the \texttt{captcont} package [131] (by Steven Douglas Cochran) is used, it must be loaded \texttt{before} the \texttt{minitoc} package, because it alters (redefines) the caption commands. Of course, the \texttt{hints} option detects this problem.

2.44 Vertical spaces (gaps) for parttocs, partlofs, and partlots titles

These vertical gaps were hard-coded like for the chapter heads in the \texttt{book} and \texttt{report} document classes. The values were 50pt and 40pt, but some users want to adjust them for the titles of the part-level mini-tables. Since version #45, these gaps are defined by \texttt{\mtcgapbeforeheads} and \texttt{\mtcgapafterheads}, with these defaults values. These commands apply globally to parttocs, partlofs and partlots. They are \texttt{commands}, not \texttt{dimensions}, so they must be modified via \texttt{\renewcommand} (but \texttt{not} via \texttt{\setlength}). An example of use is given in the \texttt{mtc-gap.tex} document file:

\begin{verbatim}
160 \documentclass[a4paper,oneside,12pt]{book}
161 \ProvidesFile{mtc-gap.tex}[2007/01/04]
162 \renewcommand{\mtcgapbeforeheads}{15mm}
163 \renewcommand{\mtcgapafterheads}{10mm}
\end{verbatim}
We use the vruler package (by Zhuhan Jiang) to display a vertical ruler showing the position of the titles:

\usepackage{txfonts,vruler} % vertical graduation to note positions (Zhuhan Jiang)
\usepackage[english2,tight,listfiles]{minitoc}
\begin{document}
\setvruler[1cm][0][10][3][0][0pt][0pt][0pt][] % with vruler package
\doparttoc \faketableofcontents
\part{First part}
A normal parttoc, with the normal gaps before and after it.
\parttoc
\chapter{First chapter of first part} \chapter{Second chapter of first part}
\part{Second part}
\mtcgapbeforeheads \mtcgapafterheads
We set large gaps. Note the new position of the parttoc.
\renewcommand{\mtcgapbeforeheads}{100pt}
\renewcommand{\mtcgapafterheads}{80pt}
\parttoc
\chapter{First chapter of second part} \chapter{Second chapter of second part}
\part{Third part}
\mtcgapbeforeheads \mtcgapafterheads
We set small gaps. Note the new position of the parttoc.
\renewcommand{\mtcgapbeforeheads}{20pt}
\renewcommand{\mtcgapafterheads}{10pt}
\parttoc
\chapter{First chapter of third part} \chapter{Second chapter of third part}
\end{document}
\langle /mtc-gap \rangle

2.45 Vertical spacing before the bottom rule of a minitable

The little spacing between a minitable and its bottom rule is implemented as a vertical kern that should be sufficient to allow the descending parts of the letters of the last entry of the minitable. The values should depend on the line spacing and of the font size. They are defined as macros that you can adjust by redefining them via \renewcommand. The (empirical) default values are given in table 2.1 on the next page.
Table 2.1: Kernings before minitable bottom rules

<table>
<thead>
<tr>
<th>Command</th>
<th>Default value</th>
</tr>
</thead>
<tbody>
<tr>
<td>\kernafterparttoc</td>
<td>\kern-1.\baselineskip\kern.5ex</td>
</tr>
<tr>
<td>\kernafterpartlof</td>
<td>\kern-1.\baselineskip\kern.5ex</td>
</tr>
<tr>
<td>\kernafterpartlot</td>
<td>\kern-1.\baselineskip\kern.5ex</td>
</tr>
<tr>
<td>\kernafterminitoc</td>
<td>\kern-.5\baselineskip\kern.5ex</td>
</tr>
<tr>
<td>\kernafterminilof</td>
<td>\kern-1.\baselineskip\kern0.ex</td>
</tr>
<tr>
<td>\kernafterminilot</td>
<td>\kern-1.\baselineskip\kern0.ex</td>
</tr>
<tr>
<td>\kernafterseccntoc</td>
<td>\kern-1.\baselineskip\kern.5ex</td>
</tr>
<tr>
<td>\kernafterseclot</td>
<td>\kern-1.\baselineskip\kern.5ex</td>
</tr>
<tr>
<td>\kernaftersectlot</td>
<td>\kern-1.\baselineskip\kern.5ex</td>
</tr>
</tbody>
</table>

2.46 Another interaction between the tocloft and minitoc packages

I encountered an interaction between tocloft and minitoc. I want to force minitoc to not display the page numbers, but because of tocloft it doesn’t. Here is an example code:

```latex
\documentclass[12pt,a4paper]{book}
\usepackage{tocloft}
\usepackage{minitoc}
\begin{document}
\frontmatter
\dominitoc\tableofcontents
\mainmatter
\chapter{Chapter}
\section{Section A} \section{Section B}
\chapter{Second Chapter}
\mtcsetfont{minitoc}{section}{\normalfont\small}
\mtcsetpagenumbers{minitoc}{off}
\minitoc
\section{Section A} \section{Section B}
\end{document}
```

If I comment the line loading the tocloft package, I will get a minitoc without page numbers as I wanted.

When using together tocloft and minitoc, the tocloft package must be loaded first, and its commands take precedence to format the entries in the TOC (and in minitocs). To suppress the page numbers, you should try the \cftpagenumbersoff{XXX} command (from tocloft), which is described in the tocloft.pdf documentation [469, pages 45-56]; XXX is the level of entry (chapter, sec, subsec, etc.). There are similar remarks about font related commands.
The `tocloft` package is more specialized in that job than `minitoc`, so if it is loaded, `minitoc` uses the `tocloft` tools. There is the corrected example (`mtc-tlo.tex`):

```
\documentclass[12pt,a4paper]{book}
\ProvidesFile{mtc-tlo.tex}[2007/06/13]%

We must load `tocloft` before `minitoc`:

\usepackage{tocloft}
\usepackage[tight]{minitoc}
\begin{document}
\mtcsetfont
We define the global font for the minitoc entries:

\mtcsetfont{minitoc}{*}{\normalfont\small}
\frontmatter
\dominitoc \tableofcontents
\cftpagenumbersoff \cftsecfont
For the section entries in the minitocs, we suppress the page numbers and change the font by using commands from the `tocloft` package:

\cftpagenumbersoff{sec}
\renewcommand{\cftsecfont}{\normalfont\small}
\mainmatter
\chapter{First Chapter} \minitoc
\section{Section A} \section{Section B}
\chapter{Second Chapter} \minitoc
\section{Section A} \section{Section B}
\end{document}
```

2.47 Use with the `hangcaption` package

If the `hangcaption` package [250] (by David M. Jones) is used, it must be loaded before the `minitoc` package, because it alters (redefines) the sectioning commands. Of course, the `hints` option detects this problem.

2.48 Use with the `flowfram` package

The `flowfram` package [433, 434], which has its own system of minitocs, is hence incompatible with `minitoc`. 
Chapter 3

Memento

Tables

<table>
<thead>
<tr>
<th>Commands</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1 Package options</td>
<td>80</td>
</tr>
<tr>
<td>3.2 General commands</td>
<td>81</td>
</tr>
<tr>
<td>3.3 Part level commands</td>
<td>82</td>
</tr>
<tr>
<td>3.4 Chapter level commands</td>
<td>83</td>
</tr>
<tr>
<td>3.5 Section level commands</td>
<td>84</td>
</tr>
<tr>
<td>3.6 Commands for horizontal rules</td>
<td>85</td>
</tr>
<tr>
<td>3.7 Commands for page numbers</td>
<td>85</td>
</tr>
<tr>
<td>3.8 Commands for mini-tables features</td>
<td>86</td>
</tr>
<tr>
<td>3.9 Preparation and insertion commands</td>
<td>87</td>
</tr>
<tr>
<td>3.10 Adjustment commands</td>
<td>87</td>
</tr>
<tr>
<td>3.11 Classes and packages needing some</td>
<td></td>
</tr>
<tr>
<td>precautions with minitoc</td>
<td>88</td>
</tr>
<tr>
<td>3.12 Checking if inside a minitable</td>
<td>89</td>
</tr>
<tr>
<td>3.13 Commands for polymorphic entries</td>
<td>89</td>
</tr>
<tr>
<td>3.14 Obsolete commands</td>
<td>89</td>
</tr>
<tr>
<td>3.15 Classes and packages needing some</td>
<td></td>
</tr>
<tr>
<td>precautions with minitoc</td>
<td>88</td>
</tr>
<tr>
<td>3.16 Checking if inside a minitable</td>
<td>89</td>
</tr>
<tr>
<td>3.17 Commands for polymorphic entries</td>
<td>89</td>
</tr>
<tr>
<td>3.18 Obsolete commands</td>
<td>89</td>
</tr>
</tbody>
</table>

Table 3.1: Package options

<table>
<thead>
<tr>
<th>Options</th>
<th>Default</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>shortext</td>
<td><em>NO</em></td>
<td>Short extensions for auxiliary files.</td>
</tr>
<tr>
<td>loose, tight</td>
<td>loose</td>
<td>Spacing of lines in mini-tables.</td>
</tr>
<tr>
<td>k-loose, k-tight</td>
<td>k-loose</td>
<td>Spacing of lines in mini-tables (KOMA-Script classes).</td>
</tr>
<tr>
<td>dotted, undotted</td>
<td>dotted</td>
<td>Presence of leaders (dotted lines).</td>
</tr>
<tr>
<td>insetion</td>
<td><em>NO</em></td>
<td>Keeps floats (figures and tables) from drifting outside of their section.</td>
</tr>
<tr>
<td>notoccite</td>
<td><em>NO</em></td>
<td>Useful if you have \cite commands in sectionning titles and use an unsorted bibliographic style.</td>
</tr>
<tr>
<td>listfiles, nolistfiles</td>
<td>listfiles</td>
<td>Lists the minitoc auxiliary files into document.maf.</td>
</tr>
<tr>
<td>hints, nohints</td>
<td>hints</td>
<td>Adds hints in the document.log file. Useful to detect some problems. Option nohints is inadvisable.</td>
</tr>
</tbody>
</table>

Language options are listed in table 1.7 on page 38. Default: english.
Table 3.2: General commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>\faketableofcontents</td>
<td>Replaces \tableofcontents if you want mini-tables of contents but no main table of contents.</td>
</tr>
<tr>
<td>\fakelistoffigures</td>
<td>Replaces \listoffigures if you want mini-lists of figures but no main list of figures.</td>
</tr>
<tr>
<td>\fakelistoftables</td>
<td>Replaces \listoftables if you want mini-lists of tables but no main list of tables.</td>
</tr>
<tr>
<td>\mtcselectlanguage{language}</td>
<td>Loads language.mld to select a language for mini-tables titles.</td>
</tr>
<tr>
<td>\mtcsetdepth{mini-table}{depth}</td>
<td>Changes the depth for some mini-tables.</td>
</tr>
<tr>
<td>\mtcsetoffset{mini-table}{offset}</td>
<td>Changes the offset for some mini-tables.</td>
</tr>
<tr>
<td>\mtcsetfeature{mini-table}{before</td>
<td>after</td>
</tr>
<tr>
<td>\mtcsetfont{mini-table}{sectionning-level}{font commands}</td>
<td>Redefines a minitoc font command.</td>
</tr>
<tr>
<td>\mtcsetformat{mini-table}{dotinterval</td>
<td>pagenumwidth</td>
</tr>
<tr>
<td>\mtcsetpagenumbers{mini-table</td>
<td>*}{on</td>
</tr>
<tr>
<td>\mtcsetrules{mini-table</td>
<td>*}{on</td>
</tr>
<tr>
<td>\mtcsettitle{mini-table}{title string}</td>
<td>Changes the title for some mini-tables.</td>
</tr>
<tr>
<td>\mtcsettitlefont{mini-table}{font commands}</td>
<td>Changes the font of the title for some mini-tables.</td>
</tr>
<tr>
<td>\mtcskip</td>
<td>To add a vertical skip between the mini-tables.</td>
</tr>
<tr>
<td>\mtcskipamount</td>
<td>Length of \mtcskip. Default: \bigskipamount.</td>
</tr>
<tr>
<td>\tightmtcfalse</td>
<td>Loose mini-tables. Default.</td>
</tr>
<tr>
<td>\tightmtctrue</td>
<td>Tight mini-tables.</td>
</tr>
<tr>
<td>\ktightmtcfalse</td>
<td>Loose mini-tables. Default. (KOMA-Script classes).</td>
</tr>
<tr>
<td>\ktightmtctrue</td>
<td>Tight mini-tables. (KOMA-Script classes).</td>
</tr>
<tr>
<td>\undottedmtcfalse</td>
<td>Dotted lines in mini-tables (from entry to page number). Default.</td>
</tr>
<tr>
<td>\undottedmtctrue</td>
<td>No dotted lines in mini-tables (from entry to page number).</td>
</tr>
</tbody>
</table>
Table 3.3: Part level commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>\doparttoc[x]</td>
<td>Before {fake}tableofcontents if you use \parttoc.</td>
</tr>
<tr>
<td>\dopartlof[x]</td>
<td>Before {fake}listoffigures if you use partlof.</td>
</tr>
<tr>
<td>\dopartlot[x]</td>
<td>Before {fake}listoftables if you use partlot.</td>
</tr>
<tr>
<td>\parttoc[x]</td>
<td>After each \part command for which a parttoc is needed.</td>
</tr>
<tr>
<td>\partlof[x]</td>
<td>After each \part command for which a partlof is needed.</td>
</tr>
<tr>
<td>\partlot[x]</td>
<td>After each \part command for which a partlot is needed.</td>
</tr>
<tr>
<td>\setcounter{parttocdepth}{depth}</td>
<td>Depth of the following parttocs. Analog to tocdepth. Default: 2.</td>
</tr>
<tr>
<td></td>
<td>Has no action on partlofs and partlots.</td>
</tr>
<tr>
<td></td>
<td>or:</td>
</tr>
<tr>
<td>\mtcsetdepth{parttoc</td>
<td>partlof</td>
</tr>
<tr>
<td>\ptcindent</td>
<td>Left/right indentation of a partial table. Default: 24pt.</td>
</tr>
<tr>
<td>\mtcsetoffset{parttoc</td>
<td>partlof</td>
</tr>
<tr>
<td>\ptcfont</td>
<td>Font command for parttoc. Default: \small\rmfamily\upshape\mdseries (article) or: \normalsize\rmfamily\upshape\mdseries (book, report).</td>
</tr>
<tr>
<td>\ptcCfont</td>
<td>Font command for parttoc, chapter entries. Default: \small\rmfamily\upshape\bseries.</td>
</tr>
<tr>
<td>\ptcsfont</td>
<td>Font command for parttoc, section entries. Default: \small\rmfamily\upshape\bseries (article) or: \small\rmfamily\upshape\bseries (book, report).</td>
</tr>
<tr>
<td>\ptcssfont</td>
<td>Font command for parttoc, subsection entries.</td>
</tr>
<tr>
<td>\ptcPfont</td>
<td>Font command for parttoc, paragraph entries.</td>
</tr>
<tr>
<td>\ptcSfFont</td>
<td>Font command for parttoc, subparagraph entries.</td>
</tr>
<tr>
<td>\ptlfonFont</td>
<td>Font for partlof. Default: \small\rmfamily\upshape\mdseries.</td>
</tr>
<tr>
<td>\pltfSfont</td>
<td>Font for partlof (subfigures). Default: \small\rmfamily\upshape\mdseries.</td>
</tr>
<tr>
<td>\ptcttitle</td>
<td>Title of parttocs. Default: Table of Contents.</td>
</tr>
<tr>
<td>\ptlttitle</td>
<td>Title of partlofs. Default: List of Figures.</td>
</tr>
<tr>
<td>\ptltitle</td>
<td>Title of partlots. Default: List of Tables.</td>
</tr>
<tr>
<td>\pttitfont</td>
<td>Font for partXXX titles. Default: \Large\rmfamily\upshape\bseries (article) or: \LARGE\rmfamily\upshape\bseries (book, report).</td>
</tr>
<tr>
<td>\mtcgapbeforeheads</td>
<td>Vertical gap before part-level mini-tables titles. Default: 50pt</td>
</tr>
<tr>
<td>\mtcgapafterheads</td>
<td>Vertical gap after part-level mini-tables titles. Default: 40pt</td>
</tr>
</tbody>
</table>

*: \[x\] is an optional argument to set the position of the title; the setting is local for the \partXXX commands, global for the \dopartXXX commands. The values of \[x\] are: \[l\] for left (default), \[c\] for centered, \[r\] for right, \[n\] or \[e\] for no title.

**: defaults like \ptcfont.
Table 3.4: Chapter level commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>\dominitoc[$x$]</td>
<td>Before {fake}tableofcontents if you use \minitoc.</td>
</tr>
<tr>
<td>\dominilof[$x$]</td>
<td>Before {fake}listoffigures if you use \minilof.</td>
</tr>
<tr>
<td>\dominilot[$x$]</td>
<td>Before {fake}listoftables if you use \minilot.</td>
</tr>
<tr>
<td>\minitoc[$x$]</td>
<td>After each \chapter command for which a minitoc is needed.</td>
</tr>
<tr>
<td>\minilof[$x$]</td>
<td>After each \chapter command for which a minilof is needed.</td>
</tr>
<tr>
<td>\minilot[$x$]</td>
<td>After each \chapter command for which a minilot is needed.</td>
</tr>
</tbody>
</table>

\setcounter{minitocdepth}{\depth}  
Depth of the following minitocs. Analog to \tocdepth. Default: 2. Has no action on minilofs and minilots.

or:
\mtcsetdepth{minitoc|minilof|minilot}{\depth}
Idem, but can also act on minilofs and minilots.

\mtcindent  

\mlfoffset  

\mltoffset  

\mtcsetoffset{minitoc|minilof|minilot}{\offset}
Idem, but can also act on minilofs and minilots.

\mtcfont  
Font command for minitoc.  
Default: \small\rmfamily\upshape\mdseries.

\mtcSfont  
Font command for minitoc, section entries.  
Default: \small\rmfamily\upshape\bfseries.

\mtcSSfont  
Font command for minitoc, subsection entries.  
Default: \small\rmfamily\upshape\bfseries.

\mtcSSSfont  
Font command for minitoc, subsubsection entries.  
Default: \small\rmfamily\upshape\bfseries.

\mlffont  
Font for minilof. Default: \small\rmfamily\upshape\mdseries.

\mlfSfont  
Font for minilof (subfigures). Default: \small\rmfamily\upshape\mdseries.

\mltfont  
Font for minilot. Default: \small\rmfamily\upshape\mdseries.

\mltSfont  
Font for minilot (subtables). Default: \small\rmfamily\upshape\mdseries.

\mtctitle  
Title of minitocs. Default: Contents.

\mlftitle  
Title of minilofs. Default: Figures.

\mltttitle  
Title of minilots. Default: Tables.

\mtifont  
Font for miniXXX titles.  
Default: \large\rmfamily\upshape\bfseries.

\*: [$x$] is an optional argument to set the position of the title; the setting is local for the miniXXX commands, global for the dominiXXX commands. The values of $x$ are: l for left (default), c for centered, r for right, n or e for no title.

**: defaults like \mtcfont.
Table 3.5: Section level commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Meaning</th>
</tr>
</thead>
</table>
| \dosecttoc\[x\] | Before \texttt{\{fake\}tableofcontents} if you use \texttt{\secttoc}.
| \dosectlof\[x\] | Before \texttt{\{fake\}listoffigures} if you use \texttt{\sectlof}.
| \dosectlot\[x\] | Before \texttt{\{fake\}listoftables} if you use \texttt{\sectlot}.
| \secttoc\[x\] | After each \texttt{\section} command for which a secttoc is needed.
| \sectlof\[x\] | After each \texttt{\section} command for which a sectlof is needed.
| \sectlot\[x\] | After each \texttt{\section} command for which a sectlot is needed.

\setcounter{secttocdepth}{\depth} Depth of the following secttocs. Analog to \texttt{tocdepth}. Default: 2. Has no action on sectlofs and sectlots.

or:
\mtcsetdepth{\secttoc|\sectlof|\sectlot}\{\depth\} Idem, but can also act on sectlofs and sectlots.


\mtcsetoffset{\secttoc|\sectlof|\sectlot}\{\offset\} Idem, but can also act on sectlofs and sectlots.

\stcfont Font command for secttoc. Default: \texttt{\small\rmfamily\upshape\mdseries}.
\stcSSfont Font command for secttoc, subsection entries.
\stcSSSfont Font command for secttoc, subsubsection entries.
\stcPfont Font command for secttoc, paragraph entries.
\stcSSPfont Font command for secttoc, subparagraph entries.
\slffont Font for sectlof. Default: \texttt{\small\rmfamily\upshape\mdseries}.
\slfSfont Font for sectlof (subfigures). Default: \texttt{\small\rmfamily\upshape\mdseries}.
\sltfont Font for sectlot. Default: \texttt{\small\rmfamily\upshape\mdseries}.
\sltSfont Font for sectlot (subtables). Default: \texttt{\small\rmfamily\upshape\mdseries}.

\stctitle Title of secttocs. Default: Contents.
\slftitle Title of sectlofs. Default: Figures.
\sltttitle Title of sectlots. Default: Tables.
\stifont Font for sectXXX titles. Default: \texttt{\large\rmfamily\upshape\bfseries}.

\*: \[x\] is an optional argument to set the position of the title; the setting is local for the \texttt{\sectXXX} commands, global for the \texttt{\dosectXXX} commands. The values of \(x\) are: \l for left (default), \c for centered, \r for right, \n or \e for no title.

\**: defaults like \texttt{\stcfont}. 
Table 3.6: Commands for horizontal rules

<table>
<thead>
<tr>
<th>Command</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>[no]ptcrule</td>
<td>Activates or inhibits rules in parttocs.</td>
</tr>
<tr>
<td>[no]mtcrule</td>
<td>Activates or inhibits rules in minitocs.</td>
</tr>
<tr>
<td>[no]stcrule</td>
<td>Activates or inhibits rules in secttocs.</td>
</tr>
<tr>
<td>[no]plfrule</td>
<td>Activates or inhibits rules in partlofs.</td>
</tr>
<tr>
<td>[no]mlfrule</td>
<td>Activates or inhibits rules in minilofs.</td>
</tr>
<tr>
<td>[no]slfrule</td>
<td>Activates or inhibits rules in sectlofs.</td>
</tr>
<tr>
<td>[no]pltrule</td>
<td>Activates or inhibits rules in partlots.</td>
</tr>
<tr>
<td>[no]mltrule</td>
<td>Activates or inhibits rules in minilot.</td>
</tr>
<tr>
<td>[no]sltrule</td>
<td>Activates or inhibits rules in sectlots.</td>
</tr>
</tbody>
</table>

\mtcsetrules{\textit{mini-table}[^]{on|off}}
Activates/inhibits horizontal rules in some or all mini-tables.

<table>
<thead>
<tr>
<th>Command</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>\kernafterparttoc</td>
<td>Vertical kerning between a parttoc and its bottom rule.</td>
</tr>
<tr>
<td>\kernafterpartlof</td>
<td>Vertical kerning between a partlof and its bottom rule.</td>
</tr>
<tr>
<td>\kernafterpartlot</td>
<td>Vertical kerning between a partlot and its bottom rule.</td>
</tr>
<tr>
<td>\kernafterminitoc</td>
<td>Vertical kerning between a minitoc and its bottom rule.</td>
</tr>
<tr>
<td>\kernafterminilof</td>
<td>Vertical kerning between a minilof and its bottom rule.</td>
</tr>
<tr>
<td>\kernafterminilot</td>
<td>Vertical kerning between a minilot and its bottom rule.</td>
</tr>
<tr>
<td>\kernaftersecttoc</td>
<td>Vertical kerning between a secttoc and its bottom rule.</td>
</tr>
<tr>
<td>\kernaftersectlof</td>
<td>Vertical kerning between a sectlof and its bottom rule.</td>
</tr>
<tr>
<td>\kernaftersectlot</td>
<td>Vertical kerning between a sectlot and its bottom rule.</td>
</tr>
</tbody>
</table>

By default, parttocs have no rules; minitocs and secttocs have rules. In articles, parttocs have rules.

Table 3.7: Commands for page numbers

<table>
<thead>
<tr>
<th>Command</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>[no]ptcpagenumbers</td>
<td>Activates or inhibits page numbers in parttocs.</td>
</tr>
<tr>
<td>[no]plfpagenumbers</td>
<td>Activates or inhibits page numbers in partlofs.</td>
</tr>
<tr>
<td>[no]pltpagenumbers</td>
<td>Activates or inhibits page numbers in partlots.</td>
</tr>
<tr>
<td>[no]mtcpagenumbers</td>
<td>Activates or inhibits page numbers in minitocs.</td>
</tr>
<tr>
<td>[no]mlfpagenumbers</td>
<td>Activates or inhibits page numbers in minilofs.</td>
</tr>
<tr>
<td>[no]mltpagenumbers</td>
<td>Activates or inhibits page numbers in minilot.</td>
</tr>
<tr>
<td>[no]stcpagenumbers</td>
<td>Activates or inhibits page numbers in secttocs.</td>
</tr>
<tr>
<td>[no]slfpagenumbers</td>
<td>Activates or inhibits page numbers in sectlofs.</td>
</tr>
<tr>
<td>[no]sltpagenumbers</td>
<td>Activates or inhibits page numbers in sectlots.</td>
</tr>
</tbody>
</table>

\mtcsetpagenumbers{\textit{mini-table}[^]{on|off}}
Activates/inhibits page numbers in some or all mini-tables.

By default, the page numbers are present.
Table 3.8: Commands for mini-tables features

<table>
<thead>
<tr>
<th>Command</th>
<th>Default</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>\beforeparttoc</td>
<td>\cleardoublepage</td>
<td>Action before a parttoc.</td>
</tr>
<tr>
<td>\beforepartlof</td>
<td>\cleardoublepage</td>
<td>Action before a partlof.</td>
</tr>
<tr>
<td>\beforepartlot</td>
<td>\cleardoublepage</td>
<td>Action before a partlot.</td>
</tr>
<tr>
<td>\afterparttoc</td>
<td>\cleardoublepage</td>
<td>Action after a parttoc.</td>
</tr>
<tr>
<td>\afterpartlof</td>
<td>\cleardoublepage</td>
<td>Action after a partlof.</td>
</tr>
<tr>
<td>\afterpartlot</td>
<td>\cleardoublepage</td>
<td>Action after a partlot.</td>
</tr>
<tr>
<td>\openparttoc</td>
<td>\cleardoublepage</td>
<td>Action before inserting a parttoc file.</td>
</tr>
<tr>
<td>\openpartlof</td>
<td>\cleardoublepage</td>
<td>Action before inserting a partlof file.</td>
</tr>
<tr>
<td>\openpartlot</td>
<td>\cleardoublepage</td>
<td>Action before inserting a partlot file.</td>
</tr>
<tr>
<td>\closeparttoc</td>
<td>\cleardoublepage</td>
<td>Action after inserting a parttoc file.</td>
</tr>
<tr>
<td>\closepartlof</td>
<td>\cleardoublepage</td>
<td>Action after inserting a partlof file.</td>
</tr>
<tr>
<td>\closepartlot</td>
<td>\cleardoublepage</td>
<td>Action after inserting a partlot file.</td>
</tr>
<tr>
<td>\thispageparttocstyle</td>
<td>\thispageminiloftocstyle</td>
<td>Page style for a parttoc.</td>
</tr>
<tr>
<td>\thispagepartlofstyle</td>
<td>\thispageminilotlofstyle</td>
<td>Page style for a partlof.</td>
</tr>
<tr>
<td>\thispagepartlotstyle</td>
<td>\thispageminilotlotstyle</td>
<td>Page style for a partlot.</td>
</tr>
<tr>
<td>\beforeminitoc</td>
<td>\empty</td>
<td>Action before a minitoc.</td>
</tr>
<tr>
<td>\beforeminiloft</td>
<td>\empty</td>
<td>Action before a miniloft.</td>
</tr>
<tr>
<td>\beforeminilot</td>
<td>\empty</td>
<td>Action before a minilot.</td>
</tr>
<tr>
<td>\afterminitoc</td>
<td>\empty</td>
<td>Action after a minitoc.</td>
</tr>
<tr>
<td>\afterminiloft</td>
<td>\empty</td>
<td>Action after a miniloft.</td>
</tr>
<tr>
<td>\afterminilot</td>
<td>\empty</td>
<td>Action after a minilot.</td>
</tr>
<tr>
<td>\openminitoc</td>
<td>\cleardoublepage</td>
<td>Action before inserting a minitoc file.</td>
</tr>
<tr>
<td>\openminiloft</td>
<td>\cleardoublepage</td>
<td>Action before inserting a miniloft file.</td>
</tr>
<tr>
<td>\openminilot</td>
<td>\cleardoublepage</td>
<td>Action before inserting a minilot file.</td>
</tr>
<tr>
<td>\closeminitoc</td>
<td>\cleardoublepage</td>
<td>Action after inserting a minitoc file.</td>
</tr>
<tr>
<td>\closeminiloft</td>
<td>\cleardoublepage</td>
<td>Action after inserting a miniloft file.</td>
</tr>
<tr>
<td>\closeminilot</td>
<td>\cleardoublepage</td>
<td>Action after inserting a minilot file.</td>
</tr>
<tr>
<td>\thispageminitocstyle</td>
<td>\empty</td>
<td>Page style for a minitoc.</td>
</tr>
<tr>
<td>\thispageminiloftstyle</td>
<td>\empty</td>
<td>Page style for a miniloft.</td>
</tr>
<tr>
<td>\thispageminilotstyle</td>
<td>\empty</td>
<td>Page style for a minilot.</td>
</tr>
<tr>
<td>\beforesecttoc</td>
<td>\empty</td>
<td>Action before a secttoc.</td>
</tr>
<tr>
<td>\beforesectlof</td>
<td>\empty</td>
<td>Action before a sectlof.</td>
</tr>
<tr>
<td>\beforesectlot</td>
<td>\empty</td>
<td>Action before a sectlot.</td>
</tr>
<tr>
<td>\aftersecttoc</td>
<td>\empty</td>
<td>Action after a secttoc.</td>
</tr>
<tr>
<td>\aftersectlof</td>
<td>\empty</td>
<td>Action after a sectlof.</td>
</tr>
<tr>
<td>\aftersectlot</td>
<td>\empty</td>
<td>Action after a sectlot.</td>
</tr>
<tr>
<td>\opensecttoc</td>
<td>\cleardoublepage</td>
<td>Action before inserting a secttoc file.</td>
</tr>
<tr>
<td>\opensectlof</td>
<td>\cleardoublepage</td>
<td>Action before inserting a sectlof file.</td>
</tr>
<tr>
<td>\opensectlot</td>
<td>\cleardoublepage</td>
<td>Action before inserting a sectlot file.</td>
</tr>
<tr>
<td>\closesecttoc</td>
<td>\cleardoublepage</td>
<td>Action after inserting a secttoc file.</td>
</tr>
<tr>
<td>\closesectlof</td>
<td>\cleardoublepage</td>
<td>Action after inserting a sectlof file.</td>
</tr>
<tr>
<td>\closesectlot</td>
<td>\cleardoublepage</td>
<td>Action after inserting a sectlot file.</td>
</tr>
<tr>
<td>\thispagesecttocstyle</td>
<td>\empty</td>
<td>Page style for a secttoc.</td>
</tr>
<tr>
<td>\thispagesectlofstyle</td>
<td>\empty</td>
<td>Page style for a sectlof.</td>
</tr>
<tr>
<td>\thispagesectlotstyle</td>
<td>\empty</td>
<td>Page style for a sectlot.</td>
</tr>
</tbody>
</table>

\mtcsetfeature{mini-table}{before|after|open|close|pagestyle}{commands}

Modifies the features for a mini-table.
Table 3.9: Preparation and insertion commands

<table>
<thead>
<tr>
<th>Type</th>
<th>Phase</th>
<th>Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>table of contents</td>
<td>preparation</td>
<td>part</td>
</tr>
<tr>
<td></td>
<td></td>
<td>chapter</td>
</tr>
<tr>
<td></td>
<td></td>
<td>section</td>
</tr>
<tr>
<td>list of figures</td>
<td>preparation</td>
<td>part</td>
</tr>
<tr>
<td></td>
<td></td>
<td>chapter</td>
</tr>
<tr>
<td></td>
<td></td>
<td>section</td>
</tr>
<tr>
<td>list of tables</td>
<td>preparation</td>
<td>part</td>
</tr>
<tr>
<td></td>
<td></td>
<td>chapter</td>
</tr>
<tr>
<td></td>
<td></td>
<td>section</td>
</tr>
<tr>
<td>all</td>
<td>preparation</td>
<td></td>
</tr>
</tbody>
</table>

Each of these commands accepts one optional argument \texttt{p}, which specifies the position of the title of the mini-table. This argument \texttt{p} has a global effect for the preparation commands, but local for the insertion commands. It is a letter: \[l\] for left aligned (default), \[c\] for centered, \[r\] for right aligned, \[e\] or \[n\] for empty (no title).

Table 3.10: Adjustment commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>\adjustptc[n]</td>
<td>Adjusts (increments) the parttoc counter \texttt{ptc} by \texttt{n}.</td>
</tr>
<tr>
<td>\adjustmtc[n]</td>
<td>Adjusts (increments) the minitoc counter \texttt{mtc} by \texttt{n}.</td>
</tr>
<tr>
<td>\adjuststc[n]</td>
<td>Adjusts (increments) the secttoc counter \texttt{stc} by \texttt{n}.</td>
</tr>
<tr>
<td>\decrementptc</td>
<td>Adjusts (decrements by 1) the parttoc counter \texttt{ptc}.</td>
</tr>
<tr>
<td>\decrementmtc</td>
<td>Adjusts (decrements by 1) the minitoc counter \texttt{mtc}.</td>
</tr>
<tr>
<td>\decrementstc</td>
<td>Adjusts (decrements by 1) the secttoc counter \texttt{stc}.</td>
</tr>
<tr>
<td>\incrementptc</td>
<td>Adjusts (increments by 1) the parttoc counter \texttt{ptc}.</td>
</tr>
<tr>
<td>\incrementmtc</td>
<td>Adjusts (increments by 1) the minitoc counter \texttt{mtc}.</td>
</tr>
<tr>
<td>\incrementstc</td>
<td>Adjusts (increments by 1) the secttoc counter \texttt{stc}.</td>
</tr>
<tr>
<td>\mtcaddpart[title]</td>
<td>Adds the title of a \texttt{part*} in the ToC.</td>
</tr>
<tr>
<td>\mtcaddchapter[title]</td>
<td>Adds the title of a \texttt{chapter*} in the ToC.</td>
</tr>
<tr>
<td>\mtcaddsection[title]</td>
<td>Adds the title of a \texttt{section*} in the ToC.</td>
</tr>
<tr>
<td>\mtcfixglossary[chapter</td>
<td>section</td>
</tr>
<tr>
<td>\mtcfixindex[chapter</td>
<td>section</td>
</tr>
<tr>
<td>\mtcfixnomenclature[chapter</td>
<td>section</td>
</tr>
<tr>
<td>\begin{mtchideinmaintoc}[depth]</td>
<td>\end{mtchideinmaintoc}</td>
</tr>
<tr>
<td>\begin{mtchideinmainlof}[depth]</td>
<td>\end{mtchideinmainlof}</td>
</tr>
<tr>
<td>\begin{mtchideinmainlot}[depth]</td>
<td>\end{mtchideinmainlot}</td>
</tr>
</tbody>
</table>

*: recommended form.
Table 3.11: Classes and packages needing some precautions with \texttt{minitoc}

<table>
<thead>
<tr>
<th>P/C</th>
<th>Names</th>
<th>Author(s)</th>
<th>Page(s)</th>
<th>Reference(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>P</td>
<td>\texttt{abstract}</td>
<td>Peter R. Wilson</td>
<td>53</td>
<td>[470]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{alnumsec}</td>
<td>Frank Küster</td>
<td>54</td>
<td>[274]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{alphanum}</td>
<td>Felix Braun</td>
<td>75</td>
<td>[103]</td>
</tr>
<tr>
<td>C</td>
<td>\texttt{amsart}</td>
<td>\texttt{AMS}</td>
<td>66</td>
<td>[8]</td>
</tr>
<tr>
<td>C</td>
<td>\texttt{amsbook}</td>
<td>\texttt{AMS}</td>
<td>66</td>
<td>[8]</td>
</tr>
<tr>
<td>C</td>
<td>\texttt{amsproc}</td>
<td>\texttt{AMS}</td>
<td>66</td>
<td>[8]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{appendix}</td>
<td>Peter R. Wilson</td>
<td>64</td>
<td>[471]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{captcont}</td>
<td>Steven Douglas Cochran</td>
<td>54</td>
<td>[131]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{caption}</td>
<td>Axel Sommerfeldt</td>
<td>54</td>
<td>[421, 422, 424]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{caption2}</td>
<td>Axel Sommerfeldt</td>
<td>54</td>
<td>[423]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{ccaption}</td>
<td>Peter R. Wilson</td>
<td>54</td>
<td>[474]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{float}</td>
<td>Anselm Lingnau</td>
<td>54</td>
<td>[302]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{floatrow}</td>
<td>Olga G. Lapko</td>
<td>54</td>
<td>[285]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{flowfram}</td>
<td>Nicola L. C. Talbot</td>
<td>79</td>
<td>[433, 434]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{fncychap}</td>
<td>Ulf A. Lindgren</td>
<td>75</td>
<td>[301]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{hangcaption}</td>
<td>David M. Jones</td>
<td>79</td>
<td>[250]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{hyperref}</td>
<td>Sebastian Rahtz and Heiko Oberdiek</td>
<td>62</td>
<td>[348, 352–354, 387, 390, 391]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{jura}</td>
<td>Felix Braun</td>
<td>75</td>
<td>[103]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{mcaption}</td>
<td>Stephan Hennig</td>
<td>54</td>
<td>[228]</td>
</tr>
<tr>
<td>C</td>
<td>\texttt{memoir}</td>
<td>Peter R. Wilson</td>
<td>65</td>
<td>[479, 481, 482]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{notoccite}</td>
<td>Donald Arseneau</td>
<td>52</td>
<td>[14]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{placeins}</td>
<td>Donald Arseneau</td>
<td>29</td>
<td>[15]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{quotchap}</td>
<td>Karsten Tinnfeld</td>
<td>53</td>
<td>[442]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{romannum}</td>
<td>Peter R. Wilson</td>
<td>54</td>
<td>[480]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{rotfloat}</td>
<td>Sebastian Rahtz and Leonor Barroca</td>
<td>54</td>
<td>[420]</td>
</tr>
<tr>
<td>C</td>
<td>\texttt{scrartcl}, \texttt{scrbook} and \texttt{scrrprt}</td>
<td>Frank Neukam, Markus Kohm, Axel Kirbhorn, and Jens-Uwe Morawski</td>
<td>75</td>
<td>[343, 344, 399]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{sectsty}</td>
<td>Rowland McDonnell</td>
<td>70</td>
<td>[319]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{sfheaders}</td>
<td>Maurizio Loreti</td>
<td>76</td>
<td>[304]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{subfig}</td>
<td>Steven Douglas Cochran</td>
<td>33</td>
<td>[132]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{subfigure}</td>
<td>Steven Douglas Cochran</td>
<td>33</td>
<td>[130]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{titlesec}</td>
<td>Javier Bezos</td>
<td>53</td>
<td>[46]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{titeloc}</td>
<td>Javier Bezos</td>
<td>53</td>
<td>[46]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{tocbibind}</td>
<td>Peter R. Wilson</td>
<td>50</td>
<td>[472]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{tocloft}</td>
<td>Peter R. Wilson</td>
<td>64, 78</td>
<td>[469]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{trivfloat}</td>
<td>Joseph A. Wright</td>
<td>54</td>
<td>[484]</td>
</tr>
<tr>
<td>P</td>
<td>\texttt{varsext}</td>
<td>Daniel Taupin²</td>
<td>53</td>
<td>[437]</td>
</tr>
</tbody>
</table>

*: \textit{Incompatible} with \texttt{minitoc}.  \hspace{1cm} C: Class.  \hspace{1cm} P: Package.

Any class not defining the main standard sectionning commands is \textit{incompatible} with \texttt{minitoc}.
Table 3.12: Checking if inside a minitable

<table>
<thead>
<tr>
<th>Level</th>
<th>Flag for tocs, for lofs, for lots.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part</td>
<td>\ifinparttoc \ifinpartlof \ifinpartlot</td>
</tr>
<tr>
<td>Chapter</td>
<td>\ifinminitoc \ifinminilof \ifinminilot</td>
</tr>
<tr>
<td>Section</td>
<td>\ifinsecttoc \ifinsectlof \ifinsectlot</td>
</tr>
</tbody>
</table>

Table 3.13: Commands for polymorphic entries

<table>
<thead>
<tr>
<th>From OA of:</th>
<th>Command</th>
<th>Arg. 1</th>
<th>Arg. 2</th>
<th>Arg. 3</th>
<th>Arg. 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>sect. command</td>
<td>\mtpolymtoc</td>
<td>\rightarrowparttoc</td>
<td>\rightarrowminitoc</td>
<td>\rightarrowsecttoc</td>
<td>\rightarrowmaintoc</td>
</tr>
<tr>
<td>figure caption</td>
<td>\mtpolymlof</td>
<td>\rightarrowpartlof</td>
<td>\rightarrowminilof</td>
<td>\rightarrowsectlof</td>
<td>\rightarrowmainlof</td>
</tr>
<tr>
<td>table caption</td>
<td>\mtpolymlot</td>
<td>\rightarrowpartlot</td>
<td>\rightarrowminilot</td>
<td>\rightarrowsectlot</td>
<td>\rightarrowmainlot</td>
</tr>
</tbody>
</table>

Table 3.14: Obsolete commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>\firstpartis(N)</td>
<td>(N) is the number of the first part.</td>
</tr>
<tr>
<td>\firstchapteris(N)</td>
<td>(N) is the number of the first chapter.</td>
</tr>
<tr>
<td>\firstsectionis(N)</td>
<td>(N) is the number of the first section.</td>
</tr>
</tbody>
</table>

These commands have no effect (except a harmless warning).
Chapter 4

Examples of documents

Contents

4.1 The mtc-2c.tex document file 91 4.19 The mtc-hia.tex document file 125
4.2 The mtc-2nd.tex document file 92 4.20 The mtc-hir.tex document file 126
4.3 The mtc-3co.tex document file 93 4.21 The mtc-hop.tex document file 127
4.4 The mtc-add.tex document file 96 4.22 The mtc-liv.tex document file 128
4.5 The mtc-ads.tex document file 100 4.23 The mtc-mem.tex document file 132
4.6 The mtc-amn.tex document file 105 4.24 The mtc-mml.tex document file 133
4.7 The mtc-apx.tex document file 105 4.25 The mtc-mu.tex document file 134
4.8 The mtc-art.tex document file 105 4.26 The mtc-nom.tex document file 136
4.9 The mtc-bk.tex document file 110 4.27 The mtc-ocf.tex document file 137
4.10 The mtc-bo.tex document file 115 4.28 The mtc-ofs.tex document file 138
4.11 The mtc-ch0.tex document file 119 4.29 The mtc-sbf.tex document file 140
4.12 The mtc-cri.tex document file 121 4.30 The mtc-scr.tex document file 141
4.13 The mtc-fko.tex document file 121 4.31 The mtc-syn.tex document file 143
4.14 The mtc-fol1.tex document file 122 4.32 The mtc-tbl.tex document file 144
4.15 The mtc-fol2.tex document file 123 4.33 The mtc-tlc.tex document file 145
4.16 The mtc-gap.tex document file 125 4.34 The mtc-tlo.tex document file 146
4.17 The mtc-hi1.tex document file 125 4.35 The mtc-tsf.tex document file 146
4.18 The mtc-hi2.tex document file 125 4.36 The mtc-vti.tex document file 148

This chapter shows the code of some examples of documents. Some are extracted or
derived from real documents, others are just demonstrations to illustrate problems or features.
The code of some specific example files has been shown earlier: mtc-apx.tex on page 67, mtc-hi1.tex on page 68, mtc-hi2.tex on page 69, mtc-gap.tex on page 76, and mtc-tlo.tex on page 79.

Note that the \texttt{lipsum} package \cite{212} is often used to provide filling text.
4.1 The \texttt{mtc-2c.tex} document file

This document shows the use of the minitoc package in a document with a two columns layout. The layout uses the multicol standard package [325] and its multicols environment. We set \texttt{\mtcindent} to zero. We test several combinations. If a minitoc is long enough to be splitted on both columns, the result may be funny.

\begin{verbatim}
\documentclass[12pt,a4paper]{report}
\ProvidesFile{mtc-2c.tex}[2007/01/04]
\usepackage{multicol}
\usepackage{lipsum}
\usepackage[tight,latin,listfiles]{minitoc}
\usepackage[latin]{babel}
\setlength{\mtcindent}{0pt}
\begin{document}
\dominitoc \tableofcontents
\chapter{Primum capitulum}
\begin{multicols}{2}\minitoc\section{Prima sectio}\lipsum[1-2]\section{Secunda sectio}\lipsum[3-4]\end{multicols}
\chapter{Secundum capitulum}
\begin{multicols}{2}\minitoc\section{Prima sectio}\lipsum[5-6]\section{Secunda sectio}\lipsum[7-8]\section{Tertia sectio}\lipsum[9-10]\end{multicols}
\chapter{Tertium capitulum}
\begin{multicols}{2}\minitoc\section{Prima sectio}\lipsum[11-12]\section{Secunda sectio}\lipsum[13-14]\section{Tertia sectio}\lipsum[15-16]\end{multicols}
\chapter{Quadrum capitulum}
\begin{multicols}{2}\minitoc\section{Prima sectio}\lipsum[16-17]\section{Secunda sectio}\lipsum[18-19]\section{Tertia sectio}\lipsum[20-21]\end{multicols}
\end{document}
\end{verbatim}
4.2 The mtc-2nd.tex document file

This document tests the french2.mld minitoc language definition file (section 13.62 on page 498) and its supporting code in the minitoc package. First, the preamble of the document uses the french2 minitoc package language option:

\begin{document}
\ifmtcsecondpart
We test if there is only two parts: if yes, we will use “seconde”. Else (three or more parts), we will use “deuxième”.
\else
Il y a une partie ou plus de deux parties: deuxième.
\fi
\clearpage
The body of the document, with two parts:
\doparttoc
\faketableofcontents
\parttoc
\part{P1}
\chapter{P1C1}
\chapter{P1C2}
\part{P2}
\chapter{P2C1}
\chapter{P2C2}
\end{document}

\footnote{The franc, frnew and mypatches packages are local additions, provided with the minitoc package documentation sources.}
Observe the titles of the parttocs when the document has two then three parts. Note the changes after each compilation. You can play by adding and removing parts. It would be interesting to add a starred part, with starred chapters, at the beginning of the document, this part and these chapters having their entries in the TOC. For instance, try to add the following lines after \faketableofcontents:

\part*{Partie étoilée}
\mtcaddpart[Partie étoilée]
\adjustptc[-2] % IMPORTANT <<<<<<<<<<<<<<<<<<<<<<
\parttoc
\chapter*{Premier chapitre étoilé}
\mtcaddchapter[Premier chapitre étoilé]
\chapter*{Second chapitre étoilé}
\mtcaddchapter[Second chapitre étoilé]

and you will see how the parttoc of the new starred part is titled.

### 4.3 The mtc-3co.tex document file

This document shows how to prepare a minitoc on three columns, with some modifications of the code the mtc@verse environment.

First, we will use a wide paper format (A3) to set the text on two columns and the main TOC on three columns, with the multitoc package [414] (by Martin Schröder).

\documentclass[oneside]{book}
\ProvidesFile{mtc-3co.tex} [% 2007/02/19]
\usepackage[a3paper]{geometry}
\usepackage{lipsum}
\usepackage{multicol}
\usepackage[doc]{multitoc}
\renewcommand{\multicolumntoc}{3}
Then we load the minitoc package, set some parameters and define the number of columns for the minitocs. We alter the \texttt{mtc@verse} environment to add a \texttt{multicols} environment:\footnote{Some vertical adjustments are necessary.}

\begin{verbatim}
\usepackage[tight]{minitoc}
\setlength{\mtcindent}{0pt}
\mtcsetformat{minitoc}{tocrightmargin}{2.55em plus 1fil}
\newcommand{\multicolumnmtc}{3}
\makeatletter
\let\SV@mtc@verse\mtc@verse
\let\SV@endmtc@verse\endmtc@verse
\def\mtc@verse#1{\SV@mtc@verse#1\removelastskip%
\begin{multicols}{\multicolumnmtc}
\raggedcolumns\leavevmode\unskip
\vskip -1.5ex \vskip -1\baselineskip}
\def\endmtc@verse{\end{multicols}\SV@endmtc@verse}
\makeatother
\end{verbatim}

We begin the document, preparing the minitocs and the main TOC. The (first) chapter begins with its minitoc, then the text on two columns. We use the \texttt{lipsum} package\cite{lipsum} to provide filling text; the section number is used to select a \texttt{lipsum} paragraph.

\begin{verbatim}
\begin{document}
\dominitoc
\tableofcontents
\chapter{First chapter}
\minitoc
\begin{multicols}{2}
\section{First section} \lipsum[\arabic{section}]
\section{Second section} \lipsum[\arabic{section}]
\section{Third section} \lipsum[\arabic{section}]
\section{Fourth section} \lipsum[\arabic{section}]
\section{Fifth section} \lipsum[\arabic{section}]
\section{Sixth section} \lipsum[\arabic{section}]
\section{Seventh section} \lipsum[\arabic{section}]
\section{Eighth section} \lipsum[\arabic{section}]
\section{Ninth section} \lipsum[\arabic{section}]
\section{Tenth section} \lipsum[\arabic{section}]
\section{Eleventh section} \lipsum[\arabic{section}]
\section{Twelfth section} \lipsum[\arabic{section}]
\section{Thirteenth section} \lipsum[\arabic{section}]
\section{Fourteenth section} \lipsum[\arabic{section}]
\section{Fifteenth section} \lipsum[\arabic{section}]
\section{Sixteenth section} \lipsum[\arabic{section}]
\section{Seventeenth section} \lipsum[\arabic{section}]
\section{Eighteenth section} \lipsum[\arabic{section}]
\end{verbatim}

\footnote{Some vertical adjustments are necessary.}
4.4 The mtc-add.tex document file

This document shows how to add special entries in the table of contents, and the interaction with the tocbibind package [472].

Load the minitoc package, or mtcoff.

\usepackage[tight,hints,listfiles]{minitoc}
\usepackage{mtcoff}

\makeindex
\begin{document}
\dominitoc \dominilof \dominilot
\tableofcontents \mtcaddchapter
\listoffigures \mtcaddchapter
\listoftables \mtcaddchapter

\chapter{First chapter}\index{chapter!normal}
\minitoc \mtcskip
\minilof \mtcskip
\minilot

We call the mini-table preparation commands:
\dominitoc \dominilof \dominilot

We are using the tocbibind package to add special entries in the table of contents, so we must take the precautions specified in section 1.5.5 on page 50:
\tableofcontents \mtcaddchapter
\listoffigures \mtcaddchapter
\listoftables \mtcaddchapter

For a chapter, we want a minitoc, a minilof and a minilot:
\chapter{First chapter}\index{chapter!normal}
\minitoc \mtcskip
\minilof \mtcskip
\minilot
\mtcskip
Then the text of the chapter, with sections, figures and tables:

\section{First section}
\begin{figure}[tp] \caption{First figure} \end{figure}
\begin{table}[tp] \caption{First table} \end{table}

\section{Second section}
A small nice citation from \cite{dark}:
\begin{quote}
A bird cannot land once on a great tree and claim to know it. But I imagine that he would, yes.
\end{quote}
Iain~M.~\textsc{Banks}~(1993), \textsl{Against a dark background.}
\begin{figure}[tp] \caption{Second figure} \end{figure}
\begin{table} \caption{Second table} \end{table}

\chapter*{Second chapter, starred}
\mtcaddchapter
A starred chapter requires a special treatment; three solutions are possible. You can test variations on the \mtcaddchapter command. Just uncomment one (and only one) of the \mtcaddchapter commands after \chapter* in the source code of \texttt{mtc-add.tex}. For each case, look at the Table of Contents and the involved chapter.

\begin{itemize}
\item \begin{quote}
This is a starred chapter; you can test here variations on the \verb|\mtcaddchapter| command. Just uncomment one (and only one) of the \verb|\mtcaddchapter| commands after \verb|\chapter*| in the source code of \texttt{mtc-add.tex}. For each case, look at the \index{Table of Contents}Table of Contents and at this chapter.
\end{quote}
\end{itemize}
I also added a lot of index entries, just to test.

\chapter{Third chapter}
\index{chapter!normal}
\minitoc \mtcskip
\minilof \mtcskip
\minilot
\section{Third section}
\begin{figure} \caption{Third figure} \end{figure}
\begin{table} \caption{Third table} \end{table}

\section{Fourth section}
\begin{figure} \caption{Fourth figure} \end{figure}
\begin{table} \caption{Fourth table} \end{table}

As we want to add an entry for the bibliography in the table of contents, and we use the \texttt{tocbibind} package for that, we must add a correction with \texttt{\adjustmtc}:

\bibliographystyle As we want to add an entry for the index in the table of contents, and we use the \texttt{tocbibind} package for that, we must add a correction; two solutions are available: use \texttt{\mtcfixindex} or the other given three lines:

\begin{verbatim}
\begin{figure} \caption{Fifth figure} \end{figure}
\end{verbatim}
The next chapter asks for a minitoc, a minilof and a minilot, but contains no tables; hence the minitoc package will give some warnings.

And we need also its small bibliographic data base:

- the english documentation of the minitoc package [157]:

```latex
@MISC{minitoc,
 TITLE="The \textsf{minitoc} package",
 AUTHOR="Drucbert, Jean-Pierre F.",
 NOTE="\url{http://mirror.ctan.org/macros/latex/contrib/minitoc/minitoc.pdf}",
 MONTH=jul,
 YEAR=2008
}
```

- the french documentation of the minitoc package [156]:

```latex
@MISC{minitoc-fr,
 TITLE="Le paquetage \textsf{minitoc}",
 AUTHOR="Drucbert, Jean-Pierre F.",
 NOTE="\url{http://mirror.ctan.org/macros/latex/contrib/minitoc/minitoc-fr.pdf}",
 MONTH=jul,
 YEAR=2008
}
```

- the documentation of the shorttoc package [155]:

```latex
• a novel [24] from which a short citation is taken:

@BOOK{dark,
  TITLE="{Against a Dark Background}" ,
  AUTHOR="Banks, Iain Menzies",
  PUBLISHER="Bantam Books",
  ISBN="0553292240 (pb)",
  YEAR=1993}

But the database created this way must be trimmed of some spurious lines; on Unix-like systems, do:

```
cat mtc-add.bib | grep -v '^%' > addbib; mv addbib mtc-add.bib
```

### 4.5 The `mtc-ads.tex` document file

This document uses the `article` class and shows some problems for adding special entries in the table of contents and some problems with floating objects. We need to use the `tocbibind` package [472] for the first ones and the `minitoc` `insection` package option to avoid the drift if floating objects outside of their section. That gives the following document preamble:

```
\documentclass[oneside,12pt,a4paper]{article}
\ProvidesFile{mtc-ads.tex} \[2007/01/04\]
\usepackage[url,tocbibind,makeidx]
\makeatletter \newif\ifscan@allowed \scan@allowedtrue \makeatother
\def\dotfil{\leaders\hbox to.6em{\hss .\hss}\hfil}
\def\pfill{\unskip~\dotfill\penalty500\strut\nobreak\dotfil~\ignorespaces}
\usepackage[tight,hints,insection]{minitoc}
%% \usepackage{mtcoff}
\makeindex
```

Note that we should use no preamble for this file in `minitoc.ins`; nevertheless, some spurious lines are still generated. This problem is not yet corrected now, so we keep the solution.
The preparation commands:
\begin{document}
560 \begin{document}
561 \doartoftoc
562 \doartoflot
563 \doartoflot
564 \doartsctoc
565 \doartsctoflot
566 \doartsctlot
567 \tableofcontents
568 \mtcaddsection
569 \listoffigures
570 \listoftables
571 \part{Part~1}
572 \parttoc \mtcskip \partlof \mtcskip \partlot
573 \section{First section}
574 \index{section!normal}
575 \secttoc \mtcskip \sectlof \mtcskip \sectlot
576 \subsection{First subsection}
577 \begin{figure}[tp] \caption{First figure} \end{figure}
578 \begin{table}[tp] \caption{First table} \end{table}
579 \subsection{Second subsection}
580 \begin{figure}[tp] \caption{Second figure} \end{figure}
581 \begin{table} \caption{Second table} \end{table}
582 \section*{Second section, starred}
583 \index{section!starred}

The commands to prepare the table of contents, the list of figures and the list of tables. As we use the tocbibind package, we must add some \mtcaddsection commands:
563 \setcounter{tocdepth}{6}
564 \setcounter{parttocdepth}{6}
565 \setcounter{secttocdepth}{6}
566 \tableofcontents \mtcaddsection
567 \listoffigures \mtcaddsection
568 \listoftables \mtcaddsection
569 \part{Part~1}
570 \parttoc \mtcskip \partlof \mtcskip \partlot
571 \section{First section}
572 \index{section!normal}
573 \secttoc \mtcskip \sectlof \mtcskip \sectlot
574 \subsection{First subsection}
575 \begin{figure}[tp] \caption{First figure} \end{figure}
576 \begin{table}[tp] \caption{First table} \end{table}
577 \subsection{Second subsection}
578 \begin{figure}[tp] \caption{Second figure} \end{figure}
579 \begin{table} \caption{Second table} \end{table}
580 \section*{Second section, starred}
581 \index{section!starred}

The body of the document: a part with its part-level mini-tables, some sections with their section-level mini-tables. The document has an index and contains figures and tables.
569 \part{Part~1}
570 \parttoc \mtcskip \partlof \mtcskip \partlot
571 \section{First section}
572 \index{section!normal}
573 \secttoc \mtcskip \sectlof \mtcskip \sectlot
574 \subsection{First subsection}
575 \begin{figure}[tp] \caption{First figure} \end{figure}
576 \begin{table}[tp] \caption{First table} \end{table}
577 \subsection{Second subsection}
578 \begin{figure}[tp] \caption{Second figure} \end{figure}
579 \begin{table} \caption{Second table} \end{table}
580 \section*{Second section, starred}
581 \index{section!starred}

Here, we try a starred section, with its entry in the table of contents. You can try several solutions (good or bad).
594 \section*{Second section, starred}
595 \index{section!starred}
This is a starred section; you can test here variations on the \verb|\mtcaddsection| command. Just uncomment one (and only one) of the \verb|\mtcaddsection| commands after \verb|\section*| in the source code of \texttt{mtc-add.tex}. For each case, look at the \index{Table of Contents}Table of Contents and at this section.

I also added a lot of index entries, just to test.
A small nice citation from \cite{dark}:

A bird cannot land once on a great tree and claim to know it. But I imagine that he would, yes.

Iain M. Banks (1993), *Against a dark background.*

This is a starred section; you can test here variations on the \verb|\mtcadsection| command. Just uncomment one (and only one) of the \verb|\mtcadsection| commands after \verb|\section*| in the source code of \texttt{mtc-add.tex}.

For each case, look at the Table of Contents and at this section. I also added a lot of index entries, just to test.
The bibliography: as we want an entry for it in the table of contents, we use the `tocbibind` package [472] and a correction with `adjuststc`:

\begin{figure} \caption{Eighth figure} \end{figure}
\begin{table} \caption{Eighth table} \end{table}
\bibliographystyle
\bibliography
\adjuststc
\nocite*
\def\noopsort#1{\relax}
\bibliographystyle{plain}
\bibliography{mtc-add}
\adjuststc
\printindex
\mtcfixindex
The index: as we want an entry for it in the table of contents, we use the `tocbibind` package [472] and a correction with `mtcfixindex`:

\printindex
\mtcfixindex % use this OR the 3 following lines
\% contains no tables but asks for a sectlot! No sectlot printed.
\section{App.~1}
\index{section!appendix}
\secttoc \mtcskip \sectlof \mtcskip \sectlot
\subsection{Ninth subsection}
\begin{figure} \caption{Ninth figure} \end{figure}
\begin{table} \caption{Ninth table} \end{table}
\subsection{Tenth subsection}
\begin{figure} \caption{Tenth figure} \end{figure}
\begin{table} \caption{Tenth table} \end{table}
\subsection{Eleventh subsection}
\begin{figure} \caption{Eleventh figure} \end{figure}
\begin{figure} \caption{Twelfth figure} \end{figure}
\subsection{Twelfth subsection}
\begin{figure} \caption{Thirteenth figure} \end{figure}
\begin{figure} \caption{Fourteenth figure} \end{figure}
4.6 The \texttt{mtc-amm.tex} document file

This example shows the use of the \texttt{appendices} environment in a \texttt{memoir} class document when the \texttt{minitoc} package is loaded. First, the preamble:

\begin{verbatim}
\documentclass[oneside]{memoir}
\ProvidesFile{mtc-amm.tex}[2007/08/29]
\usepackage{lipsum} % filling text
\usepackage{hyperref}
\usepackage[tight]{minitoc}
\begin{document}
\dominitoc \tableofcontents
\adjustmtc
\chapter{First chapter}
\minitoc
\lipsum[1]
\section{First section}
\lipsum[2]
\begin{appendices}
\addappheadtotoc
\adjustmtc %correction!
\chapter{Afterthoughts}
\minitoc
\lipsum[3]
\section{Further remarks}
\lipsum[4]
\chapter{Last wills}
\minitoc
\section{Testament}
\lipsum[5]
\end{appendices}
\chapter{Conclusion}
\minitoc
\section{Bye}
\lipsum[6]
\chapter{Back from Hell}
\section{Not dead yet!}
\lipsum[7]
\section{I will survive}
\lipsum[8]
\end{document}
\end{verbatim}

The appendices are set in an \texttt{appendices} environment; we can add an entry in the TOC with \texttt{\addappheadtotoc} (a command from the \texttt{memoir} class):

\begin{verbatim}
\begin{appendices}
\addappheadtotoc
\adjustmtc %correction!
\chapter{Afterthoughts}
\minitoc
\lipsum[3]
\section{Further remarks}
\lipsum[4]
\chapter{Last wills}
\minitoc
\section{Testament}
\lipsum[5]
\end{appendices}
\chapter{Conclusion}
\minitoc
\section{Bye}
\lipsum[6]
\chapter{Back from Hell}
\section{Not dead yet!}
\lipsum[7]
\section{I will survive}
\lipsum[8]
\end{document}
\end{verbatim}

4.7 The \texttt{mtc-apx.tex} document file

The \texttt{mtc-apx.tex} document file is described in section 2.25 on page 67.

4.8 The \texttt{mtc-art.tex} document file

This is a basic document using the \texttt{minitoc} package. It contains sections but no chapters, so it must use an \texttt{article}-like document class. You should work on a \texttt{copy} of this file and can alter its preamble and its contents to make experiments with parameters. A typical preamble follows:

\begin{verbatim}
\begin{verbatim}
\documentclass[oneside]{article}
\hyphenation{ad-dress}
\begin{document}
\section{First section}
\lipsum[1]
\section{Second section}
\lipsum[2]
\section{Third section}
\lipsum[3]
\end{document}
\end{verbatim}
\end{verbatim}
This file contains a set of tests for minitoc package. You can alter most of parameters to test. article \section must be defined)
\documentclass[12pt,a4paper]{article}
\ProvidesFile{mtc-art.tex}
[2007/06/06]
\usepackage{lipsum} % provides filling text
\usepackage{hyperref} % If used, load it BEFORE minitoc
\usepackage[tight,insection]{minitoc}
\setcounter{secnumdepth}{5} % depth of numbering of sectionning commands
\setcounter{tocdepth}{3} % depth of table of contents
\setlength{\stcindent}{24pt} % indentation of secttocs, default
\renewcommand{\stcfont}{\small\rmfamily\upshape\mdseries} % font for secttocs, default
\renewcommand{\stcSSfont}{\small\sf} % you can make experiments with
\renewcommand{\stcSSSfont}{\small\sf} % but it is "fontomania"...
\renewcommand{\stcPfont}{\small\rmfamily\upshape\mdseries} % font for secttocs, subsections
\renewcommand{\stcSPfont}{\small\rmfamily\upshape\mdseries} % or \flushbottom, at your choice

If you want to use sections numbered in each part (the section number restarts to 1 at the beginning of each part), uncomment the 3 lines of code below. This demonstrates that the numbering of the secttoc files is independent on the numbering of the sections (it is absolute).

\% TEST: uncomment the next line to test
\% resetting section number in each part
\% \makeatletter \@addtoreset{section}{part} \makeatother \% END TEST

We begin the body of the document. You can still alter some parameters (presence or absence of rules and page numbers in the mini-tables):

\begin{document}

The preparation commands, with their optional argument if necessary:
\dosecttoc
\dosectlof
\dosectlot
\doparttoc
\dopartlof
\dopartlot

It is necessary to create the contents files; use the “fake” version to not print.
\faketableofcontents
\fakelistoffigures
\fakelistoftables

\end{document}
There is the text of the document, with its sectioning commands; we define a part, with a parttoc, a partlof (with the title on the right) and a partlot:

A section, in two columns mode, with a secttoc (title on the right), and a sectlof; this section contains subsections to make a non-empty secttoc but no figures (to detect an empty sectlof).

A lot of subsections:

We return to the one column mode. Then a section with a secttoc and a sectlof (there are subsections and figures). The insection package option should ensure that floating objects (like figures) do not drift outside their section.
Examples of documents

\begin{picture}(100,50) \end{picture}  
\caption{F1}  % (I have not tested tables, but it is similar)
\end{figure}

\subsection{TT1} % tests optional arg. of a sectionning command
\lipsum[34]
\paragraph{TTT1} \lipsum[35]
\subparagraph{TTTT1} \lipsum[36]
\begin{figure}[t]
\setlength{\unitlength}{1mm}
\begin{picture}(100,50) \end{picture}
\caption{F2}  % tests optional arg. of a caption
\end{figure}

\subsection{T2} \lipsum[37]

\section*{CCCCC} % tests a pseudo-section. should have no secttoc
\addstarredsection{CCCCC}
\mtcaddsection{CCCCC}  % ADDED
\secttoc \mtcskip \sectlof  % ADDED
\lipsum[38]
\subsection{U1} \lipsum[39]
\subsubsection{UU1} \lipsum[40]
\paragraph{UUU1} \lipsum[41]
\subparagraph{UUUU1} \lipsum[42]
\subsection{U2} \lipsum[43]
\part{Second Part}
\parttoc
\partlof[c]
\partlot
\subsection{V1} \lipsum[45]
\subsubsection{VV1} \lipsum[46]
\paragraph{VVV1} \lipsum[47]
\subparagraph{VVVV1} \lipsum[48]
\begin{figure}[t] % tests compatibility with floating bodies
\setlength{\unitlength}{1mm}
\begin{picture}(100,50) \end{picture}
\caption{F3}  % (I have not tested tables, but it is similar)
\end{figure}
\lipsum[49]  
\subsection{V2} \lipsum[50]
We change the depth of the secttocs, inside a local group (a pair of braces):

\begin{verbatim}
\section
\chapter
\secttoc
919 \section{EEEEEE} % this section should have a secttoc
920 \% % left brace, see below
921 \secttocdepth{3} % depth of sect table of contents;
922 \% try with different values.
923 \secttoc
924 \mtcskip \sectlof %ADDED
925 \% right brace
926 \% this pair of braces is used to keep local the change on secttocdepth.
927 \lipsum[51]
928 \subsection{W1} % with the given depth
929 \lipsum[52]
930 \subsubsection{WW1} \lipsum[53]
931 \paragraph{WWW1} \lipsum[54]
932 \begin{figure}[t] % tests compatibility with floating bodies
933 \setlength{\unitlength}{1mm}
934 \begin{picture}(100,50) \end{picture}
935 \caption{F4} % (I have not tested tables, but it is similar)
936 \end{figure}
937 \FloatBarrier
938 \subparagraph{WWWW1} \lipsum[55]
939 \subsection{W2} \lipsum[56]
940 \% no chapter in article class \chapter*
941 \part{Appendices}
942 \parttoc \mtcskip
943 \partlof \mtcskip
944 \partlot
945 \FloatBarrier
946 \appendix
947 \section{Comments} \lipsum[57]
948 \secttoc
949 \mtcskip \sectlof %ADDED
950 \subsection{C1} \lipsum[58]
951 \subsection{C2} \lipsum[59]
952 \subsection{C3} \lipsum[60]
953 \begin{figure}[hb] % tests compatibility with floating bodies
954 \setlength{\unitlength}{1mm}
955 \begin{picture}(100,50) \end{picture}
956 \caption{F5} % (I have not tested tables, but it is similar)
957 \end{figure}
958 \FloatBarrier
959 \subsection{C4} \lipsum[61]
960 \FloatBarrier
961 \section[Evolution]
962 \secttoc
963 \sectlof % empty
964 \sectlot % empty
965 \lipsum[62]
966 \subsection{D1} \lipsum[63] \subsection{D2} \lipsum[64]
967 \subsection{D3} \lipsum[65] \subsection{D4} \lipsum[66]
968 \end{document}
\end{verbatim}
4.9 The mtc-bk.tex document file

This is a basic document using the minitoc package. It contains chapters, so it must use a book-like or report-like document class. You should work on a copy of this file and can alter its preamble and its contents to make experiments with parameters. A typical preamble follows:

\documentclass[12pt,a4paper]{report} % the report class uses less pages
\usepackage{lipsum} % provides filling text
\usepackage{mtcoff} % usepackage{tight} (minitoc) % tight option make shorter mini-tables
\setcounter{secnumdepth}{5} % depth of numbering of sectioning commands
\setcounter{tocdepth}{3} % depth of table of contents
\setlength\mtcindent{24pt} % indentation of minitocs, default
\renewcommand\mtcfont{\small\rm} % font for minitocs, default
\renewcommand\mtcSfont{\small\bf} % font for minitocs, sections, default
\renewcommand\mtcSSfont{\small\sf} % font for minitocs, subsections
\renewcommand\mtcSSSfont{\small\sffamily} % font for minitocs, sub-subsections
\renewcommand\mtcPfont{\small\bf\sffamily} % font for minitocs, paragraphs
\renewcommand\mtcSPfont{\small\sffamily\bf} % font for minitocs, section paragraphs
\renewcommand\mtcSSSPfont{\small\sf\sffamily\bf} % font for minitocs, sub-subsection paragraphs
\renewcommand\mtcSSSSfont{\small\sffamily\sffamily\bf} % font for minitocs, sub-sub-subsection paragraphs
\renewcommand\mtcSSSSSfont{\small\sffamily\sffamily\sffamily\bf} % font for minitocs, sub-sub-sub-subsection paragraphs
\raggedbottom % or \flushbottom, at your choice

If you want to use chapters numbered in each part (the chapter number restarts to 1 at the beginning of each part), uncomment the 3 lines of code below. This demonstrates that the numbering of the minitoc files is independent on the numbering of the chapters (it is absolute).

\begin{document}

We begin the body of the document. You can still alter some parameters (presence or absence of rules and page numbers in the mini-tables):

\begin{verbatims}
\begin{verbatim}
\maketitle
\tableofcontents
\chapter{Introduction}
\section{Background}
\subsection{Previous Work}
\subsubsection{Related Work}
\section{Methodology}
\subsection{Data Collection}
\subsection{Data Analysis}
\section{Results}
\subsection{Results of Experiment 1}
\subsection{Results of Experiment 2}
\section{Discussion}
\subsection{Discussion of Results}
\subsection{Future Work}
\section{Conclusion}
\end{verbatim}
\end{verbatims}
The preparation commands, with their optional argument if necessary:

\dominitoc\dominilof\dominilot\doparttoc\dopartlof\dopartlot

1005 \dominitoc % centers title of minilof’s
1006 \dominilof[c] % centers title of minilof’s
1007 \dominilot
1008 \doparttoc % test of parttoc/partlof stuff
1009 \dopartlof % added in version #15
1010 \dopartlot % added in version #15

It is necessary to create the contents files; use the “fake” version to not print.

\tableofcontents \listoffigures \fakelistoftables

1011 \tableofcontents % or \faketableofcontents
1012 \listoffigures % or \fakelistoffigures
1013 \fakelistoftables % or \listoftables

Uncomment the following line if the first chapter must be numbered “0”:

1014 \% \addtocounter{chapter}{-1} % to begin with Chapter 0

There is the text of the document, with its sectioning commands:

\part \parttoc \partlof \partlot
1015 \part{First Part} \parttoc \partlof[r] \partlot[r]

A chapter, in two column mode, with a minitoc (title on the right):

1017 \twocolumn\sloppy % the minitoc in twocolumn layout is ugly,
1018 \chapter{AAAAA} % a chapter with a lot of sections
1019 \minitoc[r] % minitoc title on the right
1020 \lipsum[1]
1021 \section{S1} \lipsum[2]
1022 \section{S2} \lipsum[3]
1023 \section{S3} \lipsum[4]

A starred section; we want an entry in the TOC, so we add it the normal way:

1024 \section*{S4}
1025 \addtocontentsline{toc}{section}{\protect\numberline{}S4}
1026 \lipsum[5]

A lot of subsections:

1027 \section{S5} \lipsum[6] 1028 \section{S6} \lipsum[6]
1029 \section{S7} \lipsum[7] 1030 \section{S8} \lipsum[9]
1031 \section{S9} \lipsum[10] 1032 \section{S10} \lipsum[11]
1033 \section{S11} \lipsum[12] 1034 \section{S12} \lipsum[13]
We return to one column mode. A new chapter, with a minitoc, a minilof and a minilot:

\chapter
\minitoc
\minilof
\minilot
\mtcskip % put some skip here
\minilof % a minilof
\minilot % a minilot
\chapchapter
\section{T1} % tests compatibility with floating bodies
\begin{figure}[t]
\setlength{\unitlength}{1mm}
\begin{picture}(100,50)
\end{picture}
\caption{F1} % (tables are similar)
\end{figure}
\begin{table}[b]
\setlength{\unitlength}{1mm}
\begin{picture}(100,50)
\end{picture}
\caption{T1} % (tables are similar)
\end{table}
\clearpage
\subsection[tt1]{TT1} % tests optional arg. of a sectionning command
\lipsum[36]
\subsubsection{TTT1} \lipsum[37]
\paragraph{TTTT1} \lipsum[38]
\begin{figure}
\setlength{\unitlength}{1mm}
\begin{picture}(100,50)
\end{picture}
\caption[F2]{F2} % tests optional arg. of a caption
\end{figure}
\section{T2} \lipsum[39]

A starred chapter with an entry added in the TOC; all subordinate (lower) sectionning commands must also be starred.
\chapter*
\section*{TTTT1} % tests optional arg. of a sectionning command
\addstarredchapter
\addcontentstoc
\addcontentstolof
This chapter has no minitoc, but if you uncomment \minitoc, the minitoc will appear.

We change the depth of the minitocs, inside a local group (a pair of braces):

\chapter{EEEEEE} % this chapter should have a minitoc
{ % left brace, see below
\setcounter{minitocdepth}{3} % depth of mini table of contents;
\minitoc % try with different values.
}\minitoc % right brace
% this pair of braces is used to keep local the change
\section*{W1} % with the given depth
Here, we encounter a classical problem: to make a local table of contents for a set of appendices, while hiding these entries in the main table of contents. First, we create a part, with its parntoc:

\appendix
\part
\parttoc

Then, we begin a \mtchideinmaintoc environment, with the hiding depth as optional argument:

\mtchideinmaintoc
\chapter
\minitoc

We terminate the part by adding a marker in the TOC file, then we must close this \mtchideinmaintoc environment:

\addtocontents
4.10 The mtc-bo.tex document file

This document shows the use of the minitoc package in a document using a two column layout for some portions and the tocloft package [469]. The aim is to begin a chapter with a special head and a preliminary block containing a minitoc and some indications, on two columns. The preamble loads the geometry package [447], which defines the global page layout, the multicol package [325], the color package [120], because we want a colored background for the minitoc, the tocloft package [469], to change some parameters of the minitoc, and, at least, the minitoc package itself:

\documentclass[10pt]{book}
\ProvidesFile{mtc-bo.tex}\
\[2007/04/17\]
\usepackage[paperwidth=8.5in,paperheight=11in,\
lmargin=1.25in,rmargin=1.25in,tmargin=1in,bmargin=1in]{geometry}
\usepackage{multicol}
\usepackage{color}
\setlength{\cftsecindent}{0cm}
\setlength{\cftsecnumwidth}{15 pt}
\setlength{\cftsubsecindent}{\cftsecindent}
\addtolength{\cftsubsecindent}{\cftsecnumwidth}
\setlength{\cftsubsecnumwidth}{20 pt}
\setlength{\cftsubsubsecindent}{\cftsubsecindent}
\addtolength{\cftsubsubsecindent}{\cftsubsecnumwidth}
\cftpagenumbersoff

Note that if we want to suppress the page numbers in the minitoc, we must use the commands from tocloft:

\cftpagenumbersoff{sec}
\cftpagenumbersoff{subsec}
We load the minitoc package and change the indentation, suppress the rules and change the minitoc title. The hyperref package can also be loaded (after minitoc).

\usepackage[francais,tight]{minitoc}
\usepackage{hyperref}
\setlength{\mtcindent}{0pt}
\nomtcrule % pas de filets en haut et en bas de la mini-tdm
\nomtcpagenumbers % pas de numéro de pages
\renewcommand{\mtctitle}{Contenu de la rencontre}

We load also some packages for the french language (some are local):

\usepackage[francais]{babel}
\usepackage{franc,frnew}
\usepackage[T1]{fontenc}
\usepackage[latin1]{inputenc}
\usepackage{mypatches}

This code redefines the format of the chapter head:

\makeatletter
\def\@makechapterhead#1{\vspace*{10\p@} %
\parindent \z@ \raggedleft \normalfont
\interlinepenalty\@M
\ifnum \c@secnumdepth >\m@ne %
\Huge\bfseries\sffamily Rencontre \thechapter\quad
\fi
\Huge \bfseries \sffamily #1\par
\vskip 10\p@ %
}%
\def\@makeschapterhead#1{\vspace*{10\p@} %
\parindent \z@ \raggedright \normalfont
\interlinepenalty\@M
\ifnum \c@secnumdepth >\m@ne %
\Huge\bfseries\sffamily Rencontre \thechapter\% \quad
\fi
\Huge \bfseries \sffamily #1\par
\vskip 10\p@ %
}%
\makeatother

We define an environment (pageUn) for the block placed at the beginning of a chapter. This block contains a minitoc, then a sequence of informations given by the 6 parameters of the environment. The block uses a multicols environment to typeset on two columns. Some decorations are added: rules, colored background for the minitoc.

\makeatletter
\def\@makechapterhead#1{%
\vspace*{10\p@} %
{\parindent \z@ \raggedleft \normalfont
\interlinepenalty\@M
\ifnum \c@secnumdepth >\m@ne %
\Huge\bfseries\sffamily Rencontre \thechapter\% \quad
\fi
\Huge\bfseries\sffamily #1\par\nobreak
\vskip 10\p@ %
}}
\def\@makeschapterhead#1{%
\vspace*{10\p@} %
{\parindent \z@ \raggedright \normalfont
\interlinepenalty\@M
\ifnum \c@secnumdepth >\m@ne %
\Huge\bfseries\sffamily Rencontre \thechapter\% \quad
\fi
\Huge \bfseries \sffamily #1\par\nobreak
\vskip 10\p@ %
}}\makeatother
% param1: date de la rencontre
% param2: nombre de périodes
% param3: liste des documents distribués
% param4: messages
% param5: lecture
% param6: exercices

\newenvironment{pageUn}{\parindent = 0.0in\rule{\linewidth}{1pt}\begin{multicols}{2}{\large \bfseries Math. pour médecine nucléaire\textit{(#2)}}\vfill\columnbreak\raggedleft \bfseries Automne 2003\ #1\end{multicols}\vspace{-18pt}\rule{\linewidth}{1pt}\setlength{\columnseprule}{.3pt} \setlength{\columnsep}{1cm}\begin{multicols}{2}\%: TABLE DES MATIÈRES (col. gauche)\colorbox[cmyk]{.1,0,0,0}{\parbox{\linewidth}{\setcounter{minitocdepth}{3}\minitoc}}\vfill \columnbreak ~ \vfill \mtcskip\%: DOCUMENTS DISTRIBUÉS (début col. droite)\begin{itemize} \renewcommand{\labelitemi}{$\star$} #3 \end{itemize}\%: MESSAGES AUX ÉTUDIANTS\vspace{12pt}{\large \bfseries Messages}\begin{itemize} \renewcommand{\labelitemi}{$\star$}#4\end{itemize}\%: LECTURE\vspace{12pt}{\large \bfseries Lecture}\vspace{-6pt} \par#5 \par\% EXERCICES\vspace{12pt}{\large \bfseries Exercices}\vspace{-6pt}\par#6\par\newpage\end{multicols}\%\newpage}
The body of the document:

%: BEGIN %
\begin{document}
\begin{minitoc}
\faketableofcontents
\chapter{Première étape (début)}
%\minitoc
\begin{pageUn}
\item Date du cours
\item Lundi 25 août
\} 
\item Durée du cours
\item 2 périodes
\} 
\item Liste des documents à distribuer
\item Plan de cours
\item Fiche d'identification
\item Grille horaire
\item Feuilles d'exercices supplémentaires
\} 
\item Ne pas oublier
\item Acheter le livre de référence
\item Apporter une disquette
\} 
\item Lecture
\item Lire les pages ppp à ppp et ppp à ppp
\} 
\item Exercice
\item Faire les exercices mnn de la page ppp
\} 
\end{pageUn}
\end{minitoc}
\end{document}
4.11 The mtc-ch0.tex document file

This document shows the use of the minitoc package in a document using a starred first chapter, inducing the “Chapter Zero” problem.
The first chapter is starred, but contains real numbered sections. We add an entry in the TOC for this chapter and see that its sections are using “0” as chapter number:

\chapter*{Chapter One (starred)}
\mtcaddchapter{Fake chapter one}
\minitoc
\section{Chap 1, section 1}
That’s right, folks -- we’re close to the release of Firefox and Thunderbird-1.0 and, just like our last 1.0-release, we want to organize worldwide parties to celebrate.

Thanks to Dominik ‘Aeneas’ Schnitzer, we have an all-new and improved Mozilla Party Webtool-2.0. You can create your own party, or sign up for one already in progress -- and, in an improvement on Webtool-1.0, organizers can now edit and update party details. The tool allows you to organize a celebration in any of 243-countries, principalities, dominions and islands around the world. Never let it be said that we do things by halves around here.

The second chapter is normal:
\chapter{Chapter Two (numbered one)}
\minitoc
\section{Chapter 2, section 1}
That’s right, folks -- we’re close to the release of Firefox and Thunderbird-1.0 and, just like our last 1.0-release, we want to organize worldwide parties to celebrate.

Thanks to Dominik ‘Aeneas’ Schnitzer, we have an all-new and improved Mozilla Party Webtool-2.0. You can create your own party, or sign up for one already in progress -- and, in an improvement on Webtool-1.0, organizers can now edit and update party details. The tool allows you to organize a celebration in any of 243-countries, principalities, dominions and islands around the world. Never let it be said that we do things by halves around here.
4.12 The mtc-cri.tex document file

This document shows the use of the minitoc package in a document with a starred part and starred chapters. Note the use of the adjustment commands. This example is not commented: just follow the insertion of the mini-tables in the mtc-cri.log file.

\mtcsetdepth{parttoc}{2}
\begin{document}
\doparttoc \dominitoc
\tableofcontents
\part*{Présentation générale}
\mtcaddpart[Présentation générale]
\adjustptc[-2]
\parttoc
Texte de la présentation générale\ldots
\chapter*{Les auteurs}
\mtcaddchapter[Les auteurs]
Présentation des auteurs\ldots
\chapter*{Les lecteurs}
\mtcaddchapter[Les lecteurs]
Présentation des lecteurs\ldots
\part{Première partie}
\parttoc
\chapter{Introduction}
\mtcaddchapter[Introduction]
\chapter{Premier chapitre}
\section{Première section~A}
\section{Deuxième section~A}
\chapter{Deuxième chapitre}
\section{Première section~B}
\section{Deuxième section~B}
\part{Deuxième partie}
\parttoc
\chapter{Premier chapitre}
\section{Première section~C}
\section{Deuxième section~C}
\chapter{Deuxième chapitre}
\section{Première section~D}
\section{Deuxième section~D}
\end{document}

4.13 The mtc-fko.tex document file

This is a document using the scrbook class. Without any precaution, some entries in the minitocs are not in the right font (bold sans serif) like in the main table of contents; moreover, the language of the minitoc titles is not correct because the options of the babel package are not transferred to the minitoc package. To solve the language problem, we just set “german” as a global option in the \documentclass command (babel and minitoc will hence use this global option).
Then we load the packages and set some parameters:
\usepackage[germanb]{babel}
\usepackage[tight]{minitoc}
\setlength{\mtcindent}{0pt} % optional
\mtcsetfont
\mtcsettitlefont
\sectfont
\mtcsettitle
\mtcsetfont{minitoc}{section}{\sectfont\small}
\mtcsettitlefont{minitoc}{\sectfont\large}
\mtcsettitle{minitoc}{Inhalt}
\dominitoc
\tableofcontents
\chapter
\minitoc % Aufruf Minitoc
\section{Dieser Text ist in minitoc serifenlos}
Auch der Text ‘Inhaltsangabe’ will so wie koma es definiert.

\begin{document}
\dominitoc \tableofcontents
\chapter{Ein serifenloses Kapitel}
\minitoc % Aufruf Minitoc
\section{Dieser Text ist in minitoc serifenlos}
Auch der Text ‘Inhaltsangabe’ will so wie koma es definiert.
\end{document}

\doparttoc
\dominitoc
\tableofcontents
\part
\doparttoc
\dominitoc
\tableofcontents
\part
\doparttoc
\dominitoc
\tableofcontents
\part
\doparttoc
\dominitoc
\tableofcontents
\part
\doparttoc
\dominitoc
\tableofcontents
\part
\doparttoc
\dominitoc
\tableofcontents
\part
\doparttoc
\dominitoc
\tableofcontents
\part
\doparttoc
\dominitoc
\tableofcontents
\part
\doparttoc
\dominitoc
\tableofcontents
\part
\doparttoc
\dominitoc
\tableofcontents
\part
\doparttoc
\dominitoc
\tableofcontents
\part
\doparttoc
\dominitoc
\tableofcontents
\part
\doparttoc
\dominitoc
\tableofcontents
\part
\doparttoc
\dominitoc
\tableofcontents
\part
\doparttoc
\dominitoc
\tableofcontents
\part
\doparttoc
\dominitoc
\tableofcontents

\subsection{The \texttt{mtc-fo1.tex} document file}

This document creates several copies of the same parttoc, but with different fonts (for the chapter level entries); you can compare the results.

\begin{itemize}
\item \texttt{\textbackslash documentclass\{report\}}
\item \texttt{\ProvidesFile\{mtc-fo1.tex\}}%
\item \texttt{\[2007/01/04\]}
\item \texttt{\usepackage\{lipsum\}}
\end{itemize}

\footnote{Here we only gave the commands for the section entries in the minitocs, but analog commands may be used for lower entries.}
4.15 The \texttt{mtc-fo2.tex} document file

This document creates several copies of the same parttoc, but with different fonts (for the chapter level entries); you can compare the results. As the fonts are not declared the same way, compare the results with those of \texttt{mtc-fo1.tex} (see section 4.14 on the preceding page).
We declare the fonts with the old method:

\def\ptcSSfont{\ptcfont} % (subsections)
\def\ptcSSSfont{\ptcfont} % (subsubsections)
\def\ptcPfont{\ptcfont} % (paragraphs)
\def\ptcSPfont{\ptcfont} % (subparagraphs)
\def\plffont{\ptcfont} % (figures)
\def\plfSfont{\ptcfont} % (subfigures)
\def\pltfont{\ptcfont} % (tables)
\def\pltSfont{\ptcfont} % (subtables)

\begin{document}
\part{Introduction}
\clearpage
\parttoc
A first version of the parttoc, with the fonts defined above:
\parttoc \clearpage

\mtcsetfont \parttoc
A second version of the parttoc, with chapter entries in a roman bold font:
\mtcsetfont\parttoc\{chapter\}{\normalsize\bfseries
\parttoc \clearpage

\mtcsetfont \parttoc
A third version of the parttoc, with chapter entries in a typewriter bold font:
\mtcsetfont\parttoc\{chapter\}{\normalsize\ttfamily\bfseries
\parttoc \clearpage

\mtcsetfont \parttoc
A fourth version of the parttoc, with chapter entries in a non bold typewriter font:
\mtcsetfont\parttoc\{*\}{\normalsize\ttfamily\mdseries
\parttoc \clearpage

\chapter{A very short chapter}
\minitoc
\lipsum[1]
\section{First section} \lipsum[2]
\subsection{Alpha} \lipsum[3] \subsection{Beta} \lipsum[4]
\section{Second section} \lipsum[5]
\subsection{Gamma} \lipsum[6] \subsection{Delta} \lipsum[7]
4.16 The mtc-gap.tex document file

The mtc-gap.tex document file is described in section 2.44 on page 76.

4.17 The mtc-hi1.tex document file

The mtc-hi1.tex document file is described in section 2.25 on page 68.

4.18 The mtc-hi2.tex document file

The mtc-hi2.tex document file is described in section 2.25 on page 69.

4.19 The mtc-hia.tex document file

This document shows the use of the minitoc package in a document where the entries for some tables must be hidden in the main list of tables. The document uses the article class.

\begin{document}
\listoftables
\section{First section}
\lipsum[1]
\begin{table}[hb]
\caption{My first visible table}
\end{table}
\lipsum[2]
\begin{table}[ht]
\caption{A second visible table}
\end{table}
\lipsum[3]
\mtchideinmainlot
\section{Second section}
\lipsum[1]
\begin{table}[hb]
\caption{My second visible table}
\end{table}
\lipsum[2]
\begin{table}[ht]
\caption{A third visible table}
\end{table}
\lipsum[3]

For the first hidden table, we add \mtchideinmainlot before its caption:
This document shows the use of the minitoc package in a document where the entries for some tables must be hidden in the main list of tables. The document uses the report class. 

\documentclass[a4paper]{report}
\ProvidesFile{mtc-hir.tex}[2007/01/04]
\usepackage{lipsum}
\usepackage[tight]{minitoc}
\begin{document}
\listoftables
\chapter{First chapter}
\lipsum[1]
\begin{table}[hb]
\caption{My first visible table}
\end{table}
\lipsum[2]
\begin{table}[ht]
\caption{A second visible table}
\end{table}
\lipsum[3]
\chapter{Second chapter}
\lipsum[8]
\begin{table}[hb]
\caption{A second hidden table}
\end{table}
\lipsum[4-6]
\begin{table}[ht]
\caption{A third visible table}
\end{table}
\lipsum[9]
\begin{table}[hb]
\caption{A fourth visible table}
\end{table}
\lipsum[10]
\begin{table}[ht]
\caption{A second hidden table}
\end{table}
\lipsum[7]
\begin{table}[hb]
\caption{My last hidden table}
\end{table}
\lipsum[4-6]
\begin{table}[ht]
\caption{A third visible table}
\end{table}
\lipsum[11]
\begin{table}[hb]
\caption{A fourth visible table}
\end{table}
\lipsum[12]
\end{document}
For the last hidden table, we add `\end{table}` after its caption:

\begin{table}[hb]
\caption{My last hidden table}
\end{table}

\lipsum[9]
\begin{table}[ht]
\caption{A third visible table}
\end{table}
\lipsum[10]
\begin{table}[hb]
\caption{A fourth visible table}
\end{table}
\lipsum[11]
\end{document}

4.21 The `mtc-hop.tex` document file

This document shows the use of the `minitoc` package in a document of class `scrbook`.

\documentclass[oneside,12pt]{scrbook}
\ProvidesFile{mtc-hop.tex}[2007/01/04]
\usepackage{lipsum}
\usepackage[hints]{minitoc}
\begin{document}
\dominitoc
\dominilof
\tableofcontents
\listoffigures

We prepare the minitocs and the minilofs, we print the TOC but not the LOF (while the LOF file is prepared):

\part*{Part 1: Strategic Marketing}
\mtcaddpart[Part 1: Strategic Marketing]

\chapter{Chapter 1}
\minitoc \minilof
\section{Section one of first chapter} \lipsum[1]
\begin{figure}
\centering Test
\caption{Picture one of first chapter}
\end{figure}
\section{Section two of first chapter} \lipsum[2]
4.22 The mtc-liv.tex document file

This document shows the use of the minitoc package in a document of book class, with customized TOC and minitocs.

First, we want that empty pages be really empty, without page number nor headers, so we redefine \cleardoublepage:

\makeatletter
\def\ps@chapterverso{\ps@empty}
\def\cleardoublepage{\clearpage
\if@twoside
\ifodd\c@page\else
\null\thispagestyle{chapterverso}\newpage
\if@twocolumn\null\newpage\fi
\fi
\fi
\def\ps@chapterverso{\ps@empty}
\makeatother
We define the encodings, for input and output, because the document is in french and uses accented letters:

\usepackage[latin1]{inputenc}
\usepackage[TS1,T1]{fontenc}

We load two packages, \texttt{tocloft} [469], to customize the TOC and the minitocs, and \texttt{sectsty} [319], to customize the sectionning commands:

\usepackage{tocloft}
\usepackage{sectsty}

We load the \texttt{minitoc} package then some complementary local packages for the french language:

\usepackage[french,undotted,tight]{minitoc}
\usepackage[english,francais]{babel}
\usepackage[franc,frnew,mypatches]{franc}
\providecommand\fup\textsuperscript
\addtolength\cftsubsecindent\cftsetrmarg\setcounter
\chapterfont\thesection\sectionfont\raggedright

We make some customizations: indentation for the subsection entries in the TOC and the minitocs, depth of the TOC, numerotation depth, depth of the minitocs, some fonts:

\addtolength{\cftsubsecindent}{1em} % for tocloft
\setcounter{tocdepth}{3}
\setcounter{secnumdepth}{1}
\setcounter{minitocdepth}{4}
\chapterfont{\huge\bfseries\sffamily}
\renewcommand{\thesection}{\arabic{section}}
\sectionfont{\Large\raggedright}

Some informations for the title page:

\title{Systèmes d'occultation} \author{Laurent~\textsc{Bloch}}

And the document body:\footnote{The text has been shortened, so there is an undefined reference; do not worry.}:

\begin{document}
\maketitle
\tableofcontents
\chapter{Définition et contrôle du travail à~faire} \label{chap+controle}
\section*{Définition et contrôle du travail à~faire}
C'est au \textsc{xviii}e siècle que la vision du travail comme marchandise est vraiment devenue dominante, pour s'imposer au \textsc{xix}e siècle dans l'organisation type de la grande usine industrielle.

Aujourd'hui le taylorisme au sens strict est en déclin parce qu'il n'est plus guère adapté aux besoins de la production industrielle contemporaine non plus qu'aux nouvelles normes de comportement individuel et collectif.

Le travail a vocation à produire du sens, pour son auteur comme pour son destinataire.

En France, les prestations de service commandées par les services publics à des entreprises font l'objet de contrôles de leur bonne réalisation selon des procédures et des règles qui sont des cas particuliers d'un ensemble plus vaste, la réglementation des marchés publics de l'État, dont nous allons donner ci-dessous une brève description.

Le dispositif juridique, réglementaire et comptable qui encadre les actes contractuels de la puissance publique en France est très-

Le premier élément du dispositif est le principe de séparation de l'ordonnateur et du comptable. Il a été instauré en 1319 par l'ordonnance portant création de la-

Le second élément du dispositif est le principe du contrôle \emph{a-priori}. Lorsque le directeur de l'organisme public de recherche pris ici comme exemple (l'ordonnateur) décide-

Le troisième pilier de la commande publique est le Code des Marchés (CMP), qui régit tous les contrats, conclus par des organismes publics ou des collectivités territoriales, dont le montant excède un-

Lorsque l'administration française fait réaliser un système informatique par un prestataire, elle est en position de maître d'ouvrage\index{maîtrise d'ouvrage}. Elle rédige (ou fait rédiger) un cahier des charges\index{cahier des charges} qui décrit les spécifications du système à-réaliser. Ce cahier des charges constitue-

Quels sont les services publics \textasciitilde rentables\textasciitilde?
Pour parler comme les informaticiens, nous pouvons identifier un « effet de bord », c'est-à-dire une conséquence non intentionnelle de la réglementation des marchés publics: les administrations ne disposent d'aucun moyen pour envisager la notion d'investissement. Le[...]

Jean-Pierre Boutinet nous guidera ici pour ce qui concerne l'histoire de la notion de[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivante: le maître d'ouvrage[...]

Nous y reviendrons au chapitre[ref{chap-travail}], mais nous savons déjà que la mise en œuvre de l'informatique s'est beaucoup inspirée des procédures de travail les plus élaborées du xx\textsc{e} siècle[...]

Jean-Pierre Boutinet nous guidera ici pour ce qui concerne l'histoire de la notion de l'informatique[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivante: le maître d'ouvrage[...]

Nous y reviendrons au chapitre[ref{chap-travail}], mais nous savons déjà que la mise en œuvre de l'informatique s'est beaucoup inspirée des procédures de travail les plus élaborées du xx\textsc{e} siècle[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivante: le maître d'ouvrage[...]

Nous y reviendrons au chapitre[ref{chap-travail}], mais nous savons déjà que la mise en œuvre de l'informatique s'est beaucoup inspirée des procédures de travail les plus élaborées du xx\textsc{e} siècle[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivante: le maître d'ouvrage[...]

Nous y reviendrons au chapitre[ref{chap-travail}], mais nous savons déjà que la mise en œuvre de l'informatique s'est beaucoup inspirée des procédures de travail les plus élaborées du xx\textsc{e} siècle[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivante: le maître d'ouvrage[...]

Nous y reviendrons au chapitre[ref{chap-travail}], mais nous savons déjà que la mise en œuvre de l'informatique s'est beaucoup inspirée des procédures de travail les plus élaborées du xx\textsc{e} siècle[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivante: le maître d'ouvrage[...]

Nous y reviendrons au chapitre[ref{chap-travail}], mais nous savons déjà que la mise en œuvre de l'informatique s'est beaucoup inspirée des procédures de travail les plus élaborées du xx\textsc{e} siècle[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivante: le maître d'ouvrage[...]

Nous y reviendrons au chapitre[ref{chap-travail}], mais nous savons déjà que la mise en œuvre de l'informatique s'est beaucoup inspirée des procédures de travail les plus élaborées du xx\textsc{e} siècle[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivante: le maître d'ouvrage[...]

Nous y reviendrons au chapitre[ref{chap-travail}], mais nous savons déjà que la mise en œuvre de l'informatique s'est beaucoup inspirée des procédures de travail les plus élaborées du xx\textsc{e} siècle[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivante: le maître d'ouvrage[...]

Nous y reviendrons au chapitre[ref{chap-travail}], mais nous savons déjà que la mise en œuvre de l'informatique s'est beaucoup inspirée des procédures de travail les plus élaborées du xx\textsc{e} siècle[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivante: le maître d'ouvrage[...]

Nous y reviendrons au chapitre[ref{chap-travail}], mais nous savons déjà que la mise en œuvre de l'informatique s'est beaucoup inspirée des procédures de travail les plus élaborées du xx\textsc{e} siècle[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivante: le maître d'ouvrage[...]

Nous y reviendrons au chapitre[ref{chap-travail}], mais nous savons déjà que la mise en œuvre de l'informatique s'est beaucoup inspirée des procédures de travail les plus élaborées du xx\textsc{e} siècle[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivante: le maître d'ouvrage[...]

Nous y reviendrons au chapitre[ref{chap-travail}], mais nous savons déjà que la mise en œuvre de l'informatique s'est beaucoup inspirée des procédures de travail les plus élaborées du xx\textsc{e} siècle[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivante: le maître d'ouvrage[...]

Nous y reviendrons au chapitre[ref{chap-travail}], mais nous savons déjà que la mise en œuvre de l'informatique s'est beaucoup inspirée des procédures de travail les plus élaborées du xx\textsc{e} siècle[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivante: le maître d'ouvrage[...]

Nous y reviendrons au chapitre[ref{chap-travail}], mais nous savons déjà que la mise en œuvre de l'informatique s'est beaucoup inspirée des procédures de travail les plus élaborées du xx\textsc{e} siècle[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivante: le maître d'ouvrage[...]

Nous y reviendrons au chapitre[ref{chap-travail}], mais nous savons déjà que la mise en œuvre de l'informatique s'est beaucoup inspirée des procédures de travail les plus élaborées du xx\textsc{e} siècle[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivante: le maître d'ouvrage[...]

Nous y reviendrons au chapitre[ref{chap-travail}], mais nous savons déjà que la mise en œuvre de l'informatique s'est beaucoup inspirée des procédures de travail les plus élaborées du xx\textsc{e} siècle[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivante: le maître d'ouvrage[...]

Nous y reviendrons au chapitre[ref{chap-travail}], mais nous savons déjà que la mise en œuvre de l'informatique s'est beaucoup inspirée des procédures de travail les plus élaborées du xx\textsc{e} siècle[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivante: le maître d'ouvrage[...]

Nous y reviendrons au chapitre[ref{chap-travail}], mais nous savons déjà que la mise en œuvre de l'informatique s'est beaucoup inspirée des procédures de travail les plus élaborées du xx\textsc{e} siècle[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivante: le maître d'ouvrage[...]

Nous y reviendrons au chapitre[ref{chap-travail}], mais nous savons déjà que la mise en œuvre de l'informatique s'est beaucoup inspirée des procédures de travail les plus élaborées du xx\textsc{e} siècle[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivante: le maître d'ouvrage[...]

Nous y reviendrons au chapitre[ref{chap-travail}], mais nous savons déjà que la mise en œuvre de l'informatique s'est beaucoup inspirée des procédures de travail les plus élaborées du xx\textsc{e} siècle[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivante: le maître d'ouvrage[...]

Nous y reviendrons au chapitre[ref{chap-travail}], mais nous savons déjà que la mise en œuvre de l'informatique s'est beaucoup inspirée des procédures de travail les plus élaborées du xx\textsc{e} siècle[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivante: le maître d'ouvrage[...]

Nous y reviendrons au chapitre[ref{chap-travail}], mais nous savons déjà que la mise en œuvre de l'informatique s'est beaucoup inspirée des procédures de travail les plus élaborées du xx\textsc{e} siècle[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivante: le maître d'ouvrage[...]

Nous y reviendrons au chapitre[ref{chap-travail}], mais nous savons déjà que la mise en œuvre de l'informatique s'est beaucoup inspirée des procédures de travail les plus élaborées du xx\textsc{e} siècle[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivante: le maître d'ouvrage[...]

Nous y reviendrons au chapitre[ref{chap-travail}], mais nous savons déjà que la mise en œuvre de l'informatique s'est beaucoup inspirée des procédures de travail les plus élaborées du xx\textsc{e} siècle[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivante: le maître d'ouvrage[...]

Nous y reviendrons au chapitre[ref{chap-travail}], mais nous savons déjà que la mise en œuvre de l'informatique s'est beaucoup inspirée des procédures de travail les plus élaborées du xx\textsc{e} siècle[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivante: le maître d'ouvrage[...]

Nous y reviendrons au chapitre[ref{chap-travail}], mais nous savons déjà que la mise en œuvre de l'informatique s'est beaucoup inspirée des procédures de travail les plus élaborées du xx\textsc{e} siècle[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivante: le maître d'ouvrage[...]

Nous y reviendrons au chapitre[ref{chap-travail}], mais nous savons déjà que la mise en œuvre de l'informatique s'est beaucoup inspirée des procédures de travail les plus élaborées du xx\textsc{e} siècle[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivante: le maître d'ouvrage[...]

Nous y reviendrons au chapitre[ref{chap-travail}], mais nous savons déjà que la mise en œuvre de l'informatique s'est beaucoup inspirée des procédures de travail les plus élaborées du xx\textsc{e} siècle[...]

La vision classique de la conduite d'un projet informatique de gestion est la suivant
Nous avons décrit ci-dessus le processus élémentaire de la programmation, celui qui consiste à écrire les instructions ou les expressions qui vont composer un programme.

La programmation structurée

Le premier courant de pensée qui associa la recherche d'une syntaxe claire et expressive à une organisation logique et commode des unités de programme fut la programmation structurée des années 1970, dont...

La programmation par objets

Après la programmation structurée vint un autre courant significatif: la programmation par objets, inventée en Norvège à la fin des années 1960 par l'équipe de...

Excès dans la pensée

Il y a eu beaucoup de verbiage autour de l'aptitude supposée du modèle de...

4.23 The mtc-mem.tex document file

This example shows the use of the minitoc package in a memoir class document. First, the preamble:

\documentclass[oneside]{memoir}
\ProvidesFile{mtc-mem.tex}
\usepackage{lipsum}
\usepackage[tight]{minitoc}
\begin{document}
\tableofcontents*

We use the starred form \tableofcontents* specific of the memoir class. Note that the \chapter command has two optional arguments in the memoir class.

\minitoc
\tableofcontents*

\chapter[oneA][oneB][oneC]
\section[oneA][oneB][oneC]
\subsection[oneA][oneB][oneC]
\subsubsection[oneA][oneB][oneC]

\chapter[twoA][twoB][twoC]
\section[twoA][twoB][twoC]
\subsection[twoA][twoB][twoC]
\subsubsection[twoA][twoB][twoC]

\end{document}
4.24 The mtc-mm1.tex document file

This example shows the use of the minitoc package in a memoir class document and shows some of the necessary adaptations for fonts. First, the preamble:

```latex
\providecommand\cftsecfont{\empty}
\providecommand\cftsubsecfont{\empty}
\renewcommand\cftsecfont{\normalsize\scshape}
\renewcommand\cftsubsecfont{\normalsize\scshape}
\mtcsetfont{minitoc}{section}{\normalsize\scshape} % <- no scshape
\mtcsetfont{minitoc}{subsection}{\normalsize\scshape}% <- no scshape
\mtcsettitlefont{minitoc}{\Large\scshape}
\mtcsettitle{minitoc}{Chapter Contents}
```

We inhibit some font commands of the memoir class:

```latex
\providecommand\cftsecfont{\empty}
\providecommand\cftsubsecfont{\empty}
\renewcommand\cftsubsecfont{\normalsize\rmfamily\scshape}
\mtcsettitlefont{\mtcsettitle}{\Large\scshape}
```

Then we redefine them:

```latex
\renewcommand\cftsecfont{\normalsize\scshape}
\renewcommand\cftsubsecfont{\normalsize\scshape}
\mtcsettitlefont{minitoc}{\normalsize\scshape} % <- no scshape
\mtcsettitlefont{minitoc}{subsection}{\normalsize\scshape}% <- no scshape
```

But if we use the font commands of the memoir class, it works!

```latex
\providecommand\cftsecfont{\empty}
\providecommand\cftsubsecfont{\empty}
\renewcommand\cftsubsecfont{\normalsize\rmfamily\scshape}
\mtcsettitlefont{\mtcsettitle}{\Large\scshape}
```

But for mini-table titles (font and text), we can use the minitoc commands:

```latex
\mtcsettitlefont{minitoc}{\Large\scshape}
\mtcsettitle{minitoc}{Chapter Contents}
```

The document body:
4.25 The mtc-mu.tex document file

This document shows the use of the minitoc package in a document, the minitoc being inserted in the text with the wrapfig package [18].

A chapter, with its minitoc set in a minipage, included in a wrapfigure environment on the half of the text width, with some vertical adjustments:

The remaining of the text:

The previous chapter examined many end-user programming environments and found that most contain cognitive programming gulfs. These gulfs were often created when programming environments used multiple notations, and could manifest themselves in a variety of usability problems, ranging from users being unable to understand...
a program representation, to not wanting to execute their programs.

Conversely, the previous chapter also found circumstances where multiple notations helped users understand programs. It concluded that there was a place for multiple notation programming environments, but developers had to be very careful to avoid creating programming gulfs.

This chapter introduces our programming environment, Mulspren. Mulspren was designed to avoid these gulfs and gain the potential benefits of multiple notations. Users program using two notations, one similar to English and one similar to conventional code. Changes in one notation are immediately reflected in the other notation, and users can move rapidly and seamlessly between the notations. This is programming using dual notations.

When the program is executed, both notations are animated.

Mulspren’s language signature is \LangSig{Re/Wr/Wa + Re/Wr/Wa + Wa}.

Papers describing Mulspren have been published in \cite{Wright02-2} and \cite{Wright03-3}.

\section{section 1}

4.26 The mtc-nom.tex document file

This document\(^6\) shows the interaction of the minitoc package with the nomencl package\([456]\), when this package uses its option intoc.

\documentclass[oneside]{book}
\ProvidesFile{mtc-nom.tex}\([2007/04/02]\)
\makenomenclature

We load the packages and prepare the nomenclature:
\usepackage[intoc]{nomencl}
\usepackage[tight]{minitoc}
\makenomenclature
\begin{document}
\dominitoc
\tableofcontents

We prepare the minitocs and the table of contents:
\dominitoc
\tableofcontents

\chapter{Angels}
\minitoc
\section{Main equations}
\begin{equation}
a = \frac{N}{A}\end{equation}
\nomenclature{$a$}{The number of angels per unit area}
\nomenclature{$N$}{The number of angels per needle point}
\nomenclature{$A$}{The area of the needle point}
\nomenclature{$\sigma$}{The total mass of angels per unit area}
\nomenclature{$m$}{The mass of one angel}
\nomenclature{$\sigma = m a$}{The equation follows easily.}
\printnomenclature \mtcfixnomenclature

It is derived from the example given in the documentation of nomencl.

\[\]
A second chapter, with its minitoc, a section, and an entry in the nomenclature. This entry will be present in the nomenclature printed above.

\chapter{Demons}
\minitoc
\section{False equations}
\begin{equation} i = \sqrt{-1} \end{equation}
\nomenclature{$i$}{The imaginary unit}
\end{document}

4.27 The mtc-ocf.tex document file

This document shows the use of the open and close features of the minitoc package to prepare a minitoc on three columns. The old package fullpage [144] is used to have a wide text area.

\documentclass[oneside]{book}
\ProvidesFile{mtc-ocf.tex}
[2007/04/02]
\usepackage{multicol} % to make multi-columns.
\usepackage[french]{babel}
\usepackage[latin1]{inputenc}
\usepackage[OT1,TS1,T1]{fontenc}
\usepackage{fullpage}
% to allow a page breaks before a section
\let\osection\section \def\section{\penalty-1\relax\osection}
\% \mtcsetfeature{minitoc}{open}{\vspace{-1ex}\begin{multicols}{3}}
\mtcsetfeature{minitoc}{close}{\end{multicols}\vspace{-1.5ex}}
As the multicol environment adds some vertical spacing before and after it, we added some corrections.
\mtcsetfeature{minitoc}{raggedcolumns}
\mtcsetfeature{minitoc}{flushcolumns}
As the number of sections is not a multiple of three (or some entries are long), we cannot always balance the columns nicely, so we use ragged columns, using the “before” and “after” features:
And the body of the document, a chapter with many sections, listed in a minitoc:

\begin{document}
\dominitoc
\tableofcontents
\begin{chapter}{Premier chapitre}
\minitoc
% A lot of sections
\section{Alfa}
\section{Bravo}
\section{Charlie}
\section{Delta}
\section{Echo}
\section{Fox-Trot}
\section{Golf}
\section{Hotel}
\section{India}
\section{Juliet}
\section{Kilo}
\section{Lima}
\section{Mike}
\section{November}
\section{Oscar}
\section{Papa}
\section{Quebec}
\section{Romeo}
\section{Sierra}
\section{Tango}
\section{Uniform}
\section{Victor}
\section{Whiskey}
\section{X-Ray}
\section{Yankee}
\section{Zulu}
\end{chapter}
\end{document}

4.28 The mtc-ofs.tex document file

This document shows the use of the \texttt{mtcsetoffset} command to shift a minitoc to the left, trying to put it along the left margin of the text. The open and close features of the minitoc package are also used to prepare the minitoc on three columns. The old package \texttt{fullpage} [144] is used to have a wide text area.

\begin{verbatim}
\documentclass[a4paper]{book}
\ProvidesFile{mtc-ofs.tex} [% 2007/04/17]
\usepackage{lipsum}
\usepackage{multicol}
\usepackage{fullpage}
\usepackage[a4paper]{geometry}
\usepackage[tight]{minitoc}
\setlength{\mtcindent}{0pt}
\mtcsetfeature{minitoc}{before}{\raggedcolumns}
\mtcsetfeature{minitoc}{open}{\kern1sp\vspace*{-.1ex}\begin{multicols}{4}\kern-2.5ex}}
\mtcsetfeature{minitoc}{close}{\end{multicols}\kern-2.ex}
\mtcsetfeature{minitoc}{after}{\flushcolumns}
\end{verbatim}

We remove the minitoc indentation and set up the open and close features:
We begin the document with a chapter and its minitoc:

\begin{document}
\dominitoc \faketableofcontents
\chapter{Introduction}
\minitoc

As the minitoc is not aligned on the left margin of the text, we set a negative offset and print again the minitoc:

\mtcsetoffset \mtcskip \minitoc

But then the minitoc does not use the full width of the text; it would be better to modify both the offset and the indentation, each by the half of the total correction:

\mtcsetoffset \setlength \mtcindent \mtcskip \minitoc

Then a lot of sections, with some text:

\section{Alfa} \lipsum[\arabic{section}]
\section{Bravo} \lipsum[\arabic{section}]
\section{Charlie} \lipsum[\arabic{section}]
\section{Delta} \lipsum[\arabic{section}]
\section{Echo} \lipsum[\arabic{section}]
\section{Fox-Trot} \lipsum[\arabic{section}]
\section{Golf} \lipsum[\arabic{section}]
\section{Hotel} \lipsum[\arabic{section}]
\section{India} \lipsum[\arabic{section}]
\section{Juliet} \lipsum[\arabic{section}]
\section{Kilo} \lipsum[\arabic{section}]
\section{Lima} \lipsum[\arabic{section}]
\section{Mike} \lipsum[\arabic{section}]
\section{November} \lipsum[\arabic{section}]
\section{Oscar} \lipsum[\arabic{section}]
\section{Papa} \lipsum[\arabic{section}]
\section{Quebec} \lipsum[\arabic{section}]
\section{Romeo} \lipsum[\arabic{section}]
\section{Sierra} \lipsum[\arabic{section}]
\section{Tango} \lipsum[\arabic{section}]
\section{Uniform} \lipsum[\arabic{section}]
\section{Victor} \lipsum[\arabic{section}]
\section{Whiskey} \lipsum[\arabic{section}]
\section{X-Ray} \lipsum[\arabic{section}]
\section{Yankee} \lipsum[\arabic{section}]
\section{Zulu} \lipsum[\arabic{section}]
\end{document}

⟨/mtc-ofs⟩
Note: if you add a sub-section in this example, the corresponding entry in the minitoc may stick out if it appears in the last column, and the offset and the indentation should then be corrected again.

4.29 The mtc-sbf.tex document file

This document shows the use of the minitoc package with a document containing subfigures (here with the subfigure package [130]). We show how to use minilofs and to adjust their depth.

The preamble loads the subfigure package and redefines the format of subfigure entries in the list of figures:

\l@subfigure
@dottedxline
ext@subfigure

\documentclass[12pt]{report}
\ProvidesFile{mtc-sbf.tex}%
[2007/01/04]
\usepackage{subfigure}
\makeatletter
\renewcommand{\l@subfigure}{\@dottedxxxline{\ext@subfigure}{2}{3.9em}{3.3em}}
\makeatother

We load the varioref [326] package (to have nice cross-references) and the minitoc package:

\usepackage{varioref}
\usepackage[tight]{minitoc}

\newcommand{\goodap}{}
\subfigtopskip
\subfigbottomskip
\setcounter{}

\begin{document}

We define some features for the layout of the subfigures, then the depth of the list of figures:

\mtcsetdepth
\mtcsetfont

\newcommand{\goodap}{%}
\hspace{\subfigtopskip}%
\hspace{\subfigbottomskip}%
\setcounter{}

\begin{document}

We define the depth of the mini-lists of figures, then some fonts:

\mtcsetdepth{minilof}{2}
\mtcsetfont{minitoc}{section}{\small\rmfamily\upshape\bfseries}
\mtcsetfont{partlof}{subfigure}{\small\rmfamily\slshape\bfseries}
\mtcsetfont{partlof}{figure}{\small\rmfamily\upshape\bfseries}
\mtcsetfont{minilof}{subfigure}{\small\rmfamily\slshape\bfseries}
\mtcsetfont{minilof}{figure}{\small\rmfamily\upshape\bfseries}
\% no tables in this document
\% \mtcsetfont{partlot}{subtable}{\small\rmfamily\slshape\bfseries}
\mtcsetfont{partlot}{table}{\small\rmfamily\upshape\bfseries}
\tableofcontents
\listoffigures

We prepare the minilofs, the table of contents and the list of figures:
\tableofcontents \listoffigures

\chapter
\minilof
\mtcskip
\mtcsetdepth

A chapter, with its minilof, twice but with different depths:
\chapter{First Chapter}
\minilof
\mtcskip
\mtcsetdepth{minilof}{1}
\minilof

A figure containing three subfigures and their captions:
\begin{figure}
\centering
\fbox{\begin{minipage}{3.5in}
\raggedright
\begin{center}
\subfigure[First]{\fbox{\hbox to 20mm{\vbox to 15mm{\vfil\null}\hfil}}}\hspace{\subfigtopskip}\hspace{\subfigbottomskip}
\subfigure[Second Figure]{\fbox{\hbox to 20mm{\vbox to 10mm{\vfil\null}\hfil}}}\\
\subfigure[Third]{\label{3figs-c}\fbox{\hbox to 20mm{\vbox to 10mm{\vfil\null}\hfil}}}\\
\caption{Three subfigures.}\label{3figs}
\end{center}
\vspace{4pt}\end{minipage}}
\end{figure}

\begin{figure}
\vref{3figs} contains two top 'subfigures' and \vref{3figs-c}.
\end{figure}

\setcounter

This document shows the use of the minitoc package with a KOMA-Script document class [343, 344, 399], scrreprt. Some precautions are needed, because these classes have specific interfaces with the TOC (class options and commands).
The `hyperref` package, if used, must be loaded \textit{before} \texttt{minitoc}:

\begin{verbatim}
\usepackage{hyperref}
\end{verbatim}

With a KOMA-Script class [343, 344, 399], use the \texttt{k-tight} package option in place of \texttt{tight}; as it is a document in german, use also a language package option:

\begin{verbatim}
\usepackage[k-tight,germanb]{minitoc}
\usepackage[germanb]{babel}
\end{verbatim}

\begin{verbatim}
\chapter{Test 1}
\label{cha:test-1}
\end{verbatim}

Text.

\begin{verbatim}
\section{Tabelle}
\label{sec:tabelle}
\begin{table}
\centering
\begin{longtable}{ll}
\hline
\underline{Ausbildungsbetrieb} & Kommanditgesellschaft \\
\underline{Ausbildender} & Hammer \\
Ausbildungsstätte & XXXXX Ort \\
\hline
\end{verbatim}
4.31 The \texttt{mtc-syn.tex} document file

This document shows the use of the \texttt{minitoc} package when the table of contents is not at the beginning of the document, but is preceded by some starred chapters.

\texttt{\texttt{\textbackslash dominitoc}} We have the preamble, then we invoke \texttt{\texttt{\textbackslash dominitoc}} to prepare the minitocs:

\begin{verbatim}
\documentclass[\texttt{a4paper,\texttt{twoside,\texttt{12pt}}]{\texttt{book}}
\ProvidesFile{\texttt{mtc-syn.tex}}\%\[2007/01/04\]
\usepackage{\texttt{minitoc}}
\begin{document}
\dominitoc
\mtcaddchapter
\section{Dedication}\% Added for a starred chapter\% without entry in the ToC
\mtcaddchapter\% Added for a starred chapter\% with an entry in the ToC
\mtcaddchapter\% Added for a starred chapter\% with an entry in the ToC
\mtcaddchapter\% Added for a starred chapter\% with an entry in the ToC
\mtcaddchapter\% Added for a starred chapter\% with an entry in the ToC
\end{document}
\end{verbatim}
The table of contents comes here. Looking at the document.log file shows that the minitoc files inserted after here are from one chapter to far: we add a correction.

The following chapters are normal (unstarred):

\chapter{Chapter One}
\minitoc
\section{Section 1} Some text.
\section{Section 2} Some text.
\chapter{Chapter Two}
\minitoc

\section{Another Section 1} Some more text.
\section{Another Section 2} Some more text.
\end{document}

\section{Another Section 1} Some more text.
\section{Another Section 2} Some more text.
\end{document}

4.32 The mtc-tbi.tex document file

This document shows the use of the minitoc package with a document using the tocbibind package [472].

We don’t want an entry for the TOC in the TOC: option nottoc for the tocbibind package:

As there is no entry for the TOC in the TOC, no correction is necessary; we comment out the usual correction 7:

A starred chapter with an entry in the TOC; we add it;

7 It is recommended to keep this comment; you could change your mind.
\chapter{Some normal chapters:}
\minitoc
\section{Title of chapter-1}
\section{as1} \section{as2}
\section{Title of chapter-2}
\section{bs1} \section{bs2}
\section{Title of chapter-3}
\section{cs1} \section{cs2}
\section{Title of chapter-4}
\section{ds1} \section{ds2}
\chapter*{Another starred chapter, with an entry in the TOC:}
\mtcaddchapter
\chapter*{Conclusion}
\mtcaddchapter[Conclusion]
\chapter*{Yet another starred chapter, with an entry in the TOC, but with starred sections, also listed in the TOC:}
\mtcaddchapter
\mtcaddchapter
\addcontentsline{toc}{section}{first appendix}
\addcontentsline{toc}{section}{second appendix}
\section*{first appendix}
\section*{second appendix}
\listoffigures \mtcaddchapter
The list of figures has an entry in the TOC (via the tocbibind package), so a correction must be applied:
\begin{thebibliography}{3}
\bibitem {s1}{title ...}
\end{thebibliography}
\adjustmtc
\adjustmtc
\langle /mtc-tbi \rangle
\section*{4.33 The mtc-tlc.tex document file}

This document shows the use of the minitoc package in a document of the article class. It is the example of [350, page 58], modernized.
4.34 The mtc-tlo.tex document file

The mtc-tlo.tex document file is described in section 2.46 on page 79.

4.35 The mtc-tsf.tex document file

This document\footnote{It is derived from one of the examples distributed with the subfig package [132].} shows the use of the minitoc package with a document containing subfigures (here with the subfig package [132]). We show how to use minilofs and to adjust their depth. The old package fullpage [144] is used to have a wide text area.
The preamble loads the `subfig` package and redefines the format of subfigure entries in the list of figures:

```
\documentclass{report}
\ProvidesFile{mtc-tsf}[2008/04/03]%
\usepackage{fullpage}
\usepackage[config=altsf]{subfig}
\usepackage[tight]{minitoc}
\newdimen\testtemp
\newcommand{\ru}[1]{\testtemp #1\advance\testtemp .5pt \divide\testtemp 2\hbox to \testtemp{\leaders\hbox to 1mm{\vrule height1mm depth0pt width.25pt\hfil}\hfil}\hbox to 0pt{\hss\vrule height3mm depth0pt width.25pt\hss}\hbox to \testtemp{\leaders\hbox to 1mm{\hfil\vrule height1mm depth0pt width.25pt}\hfil}}
```

This is utility code to make graduated rules and a box around a figure.

```
\newdimen\testtemp
\newcommand{\ru}[1]{\testtemp #1\advance\testtemp .5pt \divide\testtemp 2\hbox to \testtemp{\leaders\hbox to 1mm{\vrule height1mm depth0pt width.25pt\hfil}\hfil}\hbox to 0pt{\hss\vrule height3mm depth0pt width.25pt\hss}\hbox to \testtemp{\leaders\hbox to 1mm{\hfil\vrule height1mm depth0pt width.25pt}\hfil}}
```

The body of the document. We set the depth of the list of figures and prepare the minilofs and the list of figures:

```
\setcounter{lofdepth}{2} \dominilof \listoffigures
\chapter{Reference Test} \minilof \mtcsetdepth{minilof}{1}
\begin{figure}[ht]
\centering
\subfigure{\label{fig+A}\figbox{SUBFIGURE ONE:\(no opt)}}
\quad
\subfigure\[
\label{fig+B}\figbox{SUBFIGURE TWO:\(empty opt)}
\]
```

A chapter containing a figure with subfigures. We print its minilof twice, with different depths:
The \texttt{mtc-vti.tex} document file

The \texttt{mtc-vti.tex} example shows how to change the sectionnal titles when they appear in a mini-table: a section title (or a chapter title) can have variants in a parttoc or in a minitoc (similar effects are possible with figure ou table titles). Such entries are said “polymorphic” (section 1.4.13 on page 43). First, the preamble of the document, with utility packages:
For the demonstration, we will use a parttoc and a minitoc, so we must prepare them:
\input{parttoc}
\input{minitoc}

We will use a multi-form title for the first section: a form to appear in the parttoc ("Alfa in parttoc"), a form to appear in the minitoc of the chapter ("Alfa in minitoc"), a form to appear elsewhere ("Alfa out subtoc"), and a form as title at the beginning of the section ("Alfa the first section"). So we define a command \alfati using the flags \ifinparttoc and \ifinminitoc to select which title is used in each of its instances.

\begin{verbatim}
\newcommand{\alfati}{\ifinparttoc{Alfa in parttoc}\else\ifinminitoc{Alfa in minitoc}\else{Alfa out subtoc}\fi\fi}
\end{verbatim}

But we can define a more general macro, \varsecti, with three arguments for the three variants of a section title: 9

\begin{verbatim}
\DeclareRobustCommand{\varsecti}{\ifinparttoc[#1]\relax\else\ifinminitoc[#2]\relax\else[#3]\relax\fi\fi}
\end{verbatim}

Then the document with a table of contents, a part with its parttoc and a chapter with its minitoc. And another chapter. You can verify that the entry for the "Alfa" section varies in the main toc, the parttoc, the minitoc and the effective title of the section. Note that the variable title (here \alfati) should be defined before any use, like in the main toc or any minitable.

\begin{verbatim}
\tableofcontents
\part{Part-A}
\parttoc
\chapter{One}
\minitoc
\section[\protect{\alfati}]{Alfa the first section}
\lipsum[1]
\end{verbatim}

We can use the more general macro \varsecti \textsuperscript{10}, with its three arguments given when the section begins; but that macro must be protected (or look at the \texttt{makerobust} \textsuperscript{350} package by

\textsuperscript{9} You will eventually need to define similar macros for other sectioning commands or for figures or table titles; proceed with care from this model.

\textsuperscript{10} This macro is not part of the \texttt{minitoc} package, it is just an example.
Heiko Oberdiek), or declared “robust” by \DeclareRobustCommand as above:

\section{Bravo in parttoc}
\section{Bravo in minitoc}
\section{Bravo out of subtoc} % in maintoc and headers
\section{Bravo the second section} % local title
\lipsum[2]

\mtcpolymtoc
But is is even easier to use a “polymorphic” entry in the optionnal argument\footnote{They are no secttocs in a book-class document, but all the four arguments of \mtcpolymtoc must be specified, even empty!}:

\chapter{Two}
\minitoc
\section{mtcpolymtoc}
\section{Charlie in parttoc}
\section{Charlie in minitoc}
\section{Charlie in secttoc} % <- see/voir note
\section{Charlie out subtoc]
\lipsum[3]
\end{document}
Chapter 5

Messages

Contents

5.1 Introduction ..................................................... 151
5.2 Messages from the minitoc package .......................... 153
  5.2.1 Informative messages .................................. 153
  5.2.1.1 Informative messages for hints ................... 161
  5.2.2 Warning messages ....................................... 165
  5.2.2.1 Warning messages for hints ...................... 171
  5.2.3 Error messages ......................................... 187
5.3 Messages from the mtcoff package .......................... 201
  5.3.1 Warning messages ....................................... 201
5.4 Message from the mtcpatchmem package ...................... 204

Tables

5.1 Message identifiers ........................................... 152

5.1 Introduction

This chapter lists and comments the messages given by the minitoc package, and is associates, mtcoff and mtcpatchmem\(^1\). The table 5.1 on the following page lists all messages; in this table, you can click on a message identifier to find quickly its meaning.

- The first line of each message contains usually the name of the package and an unique identifier (this identifier may be useful to search in this chapter of the documentation,

\(^1\) The texts of the messages given in this chapter may slighty differ from the real text, because some messages contain variable elements and the layout may vary.
Table 5.1: Message identifiers (click on a message identifier to see its meaning).

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>F0001</td>
<td>F0002</td>
<td>F0003</td>
<td>F0004</td>
<td>F0005</td>
<td>F0006</td>
<td>F0007</td>
<td>F0008</td>
<td>F0009</td>
<td>F0010</td>
<td>I0000</td>
<td>I0001</td>
<td>I0002</td>
<td>I0003</td>
<td>I0004</td>
<td>I0005</td>
<td>I0006</td>
<td>I0007</td>
<td>I0008</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>W0001</td>
<td>W0002</td>
<td>W0003</td>
<td>W0004</td>
<td>W0005</td>
<td>W0006</td>
<td>W0007</td>
<td>W0008</td>
<td>W0009</td>
<td>W0010</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>W0011</td>
<td>W0012</td>
<td>W0013</td>
<td>W0014</td>
<td>W0015</td>
<td>W0016</td>
<td>W0017</td>
<td>W0018</td>
<td>W0019</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>W0021</td>
<td>W0022</td>
<td>W0023</td>
<td>W0024</td>
<td>W0025</td>
<td>W0026</td>
<td>W0027</td>
<td>W0028</td>
<td>W0029</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>W0031</td>
<td>W0032</td>
<td>W0033</td>
<td>W0034</td>
<td>W0035</td>
<td>W0036</td>
<td>W0037</td>
<td>W0038</td>
<td>W0039</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>W0041</td>
<td>W0042</td>
<td>W0043</td>
<td>W0044</td>
<td>W0045</td>
<td>W0046</td>
<td>W0047</td>
<td>W0048</td>
<td>W0049</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>W0051</td>
<td>W0052</td>
<td>W0053</td>
<td>W0054</td>
<td>W0055</td>
<td>W0056</td>
<td>W0057</td>
<td>W0058</td>
<td>W0059</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>W0061</td>
<td>W0062</td>
<td>W0063</td>
<td>W0064</td>
<td>W0065</td>
<td>W0066</td>
<td>W0067</td>
<td>W0068</td>
<td>W0069</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>W0071</td>
<td>W0072</td>
<td>W0073</td>
<td>W0074</td>
<td>W0075</td>
<td>W0076</td>
<td>W0077</td>
<td>W0078</td>
<td>W0079</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>W0081</td>
<td>W0082</td>
<td>W0083</td>
<td>W0084</td>
<td>W0085</td>
<td>W0086</td>
<td>W0087</td>
<td>W0088</td>
<td>W0089</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>W0091</td>
<td>W0092</td>
<td>W0093</td>
<td>W0094</td>
<td>W0095</td>
<td>W0096</td>
<td>W0097</td>
<td>W0098</td>
<td>W0099</td>
</tr>
</tbody>
</table>

but has no special meaning, except the leading letter: I for informative, W for warning, and E for error).

- Informative messages are written only in the document.log file; the prefix is F for the warning messages from the mtcoff package and M for the informative message from the mtcpatchmem package.
- Warning messages are shown on the screen (but often too quickly to be seen, because normally \TeX{} does not stop for warnings) and written in the document.log file.
- Error messages are shown on the screen and written in the document.log file, but \TeX{} stops, so you can ask for help by typing "h".
- In this chapter, some words in the messages are typeset in italic characters; they represent the variable parts of the message:
  - \texttt{ARG1} The first argument of the command.
  - \texttt{ARG2} The second argument of the command.
5.2 Messages from the minitoc package

5.2.1 Informative messages
Package minitoc Info: I0001
(minitoc) *** minitoc package, version 61 ***.

Remember the version of the package.

Package minitoc Info: I0002
(minitoc) Autoconfiguration of extensions.

The minitoc package tries to determine if short or long extensions for file names are used by the operating system.

Package minitoc Info: I0003
(minitoc) chapter level macros available.

The \chapter sectionning command is available, so you can use the mini-table commands at the chapter level, but not the mini-table commands at the section level.

Package minitoc Info: I0004
(minitoc) chapter level macros NOT available.

The \chapter sectionning command is not available, so you cannot use the mini-table commands at the chapter level, but, if the \section sectionning command is available, you can use mini-table commands at the section level.

Package minitoc Info: I0005
(minitoc) compatible with hyperref.

This version of minitoc is compatible with the hyperref package.
The auxiliary file for a mini-table is found empty (or inexistent) when minitoc tries to insert it. If the checkfiles option is active, it is skipped.

The horizontal rules will be present in the mini-tables of type ARG1.

The horizontal rules will be omitted in the mini-tables of type ARG1.

You have used the listfiles package option. A list of the minitoc auxiliary files is written in the document.maf file. It may be helpful to delete these files. See section 1.7 on page 52. This option is the default since version #48.

The LANGUAGE.mld file has been successfully loaded for the LANGUAGE language by the \mtcselectlanguage command at line LINE. The titles for the mini-tables are changed.

2 The english.mld file is always loaded first, to have english as default language.
The `\mtcselectlanguage` macro has successfully (indirectly) loaded the `LANGUAGE.mlo` minitoc object file.

The autoconfiguration has detected that your operating system is able to use long extensions; this will be the default.

The `\mtcsetdepth` macro changes the value of the specified depth counter and forces it to `VALUE`.

The `\mtcsetfeature` macro has redefined the internal macro `INTERNAL_NAME` with the given `SEQUENCE`.

The `\mtcsetfont` command redefines the (old style) `macro` by the given `SEQUENCE` of font commands.
The macro \mtcsetformat redefines an internal macro with the value given by its third argument.

A mini-table title is redefined via the \mtcsettitle macro.

The \mtcsettitlefont macro redefines the (old style) \INTERNAL_NAME macro which the given sequence SEQUENCE.

The hints package option has detected no potential problem.

The version of the memoir class is old. The minitoc package does not need to patch this class.
I0021
Package minitoc Info: I0021
(minitoc) Page numbers are activated
(minitoc) for the ARG1s on input line LINE.

The page numbers will be present in the mini-tables of type ARG1.

I0022
Package minitoc Info: I0022
(minitoc) Page numbers are inhibited
(minitoc) for the ARG1s on input line LINE.

The page numbers will be omitted in the mini-tables of type ARG1.

I0023
Package minitoc Info: I0023
(minitoc) part level macros available.

The \part sectionning command is available, so you can use the mini-table commands at the part level.

I0024
Package minitoc Info: I0024
(minitoc) PREPARING MINITOCs FROM FILE on input line LINE.

A \dominitoc command prepares the minitoc auxiliary files for minitocs from FILE.

I0025
Package minitoc Info: I0025
(minitoc) PREPARING PARTTOCS FROM FILE on input line LINE.

A \doparttoc command prepares the parttoc auxiliary files for parttocs from FILE.

I0026
Package minitoc Info: I0026
(minitoc) PREPARING SECTTOCS FROM FILE on input line LINE.

A \dosecttoc command prepares the secttoc auxiliary files for secttocs from FILE.
The version of the memoir class is recent. The minitoc package will try to patch it.

The \section sectioning command is available but the \chapter sectioning command is not available, so you can use the mini-table commands at the section level.

The \section sectioning command is not defined (by the document class), so the section level commands of the minitoc package are not available.

The memoir document class is used. The minitoc package tries to ensure compatibility.

The autoconfiguration has detected that your operating uses UNIX-like (long extensions) file names.
This version of the \chapter which is incompatible with the minitoc package.
We try to patch.

The \texttt{memoir} class uses a version of the \texttt{\chapter} command which needs to be corrected because its syntax has been changed. A patch is loaded.

An auxiliary file for a mini-table is written by a \texttt{minitoc} preparation command (like \texttt{\dominitoc}).

A \texttt{\dominilof} command prepares the \texttt{minilof} auxiliary files for \texttt{minilofs} from \texttt{FILE}.

A \texttt{\dopartlof} command prepares the \texttt{partlof} auxiliary files for \texttt{partlofs} from \texttt{FILE}.

A \texttt{\dosectlof} command prepares the \texttt{sectlof} auxiliary files for \texttt{sectlofs} from \texttt{FILE}. 
A \dominilot command prepares the minilot auxiliary files for minilots from \textit{FILE}.

A \dopartlot command prepares the partlot auxiliary files for partlots from \textit{FILE}.

A \dosectlot command prepares the sectlot auxiliary files for sectlots from \textit{FILE}.

5.2.1.1 Informative messages for hints

As you are using the \texttt{abstract} package with its \texttt{addtotoc} option, you should look at the \texttt{minitoc} package documentation for specific precautions. See section 2.27 on page 70.

As you are using the \texttt{amsbook} class, you should look at the \texttt{minitoc} package documentation for specific precautions. See section 2.24 on page 66.
As you are using also the `appendix` package, you should look at the `minitoc` package documentation for specific precautions. See section 2.20 on page 64.

As you are using also the `CLASS` class, you should look at the `minitoc` package documentation for specific precautions. See section 1.5.5 on page 50. The classes involved here are `scrbook`, `screeprt`, and `scratcl`, i.e., the KOMA-Script classes [343, 344, 399] compatible with `minitoc`.

As you are using the `memoir` class, you should look at the `minitoc` package documentation for specific precautions. See section 2.22 on page 65.

A minitoc preparation command has been invoked more than once.
As you are using also the `tocbibind` package, you should look at the `minitoc` package documentation for specific precautions. See section 1.5.5 on page 50.

As you are using also the `tocloft` package, you should look at the `minitoc` package documentation for specific precautions. See section 2.2.1 on page 64.

The `\mtcprepare` command invoke all the possible preparation commands, depending only on the document class and the available contents files. It does not known exactly what you want, so it can prepare too many mini-tables files.

As you have requested the `hints` package option (which is set by default), some “hints” are eventually given in the `document.log` file. You can find them easily by searching for the string “`minitoc(hints)`” with a text editor.
Package minitoc Warning: I0050
(minitoc) The required "LANGUAGE.mld" file is missing.
(minitoc) The "LANGUAGE" language option will not be available.
(minitoc) Please install it from a recent distribution
(minitoc) or from the CTAN archives.

The LANGUAGE.mld file has not been installed on your system. You should take it from a recent distribution or from the CTAN archives to complete your installation, else the LANGUAGE language option will not be available.

Package minitoc Warning: I0051
(minitoc) The required "LANGUAGE.mlo" file is missing.
(minitoc) The "LANGUAGE" language option will not be available.
(minitoc) Please install it from a recent distribution
(minitoc) or from the CTAN archives.

The LANGUAGE.mlo file has not been installed on your system. You should take it from a recent distribution or from the CTAN archives to complete your installation, else the LANGUAGE language option will not be available.

Package minitoc Info: I0052
(minitoc) \mtcsetoffset redefines the offset
(minitoc) "OFFSET" as "VALUE" on input line LINE.

The \mtcsetoffset macro changes the value of the specified offset and forces it to VALUE.

Package minitoc Info: I0053
(minitoc) You have loaded the PACK package;
(minitoc) please be aware that the minitoc package
(minitoc) facilities can not be used for new types
(minitoc) of floats defined by the PACK package

The minitoc package does not manage new types of floats defined via the float [302], floatrow [285], trivfloat [484] and rotfloat [420] packages.

3 As the trivfloat and rotfloat packages load the float package, this message will then appear twice!
5.2.2 Warning messages

Package minitoc Warning: W0001
(minitoc) \chapter and \section are undefined.
(minitoc) Cannot use \mtcfixglossary without optional argument [part].

The sectioning commands \chapter and \section are not defined (by the document class), hence the \mtcfixglossary macro cannot be used without an optional argument (try \part). This situation is very unlikely to happen, so also verify your document class.

Package minitoc Warning: W0002
(minitoc) \chapter and \section are undefined.
(minitoc) Cannot use \mtcfixindex without optional argument [part].

The sectioning commands \chapter and \section are not defined (by the document class), hence the \mtcfixindex macro cannot be used without an optional argument (try \part). This situation is very unlikely to happen, so also verify your document class.

Package minitoc Warning: W0003
(minitoc) \firstchapteris is an obsolete (ignored) command on input line LINE.

You have used an obsolete command (\firstchapteris). You should remove it.

Package minitoc Warning: W0004
(minitoc) \firstpartis is an obsolete (ignored) command on input line LINE.

You have used an obsolete command (\firstpartis). You should remove it.
Package minitoc Warning: W0005

(minitoc) \firstsectionis is an obsolete (ignored) command on input line LINE.

You have used an obsolete command (\firstsectionis). You should remove it.

Package minitoc Warning: W0006

(minitoc) \mtcfixglossary can only be used with the [part] optional argument, which becomes the default.

The \mtcfixglossary macro can only use [part] as optional argument (which becomes the default), because \chapter and \section are not defined.

Package minitoc Warning: W0007

(minitoc) \mtcfixindex can only be used with the [part] optional argument, which becomes the default.

The \mtcfixindex macro can only use [part] as optional argument (which becomes the default), because \chapter and \section are not defined.

Package minitoc Warning: W0008

(minitoc) No file FILE.
(minitoc) MINILOFS NOT PREPARED on input line LINE.

The FILE cannot be found, because it has not been created by a \minilof command. Please check if you have called \minilof in the correct sequence of commands.

Package minitoc Warning: W0009

(minitoc) No file FILE.
(minitoc) MINILOTS NOT PREPARED on input line LINE.

The FILE cannot be found, because it has not been created by a \minilot command. Please check if you have called \minilot in the correct sequence of commands.
Package minitoc Warning: W0010
(minitoc) No file FILE.
(minitoc) MINITOCs NOT PREPARED on input line LINE.

The FILE cannot be found, because it has not been created by a \dominitoc command. Please check if you have called \dominitoc in the correct sequence of commands.

Package minitoc Warning: W0011
(minitoc) No file FILE.
(minitoc) PARTLOFS NOT PREPARED on input line LINE.

The FILE cannot be found, because it has not been created by a \dopartlof command. Please check if you have called \dopartlof in the correct sequence of commands.

Package minitoc Warning: W0012
(minitoc) No file FILE.
(minitoc) PARTLOTS NOT PREPARED on input line LINE.

The FILE cannot be found, because it has not been created by a \dopartlot command. Please check if you have called \dopartlot in the correct sequence of commands.

Package minitoc Warning: W0013
(minitoc) No file FILE.
(minitoc) SECTTOCS NOT PREPARED on input line LINE.

The FILE cannot be found, because it has not been created by a \doparttoc command. Please check if you have called \doparttoc in the correct sequence of commands.

Package minitoc Warning: W0014
(minitoc) No file FILE.
(minitoc) SECTLOFS NOT PREPARED on input line LINE.

The FILE cannot be found, because it has not been created by a \dosectlof command. Please check if you have called \dosectlof in the correct sequence of commands.
Package minitoc Warning: W0015
(minitoc) No file $FILE$.
(minitoc) SECTLOTS NOT PREPARED on input line $LINE$.

The $FILE$ cannot be found, because it has not been created by a \dosectlot command. Please check if you have called \dosectlot in the correct sequence of commands.

Package minitoc Warning: W0016
(minitoc) No file $FILE$.
(minitoc) SECTTOCS NOT PREPARED on input line $LINE$.

The $FILE$ cannot be found, because it has not been created by a \dosecttoc command. Please check if you have called \dosecttoc in the correct sequence of commands.

Package minitoc Warning: W0017
(minitoc) no section or chapter level macros available
(minitoc) PLEASE VERIFY YOUR MAIN DOCUMENT CLASS.

The \chapter and \section sectionning commands are not defined. Your document class is likely without any sectionning command, so the minitoc package is pointless. Verify your main document class.

Package minitoc Warning: W0018

Package minitoc Warning: part level macros NOT available.

The \part sectionning command is not defined (by the document class), so the part level commands of the minitoc package are not available. It is a warning message because most classes with sectionning commands define the \part command, so you should verify which class you are using.

Package minitoc Warning: W0019
(minitoc) Short extensions (MSDOS-like) will be used.
(minitoc) ==> this version is configured for MSDOS-like (8+3) file names.
The autofiguration has found that the operating system uses file names with short extensions (8+3 scheme).

Package minitoc Warning: W0020  
(minitoc) You have forced the use of short extensions.

You have used the shortext package option to force the use of short extensions (8+3 scheme). This action limits the number of usable mini-tables of each kind and may be problematic if you have more than 99 mini-tables of the same kind. If your operating system allows for long extensions, do not use the shortext package option, except for testing purposes.

Package minitoc Warning: W0021  
(minitoc) Your version of latex.tex is obsolete.  
(minitoc) Trying to continue...

You are using an obsolete version of \LaTeX, but the minitoc package will still try to continue. It would be better to update your \LaTeX installation.

Package minitoc Warning: W0022  
(minitoc) Your version of latex.tex is very obsolete.  
(minitoc) Trying to continue... crossing fingers.

Your version of \LaTeX is very obsolete, and almost unusable with the minitoc package. You can try to continue the compilation, but you are urged to update your \LaTeX installation as soon as possible.

Package minitoc Warning: W0093  
(minitoc) Some ".mld" or ".mlo" files are missing in your installation.  
(minitoc) Search for the I0050 and I0051 info messages in the \jobname.log file.  
(minitoc) The full list of the missing language files is given in the W0094 warning message.  
(minitoc) Please install the missing files from a recent distribution  
(minitoc) or from the CTAN archives.
Some .mld or .mlo files have not been installed on your system. Search for the I0050 and I0051 info messages in the document.log file to find which files are missing. You can retrieve them from a recent distribution or from the CTAN archives to complete your installation, else some language options will not be available. The full list of the missing language files is given in the W0094 warning message.

---

Package minitoc Warning: W0094
(minitoc) Missing minitoc language file(s):
(minitoc) ...

---

Some .mld or .mlo files have not been installed on your system. The list is given in the message.

---

Package minitoc Warning: W0095
(minitoc) \chapter and \section are undefined.
(minitoc) Cannot use \mtcfixnomenclature without
(minitoc) optional argument [part].

The sectionning commands \chapter and \section are not defined (by the document class), hence the \mtcfixnomenclature macro cannot be used without an optional argument (try \part). This situation is very unlikely to happen, so also verify your document class.

---

Package minitoc Warning: W0096
(minitoc) \mtcfixnomenclature can only be used
(minitoc) with the [part] optional argument,
(minitoc) which becomes the default.

The \mtcfixnomenclature macro can only use [part] as optional argument (which becomes the default), because \chapter and \section are not defined.

---

Package minitoc Warning: W0098
(minitoc) +++ You have used the \nofiles command
(minitoc) in your preamble; all preparation commands
(minitoc) in the body of the document will be ignored.
You have used the `\nofiles` command in the preamble of your document; hence the preparation commands will be ignored in your document. Please verify that the mini-table auxiliary files are in their final state. See page 28.

5.2.2.1 Warning messages for hints

Some sectioning commands have been altered after the loading of the `minitoc` package. The `hyperref` package does that, but it is harmless. For other packages or user-made alterations, it is recommended to alter the sectioning commands only before loading the `minitoc` package. See section 2.17 on page 62.

The `hints` package option has detected some potential problems and written hints into the `document.log` file. You can search it for the "`minitoc(hints)`" string with a text editor.

You are using the `alphanum` package which is incompatible with the `minitoc` package. The compilation can continue, but the result could be unsatisfactory.

You are using the `amsart` document class which is incompatible with the `minitoc` package. The compilation can continue, but the result could be unsatisfactory.
Package minitoc(hints) Warning: W0027
(minitoc(hints)) --- The amsproc class is loaded.
(minitoc(hints)) It is incompatible
(minitoc(hints)) with the minitoc package.

You are using the amsproc document class which is incompatible with the minitoc package. The compilation can continue, but the result could be unsatisfactory.

Package minitoc(hints) Warning: W0028
(minitoc(hints)) --- The \chapter command is altered after minitoc.

Some packages alter the sectioning commands, like \chapter. Most of them should be loaded before the minitoc package. The hyperref package, even if it is loaded before the minitoc package (as recommended), alters the sectioning commands in an \AtBeginDocument, so this message is always printed when you use the hyperref package with minitoc, but then it is harmless.

Package minitoc(hints) Warning: W0029
(minitoc(hints)) Warning: --- The jura class is loaded.
(minitoc(hints)) It is incompatible
(minitoc(hints)) with the minitoc package.

You are using the jura document class which is incompatible with the minitoc package. The compilation can continue, but the result could be unsatisfactory.

Package minitoc(hints) Warning: W0030
(minitoc(hints)) --- The \part command is altered after minitoc.

Some packages alter the sectioning commands, like \part. Most of them should be loaded before the minitoc package. The hyperref package, even if it is loaded before the minitoc package (as recommended), alters the sectioning commands in an \AtBeginDocument, so this message is always printed when you use the hyperref package with minitoc, but then it is harmless.
You are using the `placeins` package, but without its `section` option, while `minitoc` is called with its `insection` option which implies it. See page 29, near a “dangerous bend” symbol like the one shown in the margin.

You are using an obsolete version of the `placeins` package. Please update it from the CTAN archives or a recent distribution.

The `caption` package alters some commands and must be loaded before the `minitoc` package. See section 2.31 on page 72.

You are using the `placeins` package, but without its `section` option, while `minitoc` is called with its `insection` option which implies it. Try to inverse the loading order and use consistent options. You may have got a message ! LaTeX Error: Option clash for package placeins.

You are using an obsolete version of the `placeins` package. Please update it from the CTAN archives or a recent distribution.

The `caption2` package alters some commands and must be loaded before the `minitoc` package. See section 2.31 on page 72. Note that the `caption2` package is now obsolete; please use a recent version of the `caption` package.
Package minitoc(hints) Warning: W0035
(minitoc(hints)) The ccaption package should be loaded BEFORE the minitoc package.

The ccaption package alters some commands and must be loaded before the minitoc package. See section 2.31 on page 72.

Package minitoc(hints) Warning: W0036
(minitoc(hints)) The mcaption package should be loaded BEFORE the minitoc package.

The mcaption package alters some commands and must be loaded before the minitoc package. See section 2.31 on page 72.

Package minitoc(hints) Warning: W0037
(minitoc(hints)) The sectsty package should be loaded BEFORE the minitoc package.

The sectsty package alters some commands and must be loaded before the minitoc package. See section 2.28 on page 70.

Package minitoc(hints) Warning: W0038
(minitoc(hints)) The varsects package should be loaded BEFORE the minitoc package.

The varsects package alters some commands and must be loaded before the minitoc package. See section 2.33 on page 73.

Package minitoc(hints) Warning: W0039
(minitoc(hints)) --- The \section command is altered after minitoc.
Some packages alter the sectioning commands, like `\section`. Most of them should be loaded before the minitoc package. The `hyperref` package, even if it is loaded before the minitoc package (as recommended), alters the sectioning commands in an `\AtBeginDocument`, so this message is always printed when you use the `hyperref` package with minitoc, but then it is harmless.

---

Package minitoc(hints) Warning: W0040
(minitoc(hints)) --- The titletoc package is loaded.
(minitoc(hints)) It is incompatible with the minitoc package.

You are trying to use also the titletoc package, but it is incompatible with the minitoc package. See note 21 on page 53.

---

Package minitoc(hints) Warning: W0041
(minitoc(hints)) You have attempted to insert empty minilos.

You have attempted to insert empty minilos. If you have used the nocheckfiles package option, you will get some ugly empty mini-tables, with only a title and two horizontal rules. By default (checkfiles package option), you will only get this harmless message.

---

Package minitoc(hints) Warning: W0042
(minitoc(hints)) You have attempted to insert empty minilots.

You have attempted to insert empty minilots. If you have used the nocheckfiles package option, you will get some ugly empty mini-tables, with only a title and two horizontal rules. By default (checkfiles package option), you will only get this harmless message.

---

Package minitoc(hints) Warning: W0043
(minitoc(hints)) You have attempted to insert empty minitocs.

You have attempted to insert empty minitocs. If you have used the nocheckfiles package option, you will get some ugly empty mini-tables, with only a title and two horizontal rules. By default (checkfiles package option), you will only get this harmless message.
Package minitoc(hints) Warning: W0044
(minitoc(hints)) You have attempted to insert empty partlofs.

You have attempted to insert empty partlofs. If you have used the nocheckfiles package option, you will get some ugly empty mini-tables, with only a title and two horizontal rules. By default (checkfiles package option), you will only get this harmless message.

Package minitoc(hints) Warning: W0045
(minitoc(hints)) You have attempted to insert empty partlots.

You have attempted to insert empty partlots. If you have used the nocheckfiles package option, you will get some ugly empty mini-tables, with only a title and two horizontal rules. By default (checkfiles package option), you will only get this harmless message.

Package minitoc(hints) Warning: W0046
(minitoc(hints)) You have attempted to insert empty parttocs.

You have attempted to insert empty parttocs. If you have used the nocheckfiles package option, you will get some ugly empty mini-tables, with only a title and two horizontal rules. By default (checkfiles package option), you will only get this harmless message.

Package minitoc(hints) Warning: W0047
(minitoc(hints)) You have attempted to insert empty sectlofs.

You have attempted to insert empty sectlofs. If you have used the nocheckfiles package option, you will get some ugly empty mini-tables, with only a title and two horizontal rules. By default (checkfiles package option), you will only get this harmless message.
Messages

Package minitoc(hints) Warning: W0048
(minitoc(hints)) You have attempted to insert empty sectlots.

You have attempted to insert empty sectlots. If you have used the nocheckfiles package option, you will get some ugly empty mini-tables, with only a title and two horizontal rules. By default (checkfiles package option), you will only get this harmless message.

Package minitoc(hints) Warning: W0049
(minitoc(hints)) You have attempted to insert empty secttocs.

You have attempted to insert empty secttocs. If you have used the nocheckfiles package option, you will get some ugly empty mini-tables, with only a title and two horizontal rules. By default (checkfiles package option), you will only get this harmless message.

Package minitoc(hints) Warning: W0050
(minitoc(hints)) You have invoked an obsolete (ignored) command: \firstchapteris.

You have used an obsolete command (\firstchapteris). You should remove it.

Package minitoc(hints) Warning: W0051
(minitoc(hints)) You have invoked an obsolete (ignored) command: \firstpartis.

You have used an obsolete command (\firstpartis). You should remove it.

Package minitoc(hints) Warning: W0052
(minitoc(hints)) You have invoked an obsolete (ignored) command: \firstsectionis.

You have used an obsolete command (\firstsectionis). You should remove it.
Package minitoc(hints) Warning: \texttt{W0053}
(minitoc(hints)) You have used short extensions and more than 99 chapters (NUMBER).

You have used short extensions (limited to 3 characters) and more than 99 chapters, so the number of the auxiliary file does not fit in the extension. \textit{NUMBER} is the number of effective chapters in your document. See section 1.9 on page 54.

Package minitoc(hints) Warning: \texttt{W0054}
(minitoc(hints)) You have used short extensions and more than 99 parts (NUMBER).

You have used short extensions (limited to 3 characters) and more than 99 parts, so the number of the auxiliary file does not fit in the extension. \textit{NUMBER} is the number of effective parts in your document. See section 1.9 on page 54.

Package minitoc(hints) Warning: \texttt{W0055}
(minitoc(hints)) You have used short extensions and more than 99 sections (NUMBER).

You have used short extensions (limited to 3 characters) and more than 99 sections, so the number of the auxiliary file does not fit in the extension. \textit{NUMBER} is the number of effective sections in your document. See section 1.9 on page 54.

Package minitoc(hints) Warning: \texttt{W0056}
(minitoc(hints)) You are using \texttt{\dosectlof} and/or \texttt{\sectlof} and \texttt{\sectlot}, hence the ‘‘insection’’ package option is recommended.

You are asking for mini-lists of figures or tables at the section level. But as floats (figures and tables) could drift somewhere outside the printing area of the text of the section, the sectlofs and sectlots can be rather strange. In order to have a better behaviour of these mini-tables, it may be useful to add the \texttt{insection} package option. See page 29.
Package minitoc(hints) Warning: W0057
(minitoc(hints)) You have used \minilof,
(minitoc(hints)) but not \dominilof.

You have attempted to insert some minilofs (via \minilof), but the minilofs have not been prepared (via \dominilof).

Package minitoc(hints) Warning: W0058
(minitoc(hints)) You have used \minilot,
(minitoc(hints)) but not \dominilot.

You have attempted to insert some minilots (via \minilot), but the minilots have not been prepared (via \dominilot).

Package minitoc(hints) Warning: W0059
(minitoc(hints)) You have used \minitoc,
(minitoc(hints)) but not \dominitoc.

You have attempted to insert some minitocs (via \minitoc), but the minitocs have not been prepared (via \dominitoc).

Package minitoc(hints) Warning: W0060
(minitoc(hints)) You have used \partlof,
(minitoc(hints)) but not \dopartlof.

You have attempted to insert some partlofs (via \partlof), but the partlofs have not been prepared (via \dopartlof).

Package minitoc(hints) Warning: W0061
(minitoc(hints)) You have used \partlot,
(minitoc(hints)) but not \dopartlot.

You have attempted to insert some partlots (via \partlot), but the partlots have not been prepared (via \dopartlot).
Package minitoc(hints) Warning: W0062
(minitoc(hints)) You have used \parttoc, but not \doparttoc.

You have attempted to insert some parttocs (via \parttoc), but the parttocs have not been prepared (via \doparttoc).

Package minitoc(hints) Warning: W0063
(minitoc(hints)) You have used \sectlof, but not \dosectlof.

You have attempted to insert some sectlofs (via \sectlof), but the sectlofs have not been prepared (via \dosectlof).

Package minitoc(hints) Warning: W0064
(minitoc(hints)) You have used \sectlot, but not \dosectlot.

You have attempted to insert some sectlots (via \sectlot), but the sectlots have not been prepared (via \dosectlot).

Package minitoc(hints) Warning: W0065
(minitoc(hints)) You have used \secttoc, but not \dosecttoc.

You have attempted to insert some secttocs (via \secttoc), but the secttocs have not been prepared (via \dosecttoc).

Package minitoc(hints) Warning: W0066
(minitoc(hints)) You have used \minilof, but not \listoffigures nor \fakelistoffigures.

You have tried to insert some minilofs (via \minilof), but the document.lof file is not available because you have not invoked \listoffigures nor \fakelistoffigures.
Package minitoc(hints) Warning: W0067
(minitoc(hints)) You have used \minilot but not \listoftables nor \fakelistoftables.

You have tried to insert some minilots (via \minilot), but the document.lot file is not available because you have not invoked \listoftables nor \fakelistoftables.

Package minitoc(hints) Warning: W0068
(minitoc(hints)) You have used \minitoc but not \tableofcontents nor \faketableofcontents.

You have tried to insert some minitocs (via \minitoc), but the document.toc file is not available because you have not invoked \tableofcontents nor \faketableofcontents.

Package minitoc(hints) Warning: W0069
(minitoc(hints)) You have used \partlof but not \listoffigures nor \fakelstoffigures.

You have tried to insert some partlofs (via \partlof), but the document.lof file is not available because you have not invoked \listoffigures nor \fakelstoffigures.

Package minitoc(hints) Warning: W0070
(minitoc(hints)) You have used \partlot but not \listoftables nor \fakelistoftables.

You have tried to insert some partlots (via \partlot), but the document.lot file is not available because you have not invoked \listoftables nor \fakelistoftables.
Package minitoc(hints) Warning: W0071
(minitoc(hints)) You have used \parttoc but not \tableofcontents nor \faketableofcontents. You have tried to insert some parttocs (via \parttoc), but the document.toc file is not available because you have not invoked \tableofcontents nor \faketableofcontents.

Package minitoc(hints) Warning: W0072
(minitoc(hints)) You have used \sectlof but not \listoffigures nor \fakelistoffigures. You have tried to insert some sectlofs (via \sectlof), but the document.lof file is not available because you have not invoked \listoffigures nor \fakelistoffigures.

Package minitoc(hints) Warning: W0073
(minitoc(hints)) You have used \sectlot but not \listoftables nor \fakelistoftables. You have tried to insert some sectlots (via \sectlot), but the document.lot file is not available because you have not invoked \listoftables nor \fakelistoftables.

Package minitoc(hints) Warning: W0074
(minitoc(hints)) You have used \secttoc but not \tableofcontents nor \faketableofcontents. You have tried to insert some secttocs (via \secttoc), but the document.toc file is not available because you have not invoked \tableofcontents nor \faketableofcontents.
Package minitoc(hints) Warning: W0075
(minitoc(hints)) You have used \doparttoc
(minitoc(hints)) but not \parttoc.

You have prepared some parttocs (via \doparttoc), but you never used one of them.

Package minitoc(hints) Warning: W0076
(minitoc(hints)) You have used \dopartlof
(minitoc(hints)) but not \partlof.

You have prepared some partlofs (via \dopartlof), but you never used one of them.

Package minitoc(hints) Warning: W0077
(minitoc(hints)) You have used \dopartlot
(minitoc(hints)) but not \partlot.

You have prepared some partlots (via \dopartlot), but you never used one of them.

Package minitoc(hints) Warning: W0078
(minitoc(hints)) You have used \dominitoc
(minitoc(hints)) but not \minitoc.

You have prepared some minitocs (via \dominitoc), but you never used one of them.

Package minitoc(hints) Warning: W0079
(minitoc(hints)) You have used \dominilof
(minitoc(hints)) but not \minilof.

You have prepared some minilofs (via \dominilof), but you never used one of them.
You have prepared some minilots (via \minilot), but you never used one of them.

You have prepared some secttocs (via \dosecttoc), but you never used one of them.

You have prepared some sectlofs (via \dosectlof), but you never used one of them.

You have prepared some sectlots (via \dosectlot), but you never used one of them.

You are using the placeins package, but with its above option, while minitoc is called with its insection option which is incompatible with it. See page 29, near a “dangerous bend” symbol.
Package minitoc(hints) Warning: W0085
(minitoc(hints)) --- The placeins package is loaded
(minitoc(hints)) but minitoc used the insection option
(minitoc(hints)) which is incompatible with it.
(minitoc(hints)) Try to remove the below option
(minitoc(hints)) and use consistent options.

You are using the `placeins` package, but with its `below` option, while `minitoc` is called with its `insection` option which is incompatible with it. See page 29, near a “dangerous bend” symbol.

Package minitoc(hints) Warning: W0086
(minitoc(hints)) The `fncychap` package should be
(minitoc(hints)) loaded BEFORE the `minitoc` package.

The `fncychap` package alters some commands and must be loaded before the `minitoc` package. See section 2.38 on page 75.

Package minitoc(hints) Warning: W0087
(minitoc(hints)) The `quotchap` package should be
(minitoc(hints)) loaded BEFORE the `minitoc` package.

The `quotchap` package alters some commands and must be loaded before the `minitoc` package. See section 2.39 on page 75.

Package minitoc(hints) Warning: W0088
(minitoc(hints)) The `romannum` package should be
(minitoc(hints)) loaded BEFORE the `minitoc` package.

The `romannum` package alters the numbering of some sectioning commands and must be loaded before the `minitoc` package. See section 2.40 on page 76.
Package minitoc(hints) Warning: W0089
(minitoc(hints)) The sfheaders package should be
(minitoc(hints)) loaded BEFORE the minitoc package.

The sfheaders package alters the sectioning commands and must be loaded before the minitoc package. See section 2.41 on page 76.

Package minitoc(hints) Warning: W0090
(minitoc(hints)) The alnumsec package should be
(minitoc(hints)) loaded BEFORE the minitoc package.

The alnumsec package alters the sectioning commands and must be loaded before the minitoc package. See section 2.42 on page 76.

Package minitoc(hints) Warning: W0091
(minitoc(hints)) The captcont package should be
(minitoc(hints)) loaded BEFORE the minitoc package.

The captcont package alters the caption commands and must be loaded before the minitoc package. See section 2.43 on page 76.

Package minitoc(hints) Warning: W0092
(minitoc(hints)) The hangcaption package should be
(minitoc(hints)) loaded BEFORE the minitoc package.

The hangcaption package alters some commands and must be loaded before the minitoc package. See section 2.47 on page 79.

Package minitoc(hints) Warning: W0097
(minitoc(hints)) --- The flowfram package is loaded.
(minitoc(hints)) It is incompatible
(minitoc(hints)) with the minitoc package.

You are using the flowfram package which is incompatible with the minitoc package, because it has its own definitions for minitocs. The compilation can continue, but the result could be unsatisfactory.
5.2.3 Error messages

! Package minitoc Error: E0001
(minitoc) But \part is undefined.
See the minitoc package documentation for explanation.
Type H <return> for immediate help.
\mtcfixglossary not usable

There are no adequate sectioning command available to use the \mtcfixglossary macro; even \part is undefined. Verify your document class.

! Package minitoc Error: E0002
(minitoc) But \part is undefined.
See the minitoc package documentation for explanation.
Type H <return> for immediate help.
\mtcfixindex not usable

There are no adequate sectioning command available to use the \mtcfixindex macro; even \part is undefined. Verify your document class.

! Package minitoc Error: E0003
(minitoc) Imbrication of mtchideinmainlof environments.
See the minitoc package documentation for explanation.
Type H <return> for immediate help.
\mtcfixindex not usable

The hiding in main LoF could be incorrect.
Some `mtchideinmainlof` environments are incorrectly imbricated (overlapping), so the hiding in the main list of figures will be strange.

\begin{verbatim}
! Package minitoc Error: E0004
(minitoc) Imbrication of mtchideinmainlot environments.
See the minitoc package documentation for explanation.
Type H <return> for immediate help.
? h
The hiding in main LoT could be incorrect
\end{verbatim}

Some `mtchideinmainlot` environments are incorrectly imbricated (overlapping), so the hiding in the main list of tables will be strange.

\begin{verbatim}
! Package minitoc Error: E0005
(minitoc) Imbrication of mtchideinmaintoc environments.
See the minitoc package documentation for explanation.
Type H <return> for immediate help.
? h
The hiding in main ToC could be incorrect
\end{verbatim}

Some `mtchideinmaintoc` environments are incorrectly imbricated (overlapping), so the hiding in the main table of contents will be strange.

\begin{verbatim}
! Package minitoc Error: E0006
(minitoc) LANGUAGE is not a known language,
(minitoc) LANGUAGE.mld not found.
(minitoc) Command ignored.
See the minitoc package documentation for explanation.
Type H <return> for immediate help.
? h
See the minitoc documentation.
Correct the source using a valid language name.
Press RETURN
\end{verbatim}

The \texttt{\textbackslash mtchideinmaintoc} macro has attempted to load the `LANGUAGE.mld` minitoc language definition file, but has not found it. First, verify the name of the language (likely to be misspelt), then check if your installation contains all the many distributed `.mld` files of the minitoc package, at the right place. If it is a local `.mld` file, it should be installed in the right place (in a local hierarchy) or be in the working directory.
! Package minitoc Error: E0007
\mtcselectlanguage macro has attempted to load indirectly the LANGUAGE.mlo minitoc language object file, but has not found it. First, verify the name of the language (likely to be misspelt), then check if your installation contains all the many distributed .mlo files of the minitoc package, at the right place. If it is a local .mlo file, it should be installed in the right place (in a local hierarchy) or be in the working directory.

! Package minitoc Error: E0008
\mtcsetdepth attempts to use an undefined counter (ARG1\depth). You are trying to set the depth for an inexistent or undefined type of mini-table. Verify the type given and the document class, and the loaded packages.

! Package minitoc Error: E0009
\mtcsetdepth has a wrong first argument (ARG1). It should be a mini-table type (parttoc...sectlot). Correct the source code. Type \texttt{\textbackslash h} and rerun \LaTeX
The first argument of the `\mtcsetdepth` macro is incorrect. It should be a type of mini-table (parttoc,...,sectlot).

```
! Package minitoc Error: E0010
(minitoc) \mtcsetdepth: Illegal type of table (ARG1).
See the minitoc package documentation for explanation.
Type \texttt{H <return>} for immediate help.
? h
Correct the source code.
Type <return> and rerun LaTeX
```

The first argument of the `\mtcsetdepth` macro is incorrect. It should be a mini-table type (parttoc,...,sectlot).

```
! Package minitoc Error: E0011
(minitoc) \mtcsetfeature has a wrong first argument
(minitoc) (ARG1).
(minitoc) It should be a mini-table type
(minitoc) (parttoc...sectlot).
See the minitoc package documentation for explanation.
Type \texttt{H <return>} for immediate help.
? h
Correct the source code.
Type <return> and rerun LaTeX
```

The first argument of the `\mtcsetfeature` macro is incorrect. It should be a mini-table type (parttoc,...,sectlot).

```
! Package minitoc Error: E0012
(minitoc) \mtcsetfeature has a wrong second argument
(minitoc) (ARG2).
(minitoc) It should be a feature param
(minitoc) (before, after, open, close, pagestyle).
See the minitoc package documentation for explanation.
Type \texttt{H <return>} for immediate help.
? h
Correct the source code.
Type <return> and rerun LaTeX
```

The second argument of the `\mtcsetfeature` macro is incorrect. It should be before, after, open, close, or thispagestyle.
! Package minitoc Error: E0013
\mtcsetfont has a wrong first argument
\mtcsetfont has a wrong first argument
It should be a mini-table type
\mtcsetfont has a wrong first argument
\mtcsetfont has a wrong first argument
See the minitoc package documentation for explanation.
Type H <return> for immediate help.
Correct the source code.
Type <return> and rerun LaTeX

The first argument of \mtcsetfont is incorrect; it should be the type of a mini-table (parttoc..., sectlot).

! Package minitoc Error: E0014
\mtcsetfont has a wrong second argument
\mtcsetfont has a wrong second argument
It should be a sectionning level
\mtcsetfont has a wrong second argument
\mtcsetfont has a wrong second argument
See the minitoc package documentation for explanation.
Type H <return> for immediate help.
Correct the source code.
Type <return> and rerun LaTeX

The second argument of \mtcsetfont is incorrect; it should be a sectionning level (i.e., a sectionning command without its backslash), like part ..., subparagraph.

! Package minitoc Error: E0015
\mtcsetformat has a wrong first argument
\mtcsetformat has a wrong first argument
It should be a mini-table type
\mtcsetformat has a wrong first argument
\mtcsetformat has a wrong first argument
See the minitoc package documentation for explanation.
Type H <return> for immediate help.
Correct the source code.
Type <return> and rerun LaTeX

The first argument of a \mtcsetformat macro is incorrect. It should be a mini-table type (parttoc..., sectlot).
\PackageError{minitoc}{\mtcsetformat has a wrong second argument}{ARG2}.
\PackageError{minitoc}{\mtcsetformat has a wrong second argument}{ARG2}.
See the minitoc package documentation for explanation.
\PackageError{minitoc}{\mtcsetpagenumbers has a wrong first argument}{ARG1}.
\PackageError{minitoc}{\mtcsetpagenumbers has a wrong second argument}{ARG2}.
See the minitoc package documentation for explanation.
\PackageError{minitoc}{\mtcsetpagenumbers has a wrong second argument}{ARG2}.
See the minitoc package documentation for explanation.
Correct the source code.
Correct the source code.
Correct the source code.
The second argument of the \texttt{mtcsetpagenumbers} must be a keyword chosen in the following lists:\footnote{0 and o are the letter O, 0 is the zero digit.}

- on, ON, yes, YES, y, Y, true, TRUE, t, T, vrai, VRAI, v, V, oui, OUI, o, O, +, and 1;
- off, OFF, no, NO, n, false, FALSE, faux, FAUX, f, F, non, NON, -, and 0.

\begin{verbatim}
! Package minitoc Error: E0019
\mtcsetrules has a wrong first argument
\mtcsetrules has a wrong second argument
\mtcsetrules has a wrong second argument
\end{verbatim}

The first argument of the \texttt{mtcsetrules} is incorrect. It should be a mini-table type (\texttt{parttoc}, ..., \texttt{sectlot}).
! Package minitoc Error: E0021
\mtcsettitle has a wrong first argument
\mtcsettitle{(ARG1)}.
It should be a mini-table type
(parttoc...sectlot).
See the minitoc package documentation for explanation.
Type H <return> for immediate help.
? h
Correct the source code.
Type <return> and rerun LaTeX

The first argument of a \mtcsettitle macro is incorrect; it should be a mini-table type (parttoc,...,sectlot).

! Package minitoc Error: E0022
\mtcsettitlefont has a wrong first argument
\mtcsettitlefont{(ARG1)}.
It should be a mini-table type
(parttoc...sectlot).
See the minitoc package documentation for explanation.
Type H <return> for immediate help.
? h
Correct the source code.
Type <return> and rerun LaTeX

The first argument of the \mtcsettitlefont must be a mini-table type. You likely misspelt it.

! Package minitoc Error: E0023
The macro \mtcsetfeature has incompatible
first (ARG1) and second (ARG2) arguments.
See the minitoc package documentation for explanation.
Type H <return> for immediate help.
? h
Correct the source code.
Type <return> and rerun LaTeX

The first and second arguments of the \mtcsetfeature macro are incompatible. You should verify them.
! Package minitoc Error: E0024
(minitoc) The macro \mtcsetfont has incompatible
(minitoc) first (ARG1) and second (ARG2) arguments.
See the minitoc package documentation for explanation.
Type H <return> for immediate help.
? h
Correct the source code.
Type <return> and rerun LaTeX

The \mtcsetfont macro takes a mini-table type as first argument, a sectioning level as second argument (or a star), and a sequence of font commands as third argument. The second argument must have a lower level than the first one (i.e., it is meaningless to specify the font for the chapter level entries for a minitoc or a secttoc).

! Package minitoc Error: E0025
(minitoc) The macro \mtcsetformat has incompatible
(minitoc) first (ARG1) and second (ARG2) arguments.
See the minitoc package documentation for explanation.
Type H <return> for immediate help.
? h
Correct the source code.
Type <return> and rerun LaTeX

The first and second arguments of a \mtcsetformat macro are incompatible. One is likely to be misspelt.

! Package minitoc Error: E0026
(minitoc) The optional argument of \mtcfixglossary
(minitoc) is wrong.
See the minitoc package documentation for explanation.
Type H <return> for immediate help.
? h
It must be omitted (chapter), or be part, chapter or section

The optional argument of the \mtcfixglossary macro is incorrect: it should be omitted (then it defaults to chapter) or be part, chapter, or section.
! Package minitoc Error: E0027
(minitoc) The optional argument of \mtcfixindex
(minitoc) is wrong.
See the minitoc package documentation for explanation.
Type \h <return> for immediate help.
? h
It must be omitted (chapter), or be part, chapter or section

The optional argument of the \mtcfixindex macro is incorrect: it should be omitted (then it defaults to chapter) or be part, chapter, or section.

! Package minitoc Error: E0028
(minitoc) Unable to patch the memoir class.
See the minitoc package documentation for explanation.
Type \h <return> for immediate help.
? h
So it remains incompatible. Sorry.

Your version of the memoir class is really incompatible with the minitoc package and cannot be automatically patched. Please update the memoir class and/or the minitoc package from the CTAN archives or a recent distribution.

! Package minitoc Error: E0029
(minitoc) Unbalanced \mtchideinmainlof environment.
See the minitoc package documentation for explanation.
Type \h <return> for immediate help.
? h
The hiding in main LoF could be incorrect

A \mtchideinmainlof environment is unbalanced, so the hiding in the main list of figures could be incorrect.

! Package minitoc Error: E0030
(minitoc) Unbalanced \mtchideinmainlot environment.
See the minitoc package documentation for explanation.
Type \h <return> for immediate help.
? h
The hiding in main LoT could be incorrect
A `mtchideinmainlot` environment is unbalanced, so the hiding in the main list of tables could be incorrect.

```latex
! Package minitoc Error: E0031
(minitoc) Unbalanced `mtchideinmainlot` environment.
See the minitoc package documentation for explanation.
Type H <return> for immediate help.
? h
The hiding in main ToC could be incorrect
```

A `mtchideinmainlot` environment is unbalanced, so the hiding in the main table of contents could be incorrect.

```latex
! Package minitoc Error: E0032
(minitoc) You are using the `\mtcloadmlo` command outside of a `.mld` file.
See the minitoc package documentation for explanation.
Type H <return> for immediate help.
? h
It will be ignored
```

It is *forbidden* to use the `\mtcloadmlo` macro outside of a `.mld` file (which is loaded via `\mtcselectlanguage`). The command is ignored.

```latex
! Package minitoc Error: E0033
(minitoc) The macro `\mtcsettitle` uses an illegal type of table (ARG1).
See the minitoc package documentation for explanation.
Type H <return> for immediate help.
? h
Correct the source code.
Type <return> and rerun LaTeX
```

The first argument of a minitoc macro is incorrect. It should be a type of mini-table, like `parttoc`, `partlof`, `partlot`, `minitoc`, `minilof`, `minilot`, `secttoc`, `sectlof`, or `sectlot`. 
The first argument of a minitoc macro is incorrect. It should be a type of mini-table, like parttoc, partlof, partlot, minitoc, miniloft, minilot, secttoc, sectlof, or sectlot.

The insection package option is intended for article-like document classes, to prevent floats from drifting out of their section. It is pointless for book-like or report-like document classes, where floats are contained in their chapter.

The english.mld language definition file can not be found. You should verify your installation of the minitoc package. As an interim solution, we provide the missing english titles.
! Package minitoc Error: E0037
(minitoc) The \COMMAND command is incompatible
(minitoc) with the document class.

See the minitoc package documentation for explanation.
Type H <return> for immediate help.
...

1.39 \dominitoc[r]

? h
Correct the source code.
Type <return> and rerun LaTeX

You have used a preparation or insertion command (\COMMAND) which is not available for
the document class you are using. Please verify that the document class is compatible with
minitoc and if the level of the mini-table is available in the document class (section-level
mini-tables are not available in book- or report-like classes, chapter-level mini-tables are not
available in article-like classes, mini-tables are not available in letter-like classes, etc.).

! Package minitoc Error: E0038
(minitoc) Your minitoc installation is incomplete.
(minitoc) A mandatory minitoc language object file,
(minitoc) LANGUAGE.mld, is not found.
(minitoc) We will try to continue with
(minitoc) current/default values.
Type H <return> for immediate help.
? h
See the minitoc documentation.
Please fix your minitoc installation.
Type <return> to continue

The mandatory LANGUAGE.mld language definition file can not be found. You should
verify your installation of the minitoc package. As an interim solution, we provide the
default english titles.
There are no adequate sectioning commands available to use the \mtcfixnomenclature macro; even \part is undefined. Verify your document class.

The optional argument of the \mtcfixnomenclature macro is incorrect: it should be omitted (then it defaults to chapter) or be part, chapter, or section.

You are trying to set the offset for an inexistent or undefined type of mini-table. Verify the type given and the document class, and the loaded packages.
5.3 Messages from the mtcoff package

The mtcoff package gives only warning messages; their numbers begin with F.

5.3.1 Warning messages
The \addstarredchapter command is specific of the minitoc package and simulated by the mtcoff package. If necessary, it should be replaced by the equivalent \addcontentsline{toc}{chapter}{...} command.

Package mtcoff Warning: F0002
(mtcoff) \addstarredpart{...} should be replaced
(mtcoff) by \addcontentsline{toc}{part}{...}
(mtcoff) on input line LINE.

The \addstarredpart command is specific of the minitoc package and simulated by the mtcoff package. If necessary, it should be replaced by the equivalent \addcontentsline{toc}{part}{...} command.

Package mtcoff Warning: F0003
(mtcoff) \addstarredsection{...} should be replaced
(mtcoff) by \addcontentsline{toc}{section}{...}
(mtcoff) on input line LINE.

The \addstarredsection command is specific of the minitoc package and simulated by the mtcoff package. If necessary, it should be replaced by the equivalent \addcontentsline{toc}{section}{...} command.

Package mtcoff Warning: F0004
(mtcoff) \mtcaddchapter{...} should be replaced
(mtcoff) by \addcontentsline{toc}{chapter}{...}
(mtcoff) on input line LINE.

The \mtcaddchapter command is specific of the minitoc package and simulated by the mtcoff package. If necessary, it should be replaced by the equivalent \addcontentsline{toc}{chapter}{...} command.

Package mtcoff Warning: F0005
(mtcoff) \mtcaddpart{...} should be replaced
(mtcoff) by \addcontentsline{toc}{part}{...}
(mtcoff) on input line LINE.

The \mtcaddpart command is specific of the minitoc package and simulated by the mtcoff package. If necessary, it should be replaced by the equivalent \addcontentsline{toc}{part}{...} command.
Package mtcoff Warning: F0006
(mtcoff) \mtcaddsection{...} should be replaced
(mtcoff) by \addcontentsline{toc}{section}{...}
(mtcoff) on input line LINE.

The \mtcaddsection command is specific of the minitoc package and simulated by the mtcoff package. If necessary, it should be replaced by the equivalent \addcontentsline{toc}{section}{...} command.

Package mtcoff Warning: F0007
(mtcoff) You should scan (backwards) your .log file to find some commands needing to be replaced if you decide to DEFINITELY stop using minitoc for this document. It is more wise to keep the \usepackage lines for minitoc and mtcoff and to comment out only one of them.

You have replaced the use of the minitoc package by its substitute mtcoff. It is recommended to keep the \usepackage lines for both minitoc and mtcoff and to comment out only one of them. If you decide to definitely stop using minitoc for this document, it is wise to scan (backwards) the document.log file (after a compilation using mtcoff) to locate some commands needing to be replaced.

Package mtcoff Warning: F0008
(mtcoff) The macro \kernafterSTRING should not be used out of context
(mtcoff) on line LINE.

You are using one of the \kernafterSTRING macros with the mtcoff package. The result may be unpredictable. You can only redefine these macros to adjust the position of the bottom rule of a type of minitables. Any other usage is meaningless without the minitoc package.

Package mtcoff Warning: F0009
(mtcoff) The macro \STRINGoffset should not be used out of context
(mtcoff) on line LINE.
You are using one of the `\STRINGoffset` macros with the `mtcoff` package. The result may be unpredictable. You can only redefine these macros to adjust the horizontal position of a type of minitables. Any other usage is meaningless without the `minitoc` package.

5.4 Message from the `mtcpatchmem` package

Package `mtcpatchmem` Info: M0001
Package `mtcpatchmem` Info: mtcpatchmem package to patch the memoir class.

You are using a version of the `memoir` class which needs a correction. This correction has been automatically loaded if necessary. Very recent versions should not need it anymore. See chapter 12 on page 465.
Chapter 6

Jargon

Contents

```
“.” . . . 206  E . . . . 215  J . . . . 220  O . . . . 232  T . . . . 238
A . . . . 207  F . . . . 215  K . . . . 220  P . . . . 233  U . . . . 240
B . . . . 209  G . . . . 218  L . . . . 220  Q . . . . 235  V . . . . 240
C . . . . 210  H . . . . 218  M . . . . 222  R . . . . 235  W . . . . 240
D . . . . 214  I . . . . 219  N . . . . 229  S . . . . 236  X . . . . 241
```

Tables

<table>
<thead>
<tr>
<th></th>
<th>Category codes</th>
<th>Encoding schemes implemented in</th>
<th>CJK</th>
<th>Standard document classes</th>
<th>Depths for sectioning commands</th>
<th>Various encodings</th>
<th>Most common font encodings</th>
<th>Most common font families</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.1</td>
<td></td>
<td>6.8</td>
<td>6.9</td>
<td>6.11</td>
<td>6.12</td>
<td>216</td>
<td>230</td>
<td>230</td>
</tr>
<tr>
<td>6.2</td>
<td></td>
<td>6.10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6.3</td>
<td></td>
<td>212</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6.4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6.6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6.7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

This chapter attempts to explain some terms used in this documentation, and describes some useful files and suffixes. Many data come from the documentations of various cited packages (often from the abstract), and from [4, 5, 137, 140, 266, 282, 289–291, 309, 310, 315, 330, 343, 344, 355, 356, 458].
The suffix of the name for an auxiliary file of a \LaTeXX document. It carries some information from a \LaTeXX run to the next.

The suffix for the name of a document class file, loaded via \texttt{\documentclass}.

The suffix of the name for a documented source file of a \LaTeXX package or class. This file is often associated with a \texttt{.ins} file to generate the package or class. Compiling a \texttt{.dtx} file with \LaTeXX generates the documentation.

The base suffix of the name for a minilof file when short extensions (suffixes) are used. The full suffix is \texttt{.Fnn} where \texttt{nn} is the absolute number of the minilof.

The base suffix of the name for a partlof file when short extensions (suffixes) are used. The full suffix is \texttt{.Gnn} where \texttt{nn} is the absolute number of the partlof.

The base suffix of the name for a sectlof file when short extensions (suffixes) are used. The full suffix is \texttt{.Hnn} where \texttt{nn} is the absolute number of the sectlof.

The suffix of the name for an installation file of a \LaTeXX package or class. When compiled with \LaTeXX, it extracts the files of the package or class from an \texttt{.dtx} file.

The suffix of the name of the “list of figures” file.

The suffix of the name of the log file (compilation report).

The suffix of the name of the “list of tables” file.

The base suffix of the name for a minitoc file when short extensions (suffixes) are used. The full suffix is \texttt{.Mnn} where \texttt{nn} is the absolute number of the minitoc.

The suffix of the name of the file generated by the \texttt{listfiles} package option. This file contains the list of the minitoc auxiliary files.

The suffix for the name of a minitoc language definition file. A minitoc language definition file contains the definitions for the mini-table titles in a given language.

The base suffix of the name for a minilof file when long extensions (suffixes) are used. The full suffix is \texttt{.mlfnn} where \texttt{nn} is the absolute number of the minilof.

The suffix for the name of a minitoc language object file. For some exotic languages, the encoding makes not easy to put directly the titles in a \texttt{.mld} file; hence the \texttt{.mld} file must load a \texttt{.mlo} file.

The base suffix of the name for a minilot file when long extensions (suffixes) are used. The full suffix is \texttt{.mltnn} where \texttt{nn} is the absolute number of the minilot.

The base suffix of the name for a parttoc file when short extensions (suffixes) are used. The full suffix is \texttt{.Pnn} where \texttt{nn} is the absolute number of the parttoc.
The base suffix of the name for a partlof file when long extensions (suffixes) are used. The full suffix is .plfnn where nn is the absolute number of the partlof.

.plt
The base suffix of the name for a partlot file when long extensions (suffixes) are used. The full suffix is .pltnn where nn is the absolute number of the partlot.

.ptc
The base suffix of the name for a parttoc file when long extensions (suffixes) are used. The full suffix is .ptcnn where nn is the absolute number of the parttoc.

.S
The base suffix of the name for a secttoc file when short extensions (suffixes) are used. The full suffix is .Snn where nn is the absolute number of the secttoc.

.stc
The base suffix of the name for a secttoc file when long extensions (suffixes) are used. The full suffix is .stcnn where nn is the absolute number of the secttoc.

.sty
The suffix for the name of a package file, loaded via \usepackage.

.T
The base suffix of the name for a minilot file when short extensions (suffixes) are used. The full suffix is .Tnn where nn is the absolute number of the minilot.

.tex
The suffix of the name of a \TeX or \LaTeX normal source file.

.toc
The suffix of the name of the “table of contents” file.

.U
The base suffix of the name for a partlot file when short extensions (suffixes) are used. The full suffix is .Unn where nn is the absolute number of the partlot.

.V
The base suffix of the name for a sectlot file when short extensions (suffixes) are used. The full suffix is .Vnn where nn is the absolute number of the sectlot.

absolute numbering
The auxiliary files for the mini-tables have a suffix containing an absolute number, i.e., the number is unique and always increasing from the first part, chapter or section; this has solved some obscure problems, and also made obsolete some commands, like \firstpartis, \firstchapteris, and \firstsectionis. The absolute numbering has been introduced in version #23.

abstract
The abstract package [470] (by Peter R. Wilson) needs some precautions if used with its addtotoc option.

adjustment
Some minitoc commands and environments are known as “adjustment commands” because they are used in some circumstances to “adjust” a counter or to alter the displaying of contents files. These commands and environments are \adjustptc, \adjustmtc, \adjuststc, \decrementptc, \decrementmtc, \decrementstc, \incrementptc, \incrementmtc, \incrementstc,
after A type of feature (see this term) which is executed after a given type of mini-table. Look at the documentation of the \mtcsetfeature command, in section 1.10 on page 48.

afterpage The afterpage package is used to add code to be executed after the next page break.

alnumsec The alnumsec package allows you to use alphanumeric section numbering, e.g. A. Introduction; III. International Law. Its output is similar to the alphanum package (part of the jura class), but you can use the standard \LaTeX sectioning commands. Thus it is possible to switch sectioning schemes easily. Greek letters, double letters (bb) and different delimiters around them are supported. It must be loaded before the minitoc package (see point 1.8 on page 54 and section 2.42 on page 76).

alphanum The alphanum package, which is part of the specialized jura class, by Felix Braun, is incompatible with the minitoc package.

\textbf{AMS} The American Mathematical Society. This society has developed some document classes: unfortunately, amsart and amsproc are incompatible with the minitoc package; amsbook is compatible but needs precautions.

amsart, amsart.cls A document class for articles, provided by the American Mathematical Society. Unfortunately, this class is incompatible with the minitoc package.

amsbook, amsbook.cls A document class for books, provided by the American Mathematical Society. This class is compatible with the minitoc package, but needs some precautions. See section 2.24 on page 66.

amsproc, amsproc.cls A document class for conference proceedings, provided by the American Mathematical Society. Unfortunately, this class is incompatible with the minitoc package.

Antomega Antomega (by Alexej M. Kryukov and Dmitry Ivanov) is a language support package for Lambda, based on the original omega.sty file of the Omega project. However, it provides some additional useful functionalities. Some languages definition files (.mld) use titles taken from Antomega: greek-mono.mld, greek-polydemo.mld, greek-polykatha.mld, latvian.mld, polish2.mld, russian2m.mld, russian2o.mld, and spanish3.mld.

appendices See appendix below.

appendix The appendix package provides various ways of formatting the titles of appendices. Also (sub)appendices environments are provided that can be used, for instance, for per chapter/section appendices. If this

1 http://www.ams.org
package is used with minitoc, some precautions are needed (see section 2.20 on page 64).

Arabi

Arabi [243] is a system (by Youssef J. Jabra) to prepare \LaTeX{} documents in the arabic or farsi languages. The titles in arabi.mld and farsi13.mld come from the arabic1.ldf and farsi1.ldf files of this system.

Arab\TeX

Arab\TeX{} [276, 277] is a package (by Klaus L. Lagally) to prepare \LaTeX{} documents in the arabic or hebrew languages. The titles in arab.mld (or arabic.mld), arab2.mld and hebrew.mld come from Arab\TeX{}, while those of hebrew2.mld come from babel [60, 61].

Arm\TeX

Arm\TeX{} [142] is a package (prepared by Serguei Dachian, Arnak Dalalyan and Vartan Akopian) to prepare \LaTeX{} documents in the armenian language. The titles in armenian.mld come from Arm\TeX{}.

article

A standard \LaTeX{} document class [282]. It has sectionning commands: \part and \section (and below), but not \chapter. It is compatible with the minitoc package and you can make mini-tables at the part and section levels (but, of course, not at the unavailable chapter level).

AtBeginDocument

This standard macro allows to add code to be executed at the beginning of the document (if fact, at the very end of its preamble, but inside it, which implies some restrictions), at the point where \begin{document} is processed. This allows a package (or a class) to add code without creating any conflicts with other packages trying to do the same.

AtEndDocument

This standard macro allows to add code to be executed at the end of the document, at the point where \end{document} is processed. This allows a package (or a class) to add code without creating any conflicts with other packages trying to do the same.

autoconfiguration

Since version #28, minitoc detects automatically if the extensions (suffixes) of the file names are limited to 3 characters (like under MS-DOS) or not. This process is named autoconfiguration. The package option shortext forces the limitation to 3 characters.

auxiliary

During the preparation of a document, the \LaTeX{} system uses some auxiliary files to store information. The standard auxiliary files are \texttt{document.aux} (for cross-reference labels, counters, etc.), \texttt{document.toc} for the table of contents, \texttt{document.lof} for the list of figures, and \texttt{document.lot} for the list of tables. The minitoc package creates its own auxiliary files, to store the contents of each mini-table. These files are the minitoc auxiliary files, whose names are \texttt{document.extension}, the table 1.11 on page 55 lists the possible extensions. See also the .maf extension above.

babel

The babel package [60, 61] (by Johannes L. Braams and others) is a large package useful to write \LaTeX{} documents in many languages, not only english. Many titles for mini-tables come directly from the babel package.
**BangTeX** A package for typesetting documents in the bangla (bengali) language using the \TeX/\LaTeX systems; see [362].

**before** A type of *feature* (see this term) which is executed *before* a given type of mini-table. Look at the documentation of the `\mtcsetfeature` command, in section 1.10 on page 48.

**BibTEX** A program by Oren Patashnik to make bibliographies in \LaTeX documents. Distributed with \LaTeX. See [315, 366, 367, 417].

**bibtopic** A \LaTeX package [25] for including several bibliographies in a document. These bibliographies might be considered to cover different topics (hence the name) or bibliographic material (e.g., primary and secondary literature) and the like.

**bibunits** The bibunits package [210] allows separate bibliographies for different units or parts of the text. The units can be chapters, sections or bibunit environments. This package is compatible with a wide variety of packages, including, but not limited to, natbib [145, 146], overcite [17] and KOMA-Script classes [343, 344, 399].

**book** A standard \LaTeX document class [282]. It has sectioning commands: `\part`, `\chapter`, and `\section` (and below). It is compatible with the minitoc package and you can make mini-tables at the part and chapter levels (but not at the section level, to avoid too many auxiliary files).

**booktabs** This nice package [165] helps to the preparation of better tables, *without* vertical rules nor double rules.

**calc** The `calc` [441] package makes easier the numeric computations (on counters and dimensions) when preparing a \LaTeX document.

**cappuccino** See “minutes” below.

**captcont** The `captcont` package [131] provides support for retaining a figure or caption number across several float environments — usually over several pages. It allows control over the contents of the List-of-Figures and the List-of-Tables pages. It should be compatible with all other packages that modify or extend the float environment and with the subfig package [132] in particular.

**caption** The `caption` package [421, 422, 424] provides many ways to customize the captions in floating environments such `figure` and `table` and cooperates with many other packages.

**caption2** The `caption2` package [423] used to be an experimental side-version of the regular `caption` package [421, 422, 424] and has been superseded by the new release of the regular `caption` package version 3.0 in December 2003. `caption2` is still

---

This text comes from the documentation of the `caption` package. The `caption` and `caption2` packages have the same author, Axel Sommerfeldt.
Table 6.1: Category codes

<table>
<thead>
<tr>
<th>Category</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0  Escape character</td>
<td>( \ usually)</td>
</tr>
<tr>
<td>1  Beginning of group</td>
<td>( { usually)</td>
</tr>
<tr>
<td>2  End of group</td>
<td>( ) usually)</td>
</tr>
<tr>
<td>3  Math shift</td>
<td>( $ usually)</td>
</tr>
<tr>
<td>4  Alignment tab</td>
<td>( &amp; usually)</td>
</tr>
<tr>
<td>5  End of line</td>
<td>( return usually)</td>
</tr>
<tr>
<td>6  Parameter</td>
<td>( # usually)</td>
</tr>
<tr>
<td>7  Superscript</td>
<td>( ^ usually)</td>
</tr>
<tr>
<td>8  Subscript</td>
<td>( _ usually)</td>
</tr>
<tr>
<td>9  Ignored character</td>
<td>( null usually)</td>
</tr>
<tr>
<td>10 Space</td>
<td>( _ usually)</td>
</tr>
<tr>
<td>11 Letter</td>
<td>( A, . . . , Z and a, . . . , z)</td>
</tr>
<tr>
<td>12 Other character</td>
<td>( none of the above or below)</td>
</tr>
<tr>
<td>13 Active character</td>
<td>( ~ usually)</td>
</tr>
<tr>
<td>14 Comment character</td>
<td>( % usually)</td>
</tr>
<tr>
<td>15 Invalid character</td>
<td>( delete usually)</td>
</tr>
</tbody>
</table>

Some kind of supported, that means it will be part of future releases and bugs will still be fixed, so existing documents using this package will still compile. But Axel Sommerfeldt will not answer questions about this package anymore except questions on migrating to the regular version of the caption package. And it will not be adapted or enhanced in the future.

So please don’t use this package for new documents. It’s old, it’s obsolete and it starts to begin smell bad!

Please ignore all hints in books or other documents which try to tell you that the caption2 package should be used instead of the caption package – these hints are outdated since December 2003.

catcode Short for “category code”. In the first place, it’s wise to have a precise idea of what your keyboard sends to the machine. There are 256 characters that \TeX{} might encounter at each step, in a file or in a line of text typed directly on your terminal. These 256 characters are classified into 16 categories numbered 0 to 15. See table 6.1. It’s not necessary for you to learn these code numbers; the point is only that \TeX{} responds to 16 different types of characters. At first, “The \TeX{}book” led you to believe that there were just two types — the escape character and the others — and then you were told about two more types, the grouping symbols \{ and \}. The category code for any character can be changed at any time, but it is usually wise to stick to a particular scheme.

caption The ccaption package [474] provides commands for “continuation captions”, unnumbered captions, and a legend heading for any environment. Methods are provided to define captions for use outside float environments, and to define new float environments and subfloats. Tools are provided for defining your own captioning styles.

3 This definition is taken from “The \TeX{}book” [263, 265].
Table 6.2: Encoding schemes implemented in CJK

<table>
<thead>
<tr>
<th>Encoding</th>
<th>1 byte</th>
<th>2 bytes</th>
<th>3 bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>GB</td>
<td>0xA1–0xFF</td>
<td>0xA1–0xFE</td>
<td>—</td>
</tr>
<tr>
<td>Big 5</td>
<td>0xA1–0xFF</td>
<td>0x40–0xFE</td>
<td>—</td>
</tr>
<tr>
<td>JIS</td>
<td>0xA1–0xFF</td>
<td>0xA1–0xFE</td>
<td>—</td>
</tr>
<tr>
<td>SJIS</td>
<td>0xA1–0xFE</td>
<td>0x40–0xFC</td>
<td>—</td>
</tr>
<tr>
<td>KS</td>
<td>0xA1–0xFD</td>
<td>0xA1–0xFE</td>
<td>—</td>
</tr>
<tr>
<td>UTF 8</td>
<td>0xC0–0xEF</td>
<td>0x80–0xBF</td>
<td>0x80–0xBF</td>
</tr>
<tr>
<td>CNS</td>
<td>0xA1–0xFE</td>
<td>0xA1–0xFE</td>
<td>—</td>
</tr>
</tbody>
</table>

chkpage The chkpage package (by Peter R. Wilson) provides commands to change the page layout in the middle of a document, and to robustly check for typesetting on odd or even pages.

Chapter 0 Some documents do not begin with chapter number one, but with chapter number zero (or even a weirder number). This caused a serious problem in old versions of the minitoc package: the minitocs appeared in the wrong chapters, and a first correction was the introduction of specific commands (\firstchapter and co.). With the addition of the absolute numbering of the mini-table auxiliary files (see absolute above), the problem was solved in minitoc version #23, and these commands became obsolete. See section 1.5.4 on page 49.

chapterbib The chapterbib package [19] allows multiple bibliographies in a \LaTeX document, including items \cite’d in more than one bibliography. Despite the name “chapterbib”, the bibliographies are for each included file, not necessarily for each chapter.

checkfiles A package option of minitoc. It checks every mini-table to look if it is empty; then empty mini-tables are not printed. This is the default. The opposite package option (nocheckfiles) prints even the empty mini-tables, which look ugly. See section 9.77.2 on page 408.

CJK The CJK system [127, 297, 298] (by Werner Lemberg and others), is a set of packages and fonts to prepare \LaTeX documents in some oriental language, like chinese, japanese, korean (with Hangul or Hanja fonts), and thai, plus some variants of russian. The titles of mini-tables for these languages come from some CJK files and were inserted in .mld files when possible, or in .mlo files when the encoding is incompatible with the .ins/.dtx mechanism; then the .mlg file must input the corresponding .mlo file. CJK implements the GB, Big 5, JIS, SJIS, KS, UTF 8, and CNS encodings (on 16 bits, except UTF 8 on 24 bits). See table 6.2.

Some encoding schemes (Big 5, SJIS) have gaps in the range of the second byte. It is difficult to input Big 5 and SJIS encoding directly into \LaTeX since some of the values used for the encodings’ second bytes are reserved for control characters: ‘{’, ‘}’, and ‘\’. Redefining them breaks a lot of things in \LaTeX; to avoid this, preprocessors are normally used which convert the second byte into a number followed by a delimiter character. For further details, please refer to [309, 310]: Ken Lundøy discusses in great detail all CJK encodings which are or have been in use. Please note that the minitoc package uses the .mlo files as a workaround for this problem; see section 1.4.14 on page 44.
Table 6.3: Standard document classes

<table>
<thead>
<tr>
<th>Class</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>article</td>
<td>For articles in scientific journals, presentations, short reports, program documentation, invitations, . . .</td>
</tr>
<tr>
<td>proc</td>
<td>For preparing conference proceedings; analog to the article class.</td>
</tr>
<tr>
<td>ltxdoc</td>
<td>For preparing the documentation of a package or of a class; analog to the article class.</td>
</tr>
<tr>
<td>ltnews</td>
<td>For preparing the announcement of a \LaTeX\ release; analog to the article class.</td>
</tr>
<tr>
<td>report</td>
<td>For longer reports containing several chapters, small books, PhD theses, . . .</td>
</tr>
<tr>
<td>book</td>
<td>For real books.</td>
</tr>
<tr>
<td>letter</td>
<td>For letters; as this class has no sectioning commands, do not use \minitoc with this class.</td>
</tr>
<tr>
<td>slides</td>
<td>For slides; the class uses big sans serif letters. You might want to consider using Beamer\TeX\ instead. Do not use \minitoc with these classes.</td>
</tr>
</tbody>
</table>

\documentclass[options]{class}

Here \texttt{class} specifies the type of document to be created. Table 6.3 lists the standard document classes [282]. The \LaTeX\ 2ε distribution provides additional classes for other documents, including letters and slides, but the \minitoc\ package has not been tested with all these classes. The \texttt{options} parameters customize the behaviour of the document class. The options have to be separated by commas. The standard classes supported by the \minitoc\ package are listed in section 2.7 on page 60.

\section{close}

A type of \texttt{feature} (see this term) which is executed immediately after (\texttt{close}) the insertion of the auxiliary file for a given type of mini-table. Look at the documentation of the \texttt{\mtcsetfeature} command, in section 1.10 on page 48. See the \texttt{mtc-ocf.tex} example file, in section 4.27 on page 137.

\section{cmk}

An example of shell script to prepare the documentation files in PostScript format from the ones in PDF format. This script should be adapted to your needs.

\section{CMR}


\section{comp.text.tex}

The Usenet news group about \TeX\ and \LaTeX, in english.

\section{Computer Modern}

A set of fonts [262] designed by Donald E. Knuth for \TeX. Initially they were built with \texttt{METAFONT} [149, 262] (a program also created by Knuth), but PostScript type 1 (vector) versions exist today, with extensions (for accented characters, mainly): the EC-fonts (European Computer Modern), the cm-super fonts, etc.

\footnote{This note is extracted from [356], then adapted.}
Table 6.4: Depths for sectioning commands

<table>
<thead>
<tr>
<th>Class</th>
<th>book</th>
<th>report</th>
<th>article</th>
</tr>
</thead>
<tbody>
<tr>
<td>secnumdepth</td>
<td>2</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>\part</td>
<td>-1</td>
<td>-1</td>
<td>0</td>
</tr>
<tr>
<td>\chapter</td>
<td>0</td>
<td>0</td>
<td>×</td>
</tr>
<tr>
<td>\section</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>\subsection</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>\subsubsection</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>\paragraph</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>\subparagraph</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
</tbody>
</table>

counter A \TeX register containing an integer value. There are 256 counters (from 0 to 255) in \TeX, but \L\TeX uses some of them, and many packages need some counters for their own usage. An extended version derived from \TeX, \ɛ\TeX [105], allows more counters. Omega (\Ω) also offers more counters.

CTAN The Comprehensive \TeX Archive Network, a set of computer archives containing most of the \TeX related resources (like fonts, software, documentations, packages). They are accessible via Internet. See http://ctan.org.

dblaccent I needed to use the dblaccent [328] package to typeset “The pdf\TeX Program” entry [204] in the bibliography, because its author’s first name contains a double accent (Thé Thành Hân).

de.com.text.tex The Usenet news group about \TeX and \L\TeX, in german.

depth In the standard documents classes (and in most classes) with sectioning commands, we have a notion of depth. The depth of a sectioning command determines the numbering level in its title (from the value of the secnumdepth counter), and the entries for a given sectioning command appear in the main table of contents if the depth of this sectioning command is lower than or equal to the value of the tocdepth counter; see table 6.4 for the depths of the sectioning commands in the main document classes.

The mechanism is analog for the parttocs, minitocs, and secttoccs, using the parttocdepth, minitocdepth, and secttocdepth counters. If you use sub-figures or sub-tables, the corresponding mini-tables use counters like partlofdepth, partlotdepth, minilofdepth, minilotdepth, sectlofdepth, and sectlotdepth.

descriptor (file descriptor). A software entity describing the interface between a program and a file. For most programs and operating systems, the number of file descriptors is limited. For \TeX (and \L\TeX), there are 16 file descriptors for writing and 16 file descriptors for reading.
devanagari.sty  The Devanagart for \TeX{} (Devanāgarī) package [364] provides a way to typeset high-quality Devanāgarī text with \TeX{}. Devanāgarī is a script used for writing and printing Sanskrit and a number of languages in Northern and Central India such as Hindi and Marathi, as well as Nepali. The Devanāgarī package was originally developed in May 1991 by Frans Velthuis for the University of Groningen, The Netherlands, and it was the first system to provide support for the Devanāgarī script for \TeX{}.

em  A length unit equal (approximatively) to the width of a “m” letter in the current font.

emk  An example of shell script to prepare the english documentation of the minitoc package. The script \emk{} must have be run previously. See item 10 on page 245.

en-mtc.bst  A bibliographic style derived from the plain.bst standard style, but modified with the urlbst tool [196] to add an URL field. Family names of authors and editors are in small caps, years are in old style digits.

encoding  This specifies the order that characters appear in the font (e.g., whether the 65th character is “A”). The most common value for \TeX{} font encoding is OT1. The other predefined option is T1 (extended \TeX{}). There’s also US ASCII (7 bit), ISO Latin-1 (8 bit), Adobe Standard Encoding, UTF8 (Unicode [128, 151, 448]), etc. See table 6.5 on the following page and [292, 323].

environment  An environment is a delimited domain in a document, where special rules apply. Such a domain is delimited by \begin{env} \ldots \end{env} and may take arguments, like this:

\begin{minipage}[t]{.5\textwidth}
\end{minipage}

ε-\TeX{}  ε-\TeX{} [105] is an extented version of \TeX{}, with much more registers and many new primitives; it supports also left-to-right and right-to-left writing.

ethiop  A \UTFX{} package [44] giving the ethiopian language support for the babel package [60, 61].

extension  The name of a file is often made of 2 parts: a base name and an extension, separated by a dot. On some old operating systems, the base name is limited to 8 characters and the extension to 3 characters (the “8+3” scheme). See also sections 1.9 on page 54 and 2.5 on page 58. It is strongly recommended to not have more than one dot in a file name.

farsi.sty  See Farsi\TeX{} below.
## Table 6.5: Various encodings

<table>
<thead>
<tr>
<th>Encoding</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>ansinew</td>
<td>Windows 3.1 ANSI encoding, extension of Latin-1.</td>
</tr>
<tr>
<td>applemac</td>
<td>Macintosh encoding.</td>
</tr>
<tr>
<td>ascii</td>
<td>ASCII encoding for the range 32–127.</td>
</tr>
<tr>
<td>cp1250</td>
<td>Windows 1250 (Central and Eastern Europe) code page.</td>
</tr>
<tr>
<td>cp1251</td>
<td>Windows 1251 (Cyrillic) code page.</td>
</tr>
<tr>
<td>cp1252</td>
<td>Synonym for ansinew.</td>
</tr>
<tr>
<td>cp1257</td>
<td>Windows 1257 (Baltic) code page.</td>
</tr>
<tr>
<td>cp437</td>
<td>IBM 437 code page, which is the original American code page and contains letters, digits, mathematical symbols, and some characters useful in the construction of pseudographics.</td>
</tr>
<tr>
<td>cp437de</td>
<td>IBM 437 code page (German version).</td>
</tr>
<tr>
<td>cp850</td>
<td>IBM 850 code page, almost the same as ISO Latin 1, but character arrangement is not the same.</td>
</tr>
<tr>
<td>cp852</td>
<td>IBM 852 code page.</td>
</tr>
<tr>
<td>cp855</td>
<td>IBM 855 code page (Cyrillic).</td>
</tr>
<tr>
<td>cp865</td>
<td>IBM 865 code page.</td>
</tr>
<tr>
<td>cp866</td>
<td>IBM 866 code page (MS-DOS Cyrillic).</td>
</tr>
<tr>
<td>decmulti</td>
<td>DEC Multinational Character Set encoding.</td>
</tr>
<tr>
<td>latin1</td>
<td>ASCII encoding plus the characters needed for most Western European languages, including Danish, Dutch, English, Faroese, Finnish, Flemish, French, German, Icelandic, Italian, Norwegian, Portuguese, Spanish, and Swedish. Some non-European languages, such as Hawaiian and Indonesian, are also written in this character set.</td>
</tr>
<tr>
<td>latin2</td>
<td>ASCII encoding plus the characters needed for most Central European languages, including Croatian, Czech, Hungarian, Polish, Romanian, Slovak, and Slovenian.</td>
</tr>
<tr>
<td>latin3</td>
<td>ASCII encoding plus the characters needed for Esperanto, Maltese, Turkish, and Galicean. However, latin5 is the preferred character set for Turkish.</td>
</tr>
<tr>
<td>latin4</td>
<td>ASCII encoding plus the characters needed for the Baltic languages (Latvian, Estonian, and Lithuanian), Greenlandic, and Lappish (Sámi).</td>
</tr>
<tr>
<td>latin5</td>
<td>Is essentially the same as latin1, except that some Turkish characters replace less commonly used Icelandic letters.</td>
</tr>
<tr>
<td>next</td>
<td>Next encoding.</td>
</tr>
</tbody>
</table>

### Farsi\TeX
A package [162] to typeset a document in the \textsc{farsi} (iranian, persian) language. See \url{http://www.farsitex.org}. But this package is today available only for \LaTeX\,2.09. See also sections 13.54 on page 494 and 13.55 on page 495.

### features
A feature (for the \texttt{minitoc} package) is a set of actions executed at each occurrence of a mini-table of a given type. Five features are associated to each mini-table type: a “before” feature (executed before the whole mini-table), an “after” feature (executed after the whole mini-table), a “open” feature, executed just before inserting the mini-table file, a “close” feature, executed just after inserting the mini-table file, and a “pagestyle” feature, which is executed with the mini-table to set its page style. Look at the documentation of the \texttt{\mtcsetfeature} command, in section 1.10 on page 48.
**filecontents** A special \LaTeX{} environment. It allows to create a file (whose name is passed as an argument of the environment) by writing the contents of the environment into that file:

```
\begin{filecontents}{file}
...contents ...
\end{filecontents}
```

This environment should be used before `\documentclass`. It is used in `minitoc.ins` to prepare the `.mlo` files (see section 1.4.14 on page 44) and some files used in the compilation of the documentation.

`\firstchapter` An obsolete command, temporarily used as a workaround for the Chapter 0 problem; see Chapter 0 and absolute numbering above, and section 1.5.4 on page 49.

`\firstpart` Analog to `\firstchapter` above.

`\firstsection` Analog to `\firstchapter` above.

**float** This package \cite{float} (by Anselm Lingnau) improves the interface for defining floating objects such as figures and tables in \LaTeX{}. It adds the notion of a ‘float style’ that governs appearance of floats. New kinds of floats may be defined using a `\newfloat` command analogous to `\new theorem`. This package also incorporates the functionality of David P. Carlisle’s package \cite{here}, giving floating environments a [H] option which means ‘PUT IT HERE’ (as opposed to the standard [h] option which means ‘You may put it here if you like’).

`\FloatBarrier` A macro from the `placeins` package \cite{placeins}. It sets up a “barrier” against the drift of floats (like figures or tables).

**floatrow** This package \cite{floatrow} (by Olga G. Lapko) is an extension of the `float` package \cite{float} (by Anselm Lingnau), reusing its code, with extensions from the `rotfloat` package \cite{rotfloat} (by Axel Sommerfeldt).

**flowfram** This package \cite{flowfram} is designed to enable you to create text frames in a document such that the contents of the document environment flow from one frame to the next in the order that they were defined. This is useful for creating posters or magazines or any other form of document that does not conform to the standard one or two column layout. As this package defines its own system of minitocs, it is incompatible with the `minitoc` package.

**fmk** An example of shell script to prepare the french documentation of the `minitoc` package. The script \texttt{fmk} must be run previously. See item 10 on page 245.

**fncychap** The `fncychap` package \cite{fncychap} provides a set of commands for changing the format used for some headings (chapters) in the standard \LaTeX{} document classes: `book` and `report`. It must be loaded before the `minitoc` package (see point 1.8 on page 53 and section 2.38 on page 75).

**fr.comp.text.tex** The Usenet newsgroup about \TeX{} and \LaTeX{}, in french.

**franc.sty** A small package file used to prepare the french documentation. It is generated when compiling `minitoc.ins`. 
frbib.sty  A small package file used to prepare the bibliography of the French documentation. It is generated when compiling minitoc.ins.

fr-mtc.bst  A bibliographic style file used to prepare the bibliography of the French documentation. It has been updated from the standard plain.bst for French by Ronan Keryell, then I added some adaptations for French (like last names in small caps for authors and editors, years in old style digits), then modified with the urlbst [196] tool to add an URL field.

frnew.sty  A small package file used to prepare the French documentation. It is generated when compiling minitoc.ins.

gometry  The geometry package [447] provides a flexible and complete user interface to page dimensions. You can specify them by using intuitive parameters to get your desired page layout. For instance, if you want to set margins (the left, right, top and bottom margins) to 2cm from each edge of the paper, what you need is just:

\usepackage[margin=2cm]{geometry}

This powerful (and recommended) package is used in some example documents and in this documentation.

guarani  A LATEX package to compose text in Guaraní, the main language spoken in Paraguay. The file guarani.ldf, included in this package, defines the titles. See [45] and section 13.74 on page 504.

H

hangcaption  The hangcaption package [250] defines a variant of the \caption command to produce captions with hanging indentation. This package is likely obsolete (1992, LATEX2.09).

Hindi  For the Hindi language, see the Devanāgarī package [364] above. The minitoc package accepts the devanagari and hindi language options, which are synonyms. A hindi-modern language option is also available. See also [148] about the Hindi language.

hint  An indication, a clue to detect a problem. It is also a message written (into the document.log file) by the hints option (see below).

hints  An option of the minitoc package. It verifies the loading order of some packages, the invocation order of some minitoc commands, the consistency between main minitoc commands, etc., and gives warnings and other useful hints (mainly in the document.log file). This is a default option (use the nohints option to skip these checks).

**Hl**\TeX

A system to write documents in the Korean language, using *Lambda* (Λ) (see below). Written by Un Koangui [266, in korean]. It uses special Hangûl or Hanja fonts and the UTF-8 input encoding.

**hyperlink** In a document, a reference to another object which is dynamically found (via a click with the mouse). This requires a special type of document (PDF, PostScript with hypertext features) and a suitable viewer (PDF viewer, recent PS viewer). This is useful to navigate in a document or in many documents, which can be remote documents.

**hyperref** The hyperref package [390] is used to emend cross-referencing commands in \LaTeXX to produce some sort of \special commands; there are backends for the \special set defined for Hyper\TeX\ dvix processors, for embedded pdfmark commands for processing by Acrobat Distiller (dvips and dvipsone), for dviwindo, for pdf\LaTeX, for \TeX\4ht, and for VTE\TeX's pdf and HTML backends.

This package derives from, and builds on, the work of the Hyper\TeX project, described in [371]. It extends the functionality of all the \LaTeXX cross-referencing commands (including the table of contents, bibliographies, etc.) to produce \special commands which a driver can turn into hypertext links; it also provides new commands to allow the user to write *ad hoc* hypertext links (hyperlinks), including those to external documents and URLs.

**ifmtarg** The ifmtarg package [483] provides an if-then-else command for testing if a macro argument is empty ("empty" meaning zero or more spaces only).

**ifthen** The ifthen package [118] implements an \ifthenelse command for \LaTeXX_2ε.

**imk** An example of shell script, which prepares the minitoc package from minitoc.ins and minitoc.dtx; note that imk must be run before running emk or fmk. See item 10 on page 245.

**insection** The insection package option loads the placeins package [15] with adequate options to avoid the floats (like figures and tables) to drift outside of their sections. This package option is recommended if you use sectlofs or sectlots in your document. See section 1.2 on page 30.

**insertion** The insertion commands of the minitoc package insert a mini-table in the document. A corresponding preparation command must have been invoked (only once) before. The insertion commands are (see table 3.9 on page 87):

\begin{verbatim}
\parttoc, \partlof, \partlot,
\minitoc, \minilof, \minilot,
\secttoc, \sectlof, \sectlot,
\mtcprepare
\end{verbatim}

**INSTALL** A text file describing the installation of the minitoc package. See chapter 7 on page 242.
\jobname  A \TeX{} primitive containing the name of the document in preparation, i.e., the name of the file read first by \TeX{} (or \LaTeX{}), without its extension. Very useful to build the names of other files.

\textbf{jura} The jura class \cite{103}, by Felix \textsc{Braun}, is \textit{incompatible} with the \texttt{minitoc} package. It is used for german judicial documents.

\textbf{Kannada\TeX{}} A project \cite{485} to use \LaTeX{} for typesetting in the Kannada language. See section 13.101 on page 517.

\textbf{k-loose} A \texttt{minitoc} package option useful if your document is written with one of the KOMA-Script classes \cite{343, 344, 399}. This option tries to set a loose line spacing in the mini-tables. Analog to the \texttt{loose} package option for standard classes.

\textbf{k-tight} A \texttt{minitoc} package option useful if your document is written with one of the KOMA-Script classes \cite{343, 344, 399}. This option tries to set a tight line spacing in the mini-tables. Analog to the \texttt{tight} package option for standard classes.

\textbf{KOMA-Script} KOMA-Script \cite{343, 344, 399} is a very complex bundle. You may see this, because it is not only one class or one package but a bundle of many classes and packages. The classes (\texttt{scrartcl}, \texttt{scrbook}, \texttt{scrlttr}, \texttt{scrlttr2}, and \texttt{scrreprt}) are counterparts to the standard classes but never they come with only the same commands, environments, options and optional possibilities like the standard classes nor they result in the same look-a-like.

The \texttt{scrbook}, \texttt{scrreprt}, and \texttt{scrlttr} classes are compatible with the \texttt{minitoc} package, with some precautions (see section 1.5.5 on page 50). The \texttt{scrlttr} and \texttt{scrlttr2} have no sectionning commands, so the \texttt{minitoc} package is pointless with them.

KOMA-Script comes with a lot of classes, packages, commands, environments and possibilities. Some of these you may find also at the standard classes, many of them you wouldn't. Some are even supplements to the \LaTeX{} kernel.

The main classes of the KOMA-Script bundle are designed as counterparts to the standard \LaTeX{} classes. This means that the KOMA-Script bundle contains replacements for the three standard classes \texttt{book}, \texttt{report}, and \texttt{article}. There is also a replacement for the standard class \texttt{letter}.

\textbf{Lambda} The \LaTeX{} format (in the \TeX{} meaning of that word) adapted to the special features of Omega (\textit{Ω}) is called “Lambda” (\textit{Λ}).
Lamed  The \LaTeX{} format (in the \TeX{} meaning of that word) adapted to the special features of \aleph{} (\( \aleph{} \)) is called “Lamed” (\( \mathbb{L} \)).

\LaTeX{}  \LaTeX{} [279] is a typesetting system that is very suitable for producing scientific and mathematical documents of high typographical quality. It is also suitable for producing all sorts of other documents, from simple letters to complete books. \LaTeX{} uses \TeX{} [263, 265] as its formatting engine (from [356]).

In fact, \LaTeX{} is a macro package that enables authors to typeset and print their work at the highest typographical quality, using a predefined, professional layout. \LaTeX{} was originally written by Leslie Lamport [279]. It uses the \TeX{} formatter as its typesetting engine. These days \LaTeX{} is maintained by Frank Mittelbach and his team.

In 1994 the \LaTeX{} package was (deeply) updated by the \LaTeX{}3 team, led by Frank Mittelbach, to include some long-requested improvements, and to reunify all the patched versions which had cropped up since the release of \LaTeX{}2.09 some years earlier. To distinguish the new version from the old, it is called \LaTeX{}2ε.

\LaTeX{} is pronounced “Lay-tech” or “Lah-tech.” If you refer to \LaTeX{} in an ASCII environment, you type \TeX{}. \LaTeX{}2ε is pronounced “Lay-tech two ε” and typed LaTeX2ε.

\LaTeX{}2.09  An obsolete version of the \LaTeX{} program, before 1994; it is no more supported. Do not use it\(^5\). Use the current version of \LaTeX{}2ε, which is supported and much more efficient.

\LaTeX{}2ε  The current version of the \LaTeX{} program, after 1994; it is supported.

\LaTeX{}3  The future version of \LaTeX{}, whose development is still in progress.

leaders  A repetitive sequence of dots (or of one another small character), regularly spaced, used to link two objects on the same line (leading from a title to a page number in a table of contents or the like).

letter  A standard document class [283] to prepare letters for postal mail (mail on paper). As such documents have no sectioning commands nor structure, the minitoc package is pointless (hence incompatible) with them.

lipsum  The \texttt{lipsum} package [212] allows to easily insert sentences in a test file with a minimum of typing. The sentences are in latin but are modified and made nearly senseless. I have used this package in some of the examples of documents. See also \texttt{http://lipsum.com} for the origin of this text (pieces of \textit{De Finibus Bonorum et Malorum} by Marcus Tullius Cicero).

listfiles  An option of the minitoc package. It creates a list of the minitoc auxiliary files (these files contains the mini-tables and may be removed after the \LaTeX{} run) in the \texttt{document.maf} file. Default. See section 1.7 on page 52.

\textit{LOF, LoF}  An acronym for “list of figures”.

lofdepth  This counter, if it exists, contains the depth of the list of figures.

\(^5\) Except in the case of a very old document; if possible, try to convert it.
loose  An option of the minitoc package. It gives a loose line spacing in the mini-tables. It is the default. The opposite option is tight.

LOT, LoT  An acronym for “list of tables”.

lotdepth  This counter, if it exists, contains the depth of the list of tables.

LPPL  The LaTeX Project Public License, available at

http://www.latex-project.org/lppl.txt

Its current version is 1.3 (2003-12-01). The minitoc package is distributed under this license.

ltxdoc  A standard \LaTeX{} document class [116], for preparing the documentation of a package or of a class. For the minitoc package, it is very similar to the article document class; see above.

ltnews  A standard \LaTeX{} document class [248], for preparing the announcement of a \LaTeX{} release. For the minitoc package, it is very similar to the article document class; see above.

\makeatletter and \makeatother  Many internal commands of \LaTeX{}, of packages and classes contain the @ character in their names. This effectively prevents such names from being used in documents for user-defined commands. However, it also means that they cannot appear in a document, even in the preamble, without taking special precautions. As it is sometimes necessary to have such bits of “internal code” in the preamble, the commands \makeatletter and \makeatother make it easy to do: the difficult bit is to remember to add them, failure to do so can result in some strange errors. And these two commands should never be used in a package or class file.

makefile  A special text file containing instructions describing the creation and the installation of a piece of software, using the “make” utility; make is a nice tool coming from the Unix operating system, but variants exists.

mcaption  The mcaption package [228] provides a margincap environment for putting captions in the outer document margin with either a top or bottom alignment.

MCE  A minimal [complete] example is the smallest possible complete document that illustrates a problem. A minimal example file should not include any packages or code that do not contribute to the problem, but must include a document class and the document environment (from [432]). See also [384], http://www.tex.ac.uk/cgi-bin/texfaq2html?label=minxampl and http://www.tex.ac.uk/cgi-bin/texfaq2html?label=askquestion for good advices.

\footnote{Informations from [330, page 843].}
**memoir, memoir.cls** A very general and powerful document class (by Peter R. Wilson, described in [479, 481, 482]); this class is compatible with the minitoc package (with some precautions) if you use a recent version. See section 2.22 on page 65.

**mini-bibliography** See minibbl below.

**mini-list** Synonym for “mini-table” below.

**mini-lof** See “minilof” below.

**mini-lot** See “minilot” below.

**mini-table** This term refers to a local table of contents (like a table of contents, a list of figures or a list of tables) for a sectioning unit (part, chapter or section), by opposition to a global table (the table of contents, the list of figures or the list of tables for the whole document). The main aim of the minitoc package is the creation of such mini-tables. But the term “minitoc” is also used to refer to such mini-table, as a generic term, because the first versions of the package allowed only tables of contents for chapters.

**mini-toc** See “minitoc” below.

**minibbl** Short for “mini-bibliography”, i.e., to have a bibliography per part, chapter or section, or even by theme or subject. This is out of the domain of the minitoc package. See section 2.9 on page 60.

**minilof** A list of figures for a chapter.

**minilofdepth** This counter, if defined, contains the depth of the minilofs.

**minilot** A list of tables for a chapter.

**minilotdepth** This counter, if defined, contains the depth of the minilot.

**minitoc** A table of contents for a chapter. Also used as a generic term for any mini-table (see “mini-table” above).

**minitoc-fr.bib** A bibliographic data base for the french documentation of the minitoc package.

**minitoc-fr.dtx** The source file for the french documentation of the minitoc package. In fact, it just sets \jobname then loads minitoc.dtx, which itself loads \jobname.lan to select the language used in minitoc.dtx; minitoc.dtx contains both english and french documentation fragments, selected by \ifcase constructs with the \LANG variable, set to 0 by minitoc.lan or to 1 by minitoc-fr.lan (i.e., by \jobname.lan). minitoc-fr.dtx is generated when compiling minitoc.ins.

**minitoc-fr.ist** This file contains a style for formatting the index in the french documentation. It is generated when compiling minitoc.ins.

**minitoc-fr.lan** A file used to force the french language in the documentation. It is generated when compiling minitoc.ins.

**minitoc-fr.pdf** The french documentation in PDF format.
**minitoc-fr.ps** The french documentation in PostScript format. No more distributed (but look at the cmk script).

**minitoc.bib** A bibliographic data base for the english documentation of the minitoc package.

**minitoc.bug** A plain text file containing a list of problems and questions about the minitoc package. See chapter 2 on page 56.

**minitocdepth** This counter contains the depth of the minitoc.

**minitoc.dtx** The file containing the documentation and the commented code of the minitoc package.

**minitoc-hyper.sty** A special version [454] of the minitoc package which has been prepared by Bernd Jaeck, Didier Verna and A. J. “Tony” Roberts to work with the powerful hyperref package [390]. Heiko Oberdiek has integrated their work so since version #31, minitoc is compatible with hyperref. *Hence the minitoc-hyper package [454] is now obsolete and should no more be used. It it still present on the CTAN archives for compatibility with old documents.*

**minitoc.ins** The installation file for the minitoc package. Compiling it with \LaTeX produces most of the files of the minitoc package.

**minitoc.ist** This file contains a style for formatting the index in the english documentation. It is generated when compiling minitoc.ins.

**minitoc.l** A text file containing the list of all the files being included in the minitoc package. Files not listed in minitoc.l are files used only to install the package or to produce its documentation.

**minitoc.lan** A file used to force the english language in the documentation. It is generated when compiling minitoc.ins.

**minitoc.pdf** The english documentation in PDF format.

**minitoc.pre** This file contains a \LaTeX preamble for the documentation. It is generated when compiling minitoc.ins.

**minitoc.ps** The english documentation in PostScript format. No more distributed (but look at the cmk scripts).

**minitoc.sty** This file contains the main part of the minitoc package, with comments removed. It is generated when compiling minitoc.ins.

**minitoc.sum** A plain text file containing a commented list of the minitoc commands and environments. See chapter 3 on page 80.

**minitoc.tds.zip** A ZIP archive of a TDS-compliant hierarchy containing all files in the minitoc package.

**minutes** The minutes package [300] (by Knut Licke) is used to prepare conference proceedings. The minitoc package allows to add “coffee breaks” in the table of contents via commands like \addcoffeeline and \coffeeline (and internal commands) whose names contain the string “coffee”, hence the footnote about “cappuccino” in the installation chapter!  

---

7 The little cups ☕ come from the marvosym package [227].
MonTeX is a large package to prepare documents in various dialects of the Mongol language (Bicig and Bicig2, Mongol, Bithe and Manju, Buryat, Xalx and Khalkha) and in a dialect of Russian used in Mongolia (Russian). Bicig is another name for Uighur. You can find many things about Mongolia and Mongolian at the web site [139]. See also:

- http://en.wikipedia.org/wiki/Mongolian_language
- http://en.wikipedia.org/wiki/Mongolian_script
- http://www.indiana.edu/~mongsoc/mong/language.htm

The following description is extracted from [140].

MonTeX is a package which offers support for writing documents in Mongolian, Manju, Buryat and Russian.

Mongolian can be represented in traditional Uighur script (also known as Classical or Traditional Script) and Cyrillic. Manju resembles the Traditional Mongolian script (from which it is derived) but uses a rich choice of diacritics in order to eliminate numerous ambiguities of the Mongolian script ancestor. Modern Buryat, like Mongolian in its present form, is written with a Cyrillic alphabet, but both Mongolian (35 letters) and Buryat (36 letters) use more letters than Russian (33 letters).

**Mongolian**

The word *Mongolian* is actually an umbrella term for several languages rather than the precise name of a single language. Things become more complicated when names of ethnic groups, languages and writing systems are mixed.

**Xalx** or Khalkha is the name of the Mongolian nationality residing in Mongolia proper. Their dialect forms the basis of Mongolian written with Cyrillic letters. Throughout this text, *Modern Mongolian* is used as a synonym.

**Buryat** is the name of the Mongolian nationality residing in Buryatia, north of Mongolia, east of Lake Baikal, being a part of the Russian Federation. The Buryat call themselves *Buryad* while Xalx Mongolians call them *Buriad*. The English name follows the Russian orthography. Linguistically, Xalx and Buryat Mongol are fairly close languages; Buryat has a slightly different sound system in which the phoneme /s/ partially shifted to /h/; the modern Buryat Cyrillic alphabet (virtually identical with the Cyrillic alphabet used for writing Modern Mongolian) has one additional letter (H/h, \xa1x{H/h}) for marking the difference to /s/.

**Bicig** (literally *script* in Mongolian) denotes text written in the traditional Mongolian script which is also referred to as Uighur. Throughout this document, the term *Bicig* will be used on an equal footing with *Classical* and *Traditional* Mongolian. The latter term is used in the names of the Unicode/ISO10646 character plane U1800 which contains Mongolian, Manju, Sibe and sets of special characters called Ali Gali or Galig. In order to identify Mongolian script related commands distinct for Mon-
golian and Manju, the Mongolian commands have the name root bicig whereas the Manju commands have the name root bi the.

Xalx Mongolian, or Modern Colloquial Mongolian, is about as different from the form written in Classical script as modern English in phonetical spelling (assume it be written in Shavian letters) from the highly historical orthography of Standard English. Beyond these differences, Mongolian written in Classical Script usually preserves a substantial amount of historical grammatical features which make it look a bit like Elizabethan English.

Manju Manju is a Tungusic language closely related to Mongolian. Though Manju is virtually not spoken anymore, it has been the official language during 300 years of Manju government in Qing Dynasty China. Vast amounts of official documents survive, as well as some of the finest multilingual dictionaries ever compiled, e.g. the Pentaglot, or Mirror in Five Languages, a dictionary with 18671 entries in five languages (Manju, Tibetan, Mongolian, Uighur and Chinese). See [138] for more details. Manju writing is derived from Uighur Mongolian by adding diacritics in the form of dots and circles (tongki fuka sindaha hergen, script with dots and circles).

MS-DOS (Microsoft® Disk Operating System) An old operating system for personal computers (PCs). From the minitoc point of view, its main drawback is the use of filenames with short extensions (the “8+3” scheme), which limits to 99 the number of mini-tables for each kind.

mtc-2c.tex An example file showing the use of the minitoc package with a two columns page layout. See section 4.1 on page 91.

mtc-2nd.tex An example of document using the minitoc package and its french2 language option. See section 4.2 on page 92.

mtc-add.bib A small bibliographic data base for the mtc-add.tex and mtc-adds.tex example documents. See section 4.4 on page 96.

mtc-3co.tex An example of document using the minitoc package to prepare a minitoc on three columns. See section 4.3 on page 93.

mtc-add.tex An example document showing how to use \mtcaddchapter and the tocbibind package [472] with minitoc. See section 4.4 on page 96.

mtc-adds.tex An example document showing how to use \mtcaddsection and the tocbibind package [472] with minitoc. See section 4.5 on page 100. It also shows how it is challenging to manage the mini-lists of floats at the section level.

mtc-amm.tex An example file showing the use of the appendices environment in a memoir class document with the minitoc package. See section 4.6 on page 105.

mtc-apx.tex An example file showing the use of the mtchideinmaintoc environment to hide the entries of the appendices in the main TOC and to create a part-level TOC for the appendices. See section 2.25 on page 67.

mtc-art.tex An example of document (article class) using the minitoc package. See section 4.8 on page 105.
mtc-bk.tex  An example of document (book or report class) using the minitoc package. See section 4.9 on page 110.

mtc-bo.tex  An example file showing the use of the minitoc package with a two columns page layout and using the tocloft package [469]. See section 4.10 on page 115.

mtc-ch0.tex  An example file showing the use of the minitoc package in a document with a starred first chapter. See section 4.11 on page 119.

mtc-cri.tex  An example file showing the use of the minitoc package with starred parts and chapters. See section 4.12 on page 121.

mtc-fko.tex  An example file showing the problem of fonts in minitocs when using the scrbook class. See section 4.13 on page 121.

mtc-fo1.tex  An example file showing the use of the minitoc package with changing some fonts. See section 4.14 on page 122.

mtc-fo2.tex  Another example file showing the use of the minitoc package with changing some fonts. See section 4.15 on page 123.

mtc-gap.tex  An example file showing the use of the \mtcgapbeforeheads and \mtcgapafterheads commands. See section 2.44 on page 76.

mtc-hi1.tex  An example file showing the use of the mtchideinmainlof and mtchideinmainlot specialized environments. See section 2.25 on page 68.

mtc-hi2.tex  An example file showing the use of the following pairs of commands:

- \mtchideinmainlof and \endmtchideinmainlof,
- \mtchideinmainlot and \endmtchideinmainlot.

See section 2.25 on page 69.

mtc-hia.tex  An example file showing the use of the minitoc package to hide the entries for some tables in the main list of tables of an article class document. See section 4.19 on page 125.

mtc-hir.tex  An example file showing the use of the minitoc package to hide the entries for some tables in the main list of tables of a report class document. See section 4.20 on page 126.

mtc-hop.tex  An example file showing the use of the minitoc package with the scrbook document class. See section 4.21 on page 127.

mtc-liv.tex  An example file showing the use of the minitoc package in a book with customized table of contents and minitocs. See section 4.22 on page 128.

mtc-mem.tex  An example file showing the use of the minitoc package with the memoir class. See section 4.23 on page 132.

mtc-mml.tex  An example file showing the use of the minitoc package with the memoir class, if you want to change some fonts. See section 4.24 on page 133.

mtc-mu.tex  A document using a minitoc set in a wrapfigure environment with the wrapfig package [18]. See section 4.25 on page 134.
**mtc-nom.tex** A document showing an interaction between the `minitoc` package and the `nomencl` package [456]. See section 4.26 on page 136.

**mtc-ocf.tex** A document using the `open` and `close` features to prepare a `minitoc` on three columns. See section 4.27 on page 137.

**mtc-ofs.tex** A document using the `open` and `close` features to prepare a `minitoc` on three columns and `\mtcsetoffset` to shift the `minitoc` to align it on the left. See section 4.28 on page 138.

**mtc-sbf.tex** An example file showing the use of the `minitoc` package with the `subfigure` package [130]. See section 4.29 on page 140.

**mtc-scr.tex** An example file showing the use of the `minitoc` package with a `KOMA-Script` class [343, 344, 399], `scrreprt`. See section 4.30 on page 141.

**mtc-syn.tex** An example file showing the use of the `minitoc` package when the table of contents is preceeded by some starred chapters. See section 4.31 on page 143.

**mtc-tbi.tex** An example file showing the use of the `minitoc` package with the `tocbibind` package [472]. See section 4.32 on page 144.

**mtc-tlc.tex** An example file showing the use of the `minitoc` package in a document of `article` class. It is the example of [330, page 58], modernized. See section 4.33 on page 145.

**mtc-tlo.tex** An example file showing the use of the `minitoc` package with the `tocloft` package [469] and their interaction about the page numbers in the mini-tables. See section 2.46 on page 79.

**mtc-tsf.tex** An example file showing the use of the `minitoc` package with the `subfig` package [132]. See section 4.35 on page 146.

**mtc-vti.tex** An example file showing the use of the `\mtcpolymtoc` command and explaining “polymorphic entries”. See section 4.36 on page 148.

**mtcmess** A package used to provide variants of the standard commands `\PackageInfo`, `\PackageWarning`, `\PackageWarningNoLine`, and `\PackageError` by adding an optional argument for an unique message identifier.

**mtcoff** A package which is used in place of the `minitoc` package to ignore all the commands and environments of the `minitoc` package. In fact, it defines them to do nothing. Useful if you want a version of your document without any mini-table.

**mtcpatchmem** A small package which is automatically loaded if necessary when you use the `memoir` document class with a version *incompatible* with the `minitoc` package, but correctible. It is generated when compiling `minitoc.ins`.

**mu** A length unit normally used in math mode (`\mu` means “math unit”); 18 math units make 1em (one quad), which is about the width of a “m” in the current font. So the size of `1\mu` is font dependent. The separation between dots in the dotted lines in the mini-tables is expressed in math units.
**multibib** The multibib package [211] allows to create references to multiple bibliographies within one document. It thus provides a complementary functionality to packages like bibliography [210] or chapterbib [19], which allow to create one bibliography for multiple, but different parts of the document.

**multicol** The multicol package [325] defines the multicols environment (with a “s”) to typeset text on several columns. Used in some example documents.

**multitoc** This package [414] allows setting only the table of contents, list of figures and/or list of tables in two or more columns (using the multicol package [325], of course). The number of columns can be configured via commands; the multicolumn toc(s) can be selected via package options. The mtc-3co.tex example document uses this package; see section 4.3 on page 93.

**natbib** It is a \LaTeX{}2ε (but with some support for \LaTeX{}2.09) package [145, 146] to act as a generalized interface for standard and non-standard bibliographic style files (\BibTeX{}).

**needspace** The needspace package [468] provides commands to reserve space at the bottom of a page. If there is not enough space on the current page (column) a new page (column) is started.

**NFSS** The New Font Selection Scheme. The \LaTeX{}2ε font selection system [291] was first released as the “New Font Selection Scheme” (NFSS) in 1989, and then in release 2 in 1993. \LaTeX{}2ε includes NFSS release 2 as standard.

Every text font in \LaTeX{} has five attributes:

- **encoding** This specifies the order that characters appear in the font. The two most common text encodings used in \LaTeX{} are Knuth’s “\TeX{} text” encoding (OT1), and the “\TeX{} text extended” encoding (T1) developed by the \TeX{} Users Group members during a \TeX{} Conference at Cork in 1990 (hence its informal name “Cork encoding”). See [292, 323].

- **family** The name for a collection of fonts, usually grouped under a common name by the font foundry. For instance, “Adobe Times”, “ITC Garamond”, and Knuth’s “Computer Modern Roman” are all font families.

- **series** How heavy or expanded a font is. For instance, “medium weight”, “narrow” and “bold extended” are all series.

- **shape** The form of the letters within a font family. For instance, “italic”, “oblique” and “upright” (sometimes called “roman”) are all font shapes.

- **size** The design size of the font, for instance “10pt”.

The possible values for these attributes are given short acronyms by \LaTeX{}. The most common values for the font encoding are given in table 6.6 on the next page.

The “local” encodings are intended for font encodings which are only locally available, for instance a font containing an organisation’s logo in various sizes.
Table 6.6: Most common font encodings

<table>
<thead>
<tr>
<th>Encoding</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>T1</td>
<td>\LaTeX\ extended text (“Cork encoding”)</td>
</tr>
<tr>
<td>TS1</td>
<td>\LaTeX\ symbols (Latin)</td>
</tr>
<tr>
<td>T2A, T2B, T2C</td>
<td>\LaTeX\ text (Cyrillic)</td>
</tr>
<tr>
<td>T3</td>
<td>\LaTeX\ phonetic alphabet</td>
</tr>
<tr>
<td>TS3</td>
<td>\LaTeX\ phonetic alphabet (extra symbols)</td>
</tr>
<tr>
<td>T4</td>
<td>\LaTeX\ text (African languages)</td>
</tr>
<tr>
<td>T5</td>
<td>\LaTeX\ text (Vietnamese)</td>
</tr>
<tr>
<td>T7</td>
<td>\LaTeX\ text (reserved for Greek)</td>
</tr>
<tr>
<td>OT1</td>
<td>\TeX\ text (as defined by Donald E. Knuth)</td>
</tr>
<tr>
<td>OT2</td>
<td>\TeX\ text for Cyrillic languages (obsolete)</td>
</tr>
<tr>
<td>OT3</td>
<td>International phonetic alphabet (obsolete)</td>
</tr>
<tr>
<td>OT4</td>
<td>\TeX\ text with extensions for the Polish language</td>
</tr>
<tr>
<td>OT6</td>
<td>\TeX\ text with extensions for the Armenian language</td>
</tr>
<tr>
<td>OML</td>
<td>\TeX\ math italic (Donald E. Knuth)</td>
</tr>
<tr>
<td>OMS</td>
<td>\TeX\ math symbols (Donald E. Knuth)</td>
</tr>
<tr>
<td>OMX</td>
<td>\TeX\ math large symbols (Donald E. Knuth)</td>
</tr>
<tr>
<td>X2</td>
<td>\LaTeX\ extended text (Cyrillic)</td>
</tr>
<tr>
<td>U</td>
<td>Unknown</td>
</tr>
<tr>
<td>L⟨xx⟩</td>
<td>A local encoding</td>
</tr>
<tr>
<td>L7x</td>
<td>Encoding used for the Lithuanian language</td>
</tr>
<tr>
<td>LTH</td>
<td>Encoding used for the Thai language</td>
</tr>
<tr>
<td>LV1</td>
<td>Encoding used with some \TeX\ fonts</td>
</tr>
<tr>
<td>LY1</td>
<td>Alternative to T1 encoding, for Y&amp;Y software</td>
</tr>
<tr>
<td>PD1</td>
<td>Implements the PDFDocEncoding for use with \LaTeX\ 2e’s NFSS.</td>
</tr>
<tr>
<td>PU</td>
<td>Implements the Unicode encoding for use with \LaTeX\’s NFSS.</td>
</tr>
</tbody>
</table>

Table 6.7: Most common font families

<table>
<thead>
<tr>
<th>Family</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmr</td>
<td>Computer Modern Roman</td>
</tr>
<tr>
<td>cmss</td>
<td>Computer Modern Sans</td>
</tr>
<tr>
<td>cmtt</td>
<td>Computer Modern Typewriter</td>
</tr>
<tr>
<td>cmm</td>
<td>Computer Modern MathItalic</td>
</tr>
<tr>
<td>cmy</td>
<td>Computer Modern Math Symbols</td>
</tr>
<tr>
<td>cmex</td>
<td>Computer Modern Math Extensions</td>
</tr>
<tr>
<td>ptm</td>
<td>Adobe Times</td>
</tr>
<tr>
<td>phv</td>
<td>Adobe Helvetica</td>
</tr>
<tr>
<td>pcr</td>
<td>Adobe Courier</td>
</tr>
<tr>
<td>lazy</td>
<td>Additional \LaTeX\ symbols</td>
</tr>
</tbody>
</table>

Table 6.8: Most common font series

<table>
<thead>
<tr>
<th>Series</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>u1</td>
<td>Ultra light</td>
</tr>
<tr>
<td>e1</td>
<td>Extra light</td>
</tr>
<tr>
<td>1</td>
<td>Light</td>
</tr>
<tr>
<td>s1</td>
<td>Semi light</td>
</tr>
<tr>
<td>m</td>
<td>Medium</td>
</tr>
<tr>
<td>sb</td>
<td>Semi bold</td>
</tr>
<tr>
<td>b</td>
<td>Bold</td>
</tr>
<tr>
<td>eb</td>
<td>Extra bold</td>
</tr>
<tr>
<td>bx</td>
<td>Bold extended</td>
</tr>
<tr>
<td>ub</td>
<td>Ultra bold</td>
</tr>
<tr>
<td>c</td>
<td>Condensed</td>
</tr>
</tbody>
</table>
There are far too many font families to list them all, but some common ones are listed in table 6.7 on the preceding page. The most common values for the font series are listed in table 6.8 on the page before. The most common values for the font shape are listed in table 6.9. The most common values for the font width are listed in table 6.10.

The font size is specified as a dimension, for instance \texttt{10pt} or \texttt{1.5in} or \texttt{3mm}; if no unit is specified, \texttt{pt} is assumed. These five parameters specify every \LaTeX{} font, see table 6.11, for instance.

These five parameters are displayed whenever \LaTeX{} gives an overfull box warning, for instance:

\begin{verbatim}
Overfull \hbox (3.80855pt too wide) in paragraph at lines 314--318

[]\OT1/cmr/m/n/10 Normally [] and [] will be iden-ti-cal,
\end{verbatim}

The table 6.12 on the next page lists the author commands for fonts which set these five attributes\footnote{The values used by these commands are determined by the document class.}.

nocheckfiles A package option of \texttt{minitoc}. The opposite of the \texttt{checkfiles} package option (see above).

nohints A package option of \texttt{minitoc}. The opposite of the \texttt{hints} package option (see above).

nolistfiles An option of the \texttt{minitoc} package. It is the opposite of the \texttt{listfiles} above.  

See section 1.7 on page 52.
Table 6.12: Author commands for fonts

<table>
<thead>
<tr>
<th>Author command</th>
<th>Attribute</th>
<th>Value in article class</th>
</tr>
</thead>
<tbody>
<tr>
<td>\textemdash ... or \rmfamily</td>
<td>family</td>
<td>cmr</td>
</tr>
<tr>
<td>\textsf ... or \sffamily</td>
<td>family</td>
<td>cmss</td>
</tr>
<tr>
<td>\texttt ... or \ttfamily</td>
<td>family</td>
<td>cmtt</td>
</tr>
<tr>
<td>\textmd ... or \mdseries</td>
<td>series</td>
<td>m</td>
</tr>
<tr>
<td>\textbf ... or \bfseries</td>
<td>series</td>
<td>bx</td>
</tr>
<tr>
<td>\textit ... or \itshape</td>
<td>shape</td>
<td>n</td>
</tr>
<tr>
<td>\textsc ... or \scshape</td>
<td>shape</td>
<td>sc</td>
</tr>
<tr>
<td>\textup ... or \upshape</td>
<td>shape</td>
<td>n</td>
</tr>
<tr>
<td>\scriptsize</td>
<td>size</td>
<td>5pt</td>
</tr>
<tr>
<td>\footnotesize</td>
<td>size</td>
<td>8pt</td>
</tr>
<tr>
<td>\small</td>
<td>size</td>
<td>9pt</td>
</tr>
<tr>
<td>\normalsize</td>
<td>size</td>
<td>10pt</td>
</tr>
<tr>
<td>\large</td>
<td>size</td>
<td>12pt</td>
</tr>
<tr>
<td>\Large</td>
<td>size</td>
<td>14.4pt</td>
</tr>
<tr>
<td>\LARGE</td>
<td>size</td>
<td>17.28pt</td>
</tr>
<tr>
<td>\huge</td>
<td>size</td>
<td>20.74pt</td>
</tr>
<tr>
<td>\Huge</td>
<td>size</td>
<td>24.88pt</td>
</tr>
<tr>
<td>\textnormal</td>
<td>normal</td>
<td>normal text</td>
</tr>
</tbody>
</table>

notoccite This option of the minitoc package loads the notoccite package [14] (by Donald Arsenneau). It avoids problems with \cite commands in sectioning commands or captions. See section 1.6 on page 52.

Omega The Omega typesetting system\(^9\) (\(\Omega\)) (by Yannis Haralambous and John Plaice) is an extension of \TeX{} that is aimed primarily at improving \TeX{}’s multilingual abilities.

When the \TeX{} program was originally developed in the mid seventies [\textit{circa} 1975] by Professor Donald E. Knuth it was mainly aimed at typesetting mathematical texts in the English language. Since then \TeX{} has made inroads in broader and broader areas of scientific, literary and other scholarly activities in many countries all over the world. In 1991, Knuth froze \TeX{}, mainly in the interest of stability. However, he allows the \TeX{} code to be used as the basis for further developments, so long as the resulting system is distributed under a different name.

In Omega all characters and pointers into data-structures are 31-bit wide, instead of 8-bit, thereby eliminating many of the trivial limitations of \TeX{}. Omega also allows multiple input and output character sets, and uses programmable filters.

---

\(^9\) Most but not all of this note is taken in the Omega documentation [217, 218, 221, 378, 379]. See http://omega.ensib.org/ for more information.
Table 6.13: Some systems derived from TEX and L\TeX

\[
\begin{array}{ccc}
\text{TEX} & \rightarrow & \Omega \\
+ & & \text{\L\TeX} \\
\text{TEX} & \rightarrow & \varepsilon - \text{TEX} \\
\end{array}
\]

\begin{array}{c}
\text{NL} \\
\end{array}

\rightarrow

\begin{array}{c}
\Lambda \\
\varepsilon - \text{L\TeX} \\
\end{array}

\]

to translate from one encoding to another, to perform contextual analysis, etc.

Internally, Omega uses the universal Unicode/ISO-10646 character set. Omega also includes support for multiple writing directions.

These improvements not only make it a lot easier for TEX users to cope with multiple or complex languages, like Arabic, Indic, Khmer, Chinese, Japanese or Korean, in one document, but also form the basis for future developments in other areas, such as native color support and hypertext features.

The \L\TeX format (in the \TeX meaning of that word) adapted to the special features of Omega is called “Lambda” (Λ). Extending Omega with the \varepsilon-TEX [105] extensions is a separate project, known as “Aleph” (ℵ) [39, 201] and led by Giuseppe Bilotta. The \L\TeX for Aleph is known as "Lamed" (Ł). There is an experimental system, named Lua\TeX [230, 231], which will regroup \text{PDF}X, Aleph, \varepsilon-TEX and other developments. A promising development is X\L\TeX [256] by Jonathan Kew, with X\L\TeX.

open A type of feature (see this term) which is executed immediately before (open) the insertion of the auxiliary file for a given type of mini-table. Look at the documentation of the \texttt{\mtcsetfeature} command, in section 1.10 on page 48. See the \texttt{mtc-ocf.tex} example file, in section 4.27 on page 137.

P

package Packages\textsuperscript{\footnote{This info is taken from [270, page 12–13] and adapted.}} are a very important feature of \L\TeX. These are extensions to the basic \L\TeX commands that are written to files with names that end with \texttt{.sty} and are loaded with the command \texttt{\usepackage} in the preamble. Packages can be classified by they origin.

- Core packages (in fact, base and required packages) are an integral part of the \L\TeX basic installation and are therefore fully standard.
- Tools packages are a set written by members of the \L\TeX Team and should always be in the installation.
- Graphics packages are a standardized set for including pictures generated by other programs and for handling colors; they are at the same level as the tools packages.

\textsuperscript{\footnote{This info is taken from [270, page 12–13] and adapted.}}
• \texttt{AMS-LaTeX} packages, published by the American Mathematical Society\footnote{http://www.ams.org}, should be in any installation\footnote{They are indispensable if you use a lot of mathematics.}.

• \textbf{Contributed} packages have been submitted by actual users; certain of these have established themselves as “essential” to standard \LaTeX usage, but all are useful.

\texttt{pagestyle} A type of \textit{feature} (see this term) which is executed at each occurrence of a given type of mini-table, to force the page style to use for the current page. Look at the documentation of the \texttt{\setfeature} command, in section 1.10 on page 48.

\texttt{partlof} A list of figures for a part.

\texttt{partlofdepth} This counter, if defined, contains the depth of the \texttt{partlofs}.

\texttt{partlot} A list of tables for a part.

\texttt{partlotdepth} This counter, if defined, contains the depth of the \texttt{partlots}.

\texttt{parttoc} A table of contents for a part.

\texttt{parttocdepth} This counter contains the depth of the \texttt{parttocs}.

\texttt{PDF} Portable Document Format [183]. A descendant of the PostScript language from Adobe, optimized for navigation on the Internet. It adds hypertext, font substitution, and compression features.

\texttt{placeins} The \texttt{placeins} package [15] keeps floats “in their place”, preventing them from floating past a \texttt{\FloatBarrier} command into another section. To use it, declare \texttt{\usepackage{placeins}} in the preamble and insert \texttt{\FloatBarrier} commands at places that floats should not move past, perhaps at every \texttt{\section}. The \texttt{insection} package option of the \texttt{minitoc} package does that with adequate options, and loads also the \texttt{flafter} package (described in [288] and [330, page 286]); see section 1.3.3 on page 29.

\texttt{placeins.txt} A plain text file containing the documentation of the \texttt{placeins} package [15].

\texttt{pL\LaTeX} A version of \LaTeX customized for the polish (\texttt{polski}) language. It has been replaced by the \texttt{polski} package. See [357, 463]. But the \texttt{same} name was referring also to a version of \LaTeX customized for the japanese language, \texttt{P\LaTeX 2E} [239, 254].

\texttt{pmk} An example of shell script to prepare the \texttt{minitoc} package and its documentation; you should adapt it to your needs. See item 10 on page 245.

\texttt{polymorphic entry} An entry in the TOC, LOF or LOT which changes its aspect depending on the place where it appears (main TOC, minitable, etc.); see section 1.4.13 on page 43.

\texttt{PostScript} A page description language, by Adobe. It describes the appearance of a page, including elements such as text, graphics, and scanned images, to a printer or visualization device. Introduced by Adobe in 1985, it has become the language of choice in high quality printing.
preamble  In the main file of a \LaTeX{} source document, the part of it between the commands \texttt{\textbackslash documentclass}[...]{...} and \texttt{\textbackslash begin\{document\}}. In the preamble, you can insert global declarations and the loading of packages via \texttt{\textbackslash usepackage} commands.

preparation  The preparation commands of the \texttt{minitoc} package prepare the auxiliary files for the mini-tables of a given type. A \textit{preparation} command must have been invoked (only once) before any insertion command for the mini-table type. The preparation commands are (see table 3.9 on page 87):

\begin{itemize}
\item \texttt{doparttoc}, \texttt{dopartlof}, \texttt{dopartlot}, (part level)
\item \texttt{dominitoc}, \texttt{dominiolo}, \texttt{dominiolot}, (chapter level)
\item \texttt{dosecttoc}, \texttt{dosectlof}, \texttt{dosectlot}, (section level)
\end{itemize}

\texttt{\textbackslash mtcprepare} (all levels)

proc  A standard \LaTeX{} document class, for preparing conference proceedings. For the \texttt{minitoc} package, it is very similar to the \texttt{article} document class; see above.

pseudo-chapter  Or starred chapter. A chapter introduced by a \texttt{\textbackslash chapter*} command. By default, it has no entry in the table of contents. \texttt{\textbackslash chapter*} needs some precautions with the \texttt{minitoc} package. See section 1.3.4 on page 33.

quotchap  The \texttt{quotchap} package \cite{quotchap} provides a set of commands for adding quotations to some headings (chapters) in the standard \LaTeX{} document classes: \texttt{book}, and \texttt{report}. It must be loaded \textit{before} the \texttt{minitoc} package (see point 1.8 on page 53 and section 2.39 on page 75).

README  is a plain text file (english) describing briefly the \texttt{minitoc} package, plus some useful infos.

report  A standard \LaTeX{} document class \cite{report}. It has sectionning commands: \texttt{\textbackslash part}, \texttt{\textbackslash chapter}, and \texttt{\textbackslash section} (and below). It is compatible with the \texttt{minitoc} package and you can make mini-tables at the part and chapter levels (but not at the section level, to avoid too many auxiliary files).

rmk  An example of shell script, which sorts the files of the \texttt{minitoc} package into classes (one directory for each class). It should be run after the scripts \texttt{imk} (mandatory) and \texttt{emk} and/or \texttt{fmk}, in that sequence. See item 10 on page 245.

romannum  The \texttt{romannum} package \cite{romannum} changes the numbers (for sectionning commands) generated by \LaTeX{} from arabic digits to roman numerals. This package uses the \texttt{stdclsdv} package \cite{stdclsdv}. It must be loaded \textit{before} the \texttt{minitoc} package (see point 1.8 on page 54 and section 2.40 on page 76).
rotating  The rotating [389] package performs all the different sorts of rotation one might like, including complete figures.

rotfloat  The packrotfloat [420] package provides commands to define new floats of various styles (plain, boxed, ruled, and userdefined ones); the rotating package [389] provides new environments (sidewaysfigure and sidewaysstable) which are rotated by 90° or 270°. But what about new rotated floats, e.g. a rotated ruled one? This package makes this possible; it builds a bridge between both packages and extend the commands from the float package to define rotated versions of the new floats, too.

rubber  rubber [34] is a wrapper for \LaTeX{} and companion programs. Its purpose is, given a \LaTeX{} source to process, to compile it enough times to resolve all references, possibly running satellite programs such as \texttt{BibTeX}, \texttt{makeindex}, \texttt{Metapost}, etc., to produce appropriate data files. It has facilities to make some post-processing cleanup actions, like deleting the auxiliary files created by \texttt{minitoc}.

sectionning commands  These are the \LaTeX{} commands which specify the logical structure of your document. The main sectionning commands are \texttt{\part}, \texttt{\chapter}, \texttt{\section}, \texttt{\subsection}, \texttt{\subsubsection}, \texttt{\paragraph}, or \texttt{\subparagraph}. Some standard document classes have not the \texttt{\chapter} command (like the \texttt{article} and \texttt{proc} classes), some have no sectionning commands (like the \texttt{letter} class). In the later case, the \texttt{minitoc} package is pointless. If some of the \texttt{\part}, \texttt{\chapter}, or \texttt{\section} commands are not defined, the \texttt{minitoc} commands for that level are unavailable. If \texttt{\chapter} is defined, the \texttt{minitoc} commands at the section level are defined in the current and older versions of the \texttt{minitoc} package, but if \texttt{\chapter} is not defined and \texttt{\section} is defined, then the \texttt{minitoc} commands at the section level are defined. See section 1.1.1 on page 26. In non-standard document classes, sectionning commands with non-standard names cannot be recognized by the \texttt{minitoc} package.

sectlof  A list of figures for a section.

sectlofdepth  This counter, if defined, contains the depth of the sectlofs.

sectlot  A list of tables for a section.

sectlotdepth  This counter, if defined, contains the depth of the sectlots.

sectsty  The sectsty package [319] provides a set of commands for changing the font used for the various sectional headings in the standard \LaTeX{} 2e document classes: article, book, and report. This package also works with the KOMA-Script classes [343, 344, 399] \texttt{scrartcl}, \texttt{scrbook}, and \texttt{scrreprt}. It must be loaded \texttt{before} the \texttt{minitoc} package (see point 1.8 on page 53 and section 2.28 on page 70).
secttoc A table of contents for a section.

secttocdepth This counter contains the depth of the secttocs.

sfheaders The sfheaders package borrows some definitions from the standard article/report/book classes and modifies them in order to print the part, chapter, section, subsection... headers with the Sans-Serif variant of the current font. It must be loaded before the minitoc package (see point 1.8 on page 54 and section 2.41 on page 76).

shell In the Unix, Unix-like and Linux operating systems, the shell is a program used as an interface between the operating system and the user. It can also be used as a scripting language to write programs or scripts to prepare routinely used sequences of tasks. The main shells are the Bourne shell (sh), the C shell (csh), the Korn shell (ksh), and their many successors (like bash, tcsh, etc.).

shortext An option of the minitoc package. It forces the use of short extensions (3 characters) in the names of the minitoc auxiliary files. This option is inactive by default, but is automatically activated if your operating system needs short extensions. See autoconfiguration above and the section 2.5 on page 58.

shorttoc The shorttoc package allows to create an other table of contents in a document, with an other title and an other depth than the main table of contents.

SJIS The SJIS character encoding (for the japanese language), also known as MS-Kanji (Kanji for Microsoft®), consists of two overlaid character sets: the so-called halfwidth Katakana (JIS X0201-1976, 1-byte characters encoded in the range 0xA1 to 0xDF) and the (fullwidth) JIS character set (JIS X0208-1990, mapped to the remaining code points). This information is taken from [127, 297, 298].

SL A version of LATEX customized for the swedish language. See [318].

splitbib A LATEX package which allows for sorting a bibliography into categories and subcategories; this is interesting for lists of publications, for grouping references by subject, by year, ...

stdclsdv The stdclsdv package is intended to be used by the authors of LATEX packages that need to know about the sectional divisions provided by the document class.

strut A vertical invisible rule used to force a minimal separation between two lines of text.

subfig The subfig package provides support for the inclusion of small, “subfigures” and “sub-tables”. It simplifies the positioning, captioning and labeling of them within a single figure or table environment. In addition, this package allows such sub-captions to be written to the List of Figures or List of Tables if desired.

subfigure The subfigure package is an obsolete version (by the same author) of the subfig package.

suffix See “extension” above.
The \TeX{} Directory Structure [445, 446]; a directory structure highly recommended to store macros, fonts, and the other implementation-independent \TeX{} system files; it also suggests how to incorporate the rest of the \TeX{} files in a single structure; the TDS has been designed to work on all modern systems.

\TeX{} \TeX{} is a computer program created by Donald E. Knuth [263, 265]. It is aimed at typesetting text and mathematical formulae. Knuth started writing the \TeX{} typesetting engine in 1977 to explore the potential of the digital printing equipment that was beginning to infiltrate the publishing industry at that time, especially in the hope that he could reverse the trend of deteriorating typographical quality that he saw affecting his own books and articles. \TeX{} as we use it today was released in 1982, with some slight enhancements added in 1989 to better support 8-bit characters and multiple languages. \TeX{} is renowned for being extremely stable, for running on many different kinds of computers, and for being virtually bug free. The version number of \TeX{} is converging to \( \pi \) and is now at 3.141592.

\TeX{} is pronounced “Tech,” with a “ch” as in the German word “Ach” or in the Scottish “Loch.” In an ASCII environment, \TeX{} becomes \TeX{}.

\texttt{thailatex} The \texttt{thailatex} package [320] allows to typeset documents in the Thai language. You can also use the CJK system [127, 297, 298].

\texttt{tight} An option of the \texttt{minitoc} package. It gives a tight line spacing in the mini-tables. The opposite option is \texttt{loose}.

\texttt{titlesec} The \texttt{titlesec} package [46] allows to change the sectioning titles. Amongst its many features it provides margin titles, different format in left and right pages, rules above and below the title, etc. Unfortunately, it is \textit{incompatible} with the \texttt{minitoc} package.

\texttt{titletoc} The \texttt{titletoc} package is useful for toc entries formatting, providing the possibility of changing the format in the middle of a document, grouping the entries in a single paragraph, pretty free-forms entries, partial tocs, etc. Unfortunately, it is \textit{incompatible} with the \texttt{minitoc} package.

\texttt{titletoc.sty} file is not part of the \texttt{titlesec} package; it’s an independent package, but it’s described in the \texttt{titlesec} package documentation [46].

\texttt{tmk} A script file which creates a TDS-compliant hierarchy [445, 446] (to be adjusted to your system).

\texttt{TOC, ToC} Acronym for “table of contents”.

\texttt{tocbibind} The \texttt{tocbibind} package [472] can be used to add the ToC and/or bibliography and/or the index etc., to the Table of Contents listing. But it needs some precautions when used with the \texttt{minitoc} package. See section 1.5.5 on page 50.

\texttt{tocdepth} This counter contains the depth of the table of contents.

\texttt{tocloft} The \texttt{tocloft} package [469] provides means of controlling the typographic design of the Table of Contents, List of Figures and List of Tables. New kinds of “List of …” can be defined. If you use the \texttt{tocloft} package and the \texttt{minitoc} package, see section 2.21 on page 64 about fixing some minor compatibility issues.
TODO is a plain text file (english) which lists some suggested developments of the package, not yet implemented. Comments and suggestions are welcome.

token A token is either (a) a single character with an attached category code (see “cat-code” above), or (b) a control sequence. You should remember two chief things about \TeX’s tokens: (1) A control sequence is considered to be a single object that is no longer composed of a sequence of symbols. Therefore long control sequence names are no harder for \TeX to deal with than short ones, after they have been replaced by tokens. Furthermore, spaces are not ignored after control sequences inside a token list; the ignore-space rule applies only in an input file, during the time that strings of characters are being tokenized. (2) Once a category code has been attached to a character token, the attachment is permanent. For instance, if character ‘{’ were suddenly declared to be of category 12 instead of category 1, the characters ‘{1’ already inside token lists of \TeX would still remain of category 1; only newly made lists would contain ‘{12’ tokens. In other words, individual characters receive a fixed interpretation as soon as they have been read from a file, based on the category they have at the time of reading. Control sequences are different, since they can change their interpretation at any time. \TeX’s digestive processes always know exactly what a character token signifies, because the category code appears in the token itself; but when the digestive processes encounter a control sequence token, they must look up the current definition of that control sequence in order to figure out what it means.

trivfloat The trivfloat package [484] (by Joseph A. Wright) provides a quick method for defining new float types in \LaTeX. A single command sets up a new float in the same style as the \LaTeX kernel figure and table float types.

txfonts The txfonts package [403] provides the \TeX fonts, which consist of

1. virtual text roman fonts using Adobe Times (or URW NimbusRomNo9L) with some modified and additional text symbols in OT1, T1, TS1, and LY1 encodings;
2. virtual text sans serif fonts using Adobe Helvetica (or URW NimbusSanL) with additional text symbols in OT1, T1, TS1, and LY1 encodings;
3. monospaced typewriter fonts in the OT1, T1, TS1, and LY1 encodings;
4. math alphabets using Adobe Times (or URW NimbusRomNo9L) with modified metrics;
5. math fonts of all symbols corresponding to those of Computer Modern math fonts (CMSY, CMMI, CMEX, and Greek letters of CMR);
6. math fonts of all symbols corresponding to those of AmS fonts (MSAM and MSBM);
7. additional math fonts of various symbols.

All fonts are in the Type 1 format (in .afm and .pfb files). Necessary .tfm and .vf files together with \TeX 2ε package files and font map files (.map) for dvips are provided.

13This definition is taken from “The \TeXbook” [263, 265].
UNIX  A modern operating system, available on many computers and in various flavors. From the minitoc point of view, it has the advantage of using filenames with long extensions (the length limit is too high to be a problem with the number of mini-tables).

UNIX-like Operating systems analog to Unix, with the same advantages. Linux is a good example, but others exist.

urlbst A PERL script, by Norman Gray [196], to add a webpage BibTEX entry type, and add support for general url and lastchecked fields, to (most) BibTEX bst files. Optionally adds basic support for eprint and doi fields, and HyperTEX/hyperref support, too. See [371, 390].

UTF 8 UTF 8 (Unicode Transformation Format 8), also called UTF 2 or FSS-UTF, is a special representation of Unicode (resp. ISO 10 646). It uses multibyte sequences of various lengths, but only 2-byte and 3-byte sequences are implemented in CJK. ASCII characters will be used as-is — without this property it would be impossible to use UTF 8 with \TeX. See table 6.2 on page 212.

varsects The varsects package [437] provides a set of commands for changing the font used for the various sectional headings in the standard \LaTeX document classes: article, book, and report. It must be loaded before the minitoc package (see point 1.8 on page 53 and section 2.33 on page 73).

Wikipedia The Wikipedia is a free, multilingual, open content (neutral, verifiable, modifiable and improvable by anyone) encyclopedia project operated by the non-profit Wikimedia Foundation. Its name is a portmanteau of the words wiki (a type of collaborative website) and encyclopedia. Launched in 2001 by Jimmy Wales and Larry Sanger, it is the largest, fastest growing and most popular general reference work currently available on the Internet.

wrapfig The wrapfig package [18] provides the wrapfigure and wraptable environments to place a figure or table at the side of the page and wrap text around it.
**xmk**  An example of shell script, which typesets the example document files into PDF documents.

**xr**  The xr package [114] implements a system for eXternal References. I wrote the first version of this package, but it had severe problems. David P. Carlisle rewrote it in a much better and more robust way. With his permission, I used some of his code in the minitoc package to implement the preparation commands (like \dominitoc). If you use also the hyperref package [390], use xr-hyper [117] in place of the xr package.
Chapter 7

Installation

Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1</td>
<td>List of files (minitoc.l), first part</td>
<td>243</td>
</tr>
<tr>
<td>7.2</td>
<td>List of files (minitoc.l), second part</td>
<td>244</td>
</tr>
<tr>
<td>7.3</td>
<td>List of the graphic files</td>
<td>244</td>
</tr>
<tr>
<td>7.4</td>
<td>List of the flag files</td>
<td>244</td>
</tr>
<tr>
<td>7.5</td>
<td>A TDS-compliant hierarchy for the minitoc files</td>
<td>247</td>
</tr>
</tbody>
</table>

This chapter describes the installation of the minitoc package (version #61).

This package contains a lot of files. The list of all files is given in minitoc.l. See tables 7.1 to 7.2 on pages 243–244. The files are sorted into “classes” below (a file can appear in more than one class). Each class specifies the function and the placement of its files.

(0) The files minitoc.ins and minitoc.dtx are the basic source files of this package.

The file minitoc-fr.dtx loads minitoc.dtx but selects the french documentation.

The language selection is done by using \ifcase ... \or ... \fi constructs.

(1) The files minitoc.sty, mtcdef.sty, mtcmess.sty, and all *.mld and *.mlo files are the package itself.

The table 1.7 on page 38 lists the available languages; for each of these languages, a language*.mld file is available; the languages in parentheses are aliases of a main language and their .mld files will load the .mld file of that main language.

mtcpatchmem.sty is a temporary fix for compatibility with the memoir class.

The files of this class must be all installed in a directory where \if\pdf\else\latex2e\fi finds the .sty files.

\[ ^{1} \] The large number of *.mld files is (partially) a consequence of the fact that some languages have aliases (or dialects) and hence one *.mld file for each name (a *.mld file may load another one) and, if necessary, a .mlo file; the english and french languages are evident examples. For some languages, the multiplicity of the *.mld files corresponds to a multiplicity of fonts and/or encodings (chinese, greek, japanese, korean, malayalam, polish, russia, serbian), or even for spelling reforms (german, greek, norsk). Note that the presence of the english.mld file is mandatory. Since version #50, the minitoc package signals the missing .mld or .mlo files and gives their list in a warning message.
### Table 7.1: List of files (minitoc.1), first part

<table>
<thead>
<tr>
<th>Class (0):</th>
<th>-finnish.mld</th>
<th>-magyar.mld</th>
<th>-spanish.mld</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>-franc.mld</td>
<td>-malay.mld</td>
<td>-swahilli.mld</td>
</tr>
<tr>
<td></td>
<td>-french.mld</td>
<td>-malayalam.mld</td>
<td>-swedish.mld</td>
</tr>
<tr>
<td></td>
<td>-french1.mld</td>
<td>-malayalam-keli.mld</td>
<td>-thai.mld</td>
</tr>
<tr>
<td></td>
<td>-french2.mld</td>
<td>-malayalam-keli2.mld</td>
<td>-turkish.mld</td>
</tr>
<tr>
<td></td>
<td>-french2.mld</td>
<td></td>
<td>-uighur.mld</td>
</tr>
<tr>
<td></td>
<td>-french1.mld</td>
<td></td>
<td>-uighur3.mld</td>
</tr>
<tr>
<td></td>
<td>-french.mld</td>
<td></td>
<td>-vietnam.mld</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>-vietnamese.mld</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>-welsh.mld</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>-xalx.mld</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>-xalx2.mld</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>-xalx3.mld</td>
</tr>
<tr>
<td>Class (1):</td>
<td>-minitoc.1.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-minitoc.fst.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-minitoc-fr.fst.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-minitoc.ins.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-mcoff.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-mtcmess.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-mtcpatchm.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-acadian.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-academic.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-american.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-arab.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-arab2.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-arabi.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-arabic.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-armenian.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-australian.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-austrian.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-bahasa.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-baha.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-baham.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-bangla.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-banj.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-bengali.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-bich.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-bicher2.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-binh.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-binh2.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-binh3.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-bithie.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-brazil.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-brazilian.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-breton.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-british.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-bulgarian.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-bulgarianb.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-burr.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-burr2.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-canadian.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-canadien.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-castallan.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-castill.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-castilian.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-chinese1.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-chinese2.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-croatian.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-czech.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-danish.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-devanagari.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-dutch.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-english.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-english1.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-english2.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-esperant.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-esperanto.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-estonian.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-ethiopia.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-ethipian.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-ethipian2.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-farsj1.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-farsj2.mld</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-finnish.mld</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

See continuation in table 7.2 on the following page.
(2) Informative text files:

- INSTALL is a file describing the installation of the package. You are (almost) reading it (but it is shorter).
- minitoc.l contains the list of all files of the minitoc distribution. See tables 7.1 to 7.2 on pages 243–244.
- README is a file describing briefly the minitoc package, plus some useful infos.
- TODO lists some suggested developments of the package, not yet implemented. Comments and suggestions are welcome.


(4) minitoc.bug, minitoc.sum are plain text documentation: list of problems (FAQ, see chapter 2 on page 56) and summary of commands (see chapter 3 on page 80).

(5) minitoc.ins, minitoc.ist, minitoc.pre, minitoc.lan, en-mtc.bst, minitoc.dtx and minitoc.bib are the source of the documentation in (non perfect) English. minitoc.pre is the common preamble code for the documentation.
(6) The list of the graphic and flag files is given in tables 7.3 to 7.4 on the page before; they are images to include.

(7) \texttt{minitoc.dtx}, \texttt{minitoc-fr.dtx}, \texttt{minitoc-fr.bib}, \texttt{minitoc-fr.ist}, \texttt{minitoc-fr.lan}, \texttt{minitoc.pre}, \texttt{franc.sty}, \texttt{frbib.sty}, \texttt{frnew.sty}, \texttt{fr-mtc.bst} are the source (and tools) of the documentation in French.\footnote{This seems rather strange. In fact, the English and French documentations are both contained in the \texttt{minitoc.dtx} file. \texttt{minitoc-fr.dtx} sets a flag then loads \texttt{minitoc.dtx}; hence the file \texttt{minitoc-fr.dtx} is much smaller than \texttt{minitoc.dtx}. Thus, \texttt{minitoc.ins} contains also some utilitary files which are automatically created (some \texttt{.sty} files, \texttt{minitoc.ist}, \texttt{minitoc-fr.ist}, \texttt{minitoc.lan}, \texttt{minitoc-fr.lan}). The English and French versions are not word-by-word translation, but they are in parallel in the \texttt{minitoc.dtx} file, and this helps the maintenance.}

(8) \texttt{minitoc.pdf} is the documentation in (non perfect) English, in PDF format.

(9) \texttt{minitoc-fr.pdf} is the documentation in French, in PDF format. The French documentation and its source files must not be left out.

(10) \texttt{pmk} is a shell script\footnote{You can sip a big cappuccino \texttt{K} while this script is running! Be patient.} to prepare the package and its documentation; the \texttt{pmk} script uses the \texttt{/tmp/'whoami'.imk} and \texttt{/tmp/'whoami'.tmk} directories to not waste disk space under your home directory; there are also six partial scripts\footnote{In fact, \texttt{pmk} assembles the scripts \texttt{imk}, \texttt{emk}, \texttt{fmk}, \texttt{xmk}, \texttt{rmk}, and \texttt{tmk} (but not \texttt{cmk}).} and a supplementary one, \texttt{cmk} (all to be adapted):

- \texttt{imk}, which prepares the package from \texttt{minitoc.ins} and \texttt{minitoc.dtx}; note that \texttt{imk} must be run before running \texttt{emk} or \texttt{fmk}; it creates also some \texttt{.sty} files necessary to prepare the documentation but that are to be installed with it; the \texttt{imk} script uses the \texttt{/tmp/'whoami'.imk} directory to not waste disk space under your home directory;

- \texttt{emk}, which prepares the English documentation from \texttt{minitoc.dtx};

- \texttt{fmk}, which prepares the French documentation from \texttt{minitoc-fr.dtx} and \texttt{minitoc.dtx};

- \texttt{xmk}, which typesets the example files (in PDF format);

- \texttt{rmk}, which sorts the files into classes (one directory for each class);

- \texttt{tmk}, which creates a TDS-compliant hierarchy \cite{445,446} (to be tailored to your system); see table 7.5 on page 247; this hierarchy is saved in \texttt{minitoc.tds.zip};

- \texttt{cmk}, which converts the PDF documentation files into PostScript files.

These scripts are currently written in C-shell, but they are very simple, and should be easy to convert in another classic shell. The documentation in PostScript format is no more distributed on the CTAN archives, but the \texttt{cmk} script can prepare it from the documentation in PDF format (recto-verso printing).

(11) \texttt{minitoc.tds.zip} is a \texttt{ZIP}-archive file containing a TDS-compliant hierarchy with all the files of the \texttt{minitoc} package.
Some remarks about the \texttt{rmk}, \texttt{tmk} and \texttt{pmk} scripts (which you should tailor to your needs):

- **with \texttt{rmk}:**
  1. the hierarchy is \textit{not} TDS-compliant;
  2. the files of (0) must be installed in a directory where \LaTeXe\ finds .dtx and .ins files;
  3. the files of (1) must be installed in a directory where \LaTeXe\ finds .sty files;
  4. the files of (2), (3), (4), (5), (6), (7) and (10) must be installed in a separate directory, but must not be left out;
  5. the files of (8) and (9) must be installed as on-line documentation;
  6. the directories created by the \texttt{rmk} script are under /tmp/`whoami`.rmk to not waste disk space under your home directory.

- **with \texttt{tmk}:**
  1. the hierarchy is TDS-compliant;
  2. each file appears only once in the hierarchy;
  3. the installation is much easier: you only need a .zip or a .tar (or .tgz) dump file\footnote{The \texttt{tmk} script creates the \texttt{minitoc.tds.zip} file.} of the hierarchy to be deployed into the installed TDS hierarchy; you should examine \textit{very carefully} (by comparison with your TDS installation) and tailor the \texttt{tmk} script before using it;
  4. the directories created by the \texttt{tmk} script are under /tmp/`whoami`.tmk to not waste disk space under your home directory;
  5. the file \texttt{minitoc.tds.zip} (11) should not be installed; it is just a method to help making a TDS-compliant installation.

- **with \texttt{pmk}:**
  1. the \texttt{pmk} script performs the actions of \texttt{imk} (preparation of the basic files), \texttt{emk} and \texttt{fmk} (preparation of the english and french documentation), \texttt{rmk} and \texttt{tmk} (preparation of the examples of documents), \texttt{rmk} and \texttt{tmk} (repartition of files into classes and in a TDS-compliant hierarchy);
  2. the directories created by the \texttt{pmk} script are under /tmp/`whoami`.rmk and /tmp/`whoami`.tmk to not waste disk space under your home directory;
  3. the \textit{same precautions} as for \texttt{tmk} are needed.

The file \texttt{minitoc.tds.zip} contains a ZIP archive of a TDS-compliant hierarchy of all files of the \texttt{minitoc} package. It has been prepared by the \texttt{pmk} or \texttt{tmk} scripts.

Note that \texttt{minitoc.dtx} and hence \texttt{minitoc-fr.dtx} are (not so trivial) examples of using \texttt{minitoc} with \texttt{hyperref}. They show how the combinaison of these two packages may be useful.
Table 7.5: A TDS-compliant hierarchy for the minitoc files

<table>
<thead>
<tr>
<th>TDS Root Directory</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>bibtex/</td>
<td>doc/</td>
<td>makeindex/</td>
<td>scripts/</td>
<td>source/</td>
<td>tex/</td>
</tr>
<tr>
<td>bib/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bst/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>latex/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>minitoc/</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>minitoc/</td>
<td>(A)</td>
<td>(B)</td>
<td>(C)</td>
<td>(D)</td>
<td>(E)</td>
</tr>
<tr>
<td>examples/</td>
<td>(F)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>images/</td>
<td>(G)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(A) minitoc-fr.bib, minitoc.bib;  
(B) en-mtc.bst, fr-mtc.bst;  
(C) INSTALL, README, TODO, minitoc-fr.lan,  
    minitoc-fr.pdf, minitoc.bug, minitoc.1,  
    minitoc.lan, minitoc.pdf, minitoc.pre,  
    minitoc.sum, Franco.sty, frn.bib, sty, frnew.sty;  
(D) minitoc-fr.ist, minitoc.ist;  
(E) cmk, enk, fnk, ink, pmk, rnk, tkk, xmk;  
(F) minitoc-fr.dtx, minitoc.dtx,  
    minitoc.ins;  
(G) minitoc.sty, mtcoff.sty, mtcmess.sty,  
    mtcpatchmem.sty, *.mld, *.mlo;  
(H) mtc-2c.pdf, mtc-2c.tex, mtc-2nd.pdf,  
    mtc-2nd.tex, mtc-3co.pdf, mtc-3co.tex,  
    mtc-add.bib, mtc-add.pdf, mtc-add.tex,  
    mtc-ads.pdf, mtc-ads.tex, mtc-amm.pdf,  
    mtc-amn.tex, mtc-apx.pdf, mtc-apx.tex,  
    mtc-art.pdf, mtc-art.tex, mtc-bk.pdf,  
    mtc-bk.tex, mtc-bo.pdf, mtc-bo.tex,  
    mtc-ch0.pdf, mtc-ch0.tex, mtc-cri.pdf,  
    mtc-cri.tex, mtc-fko.pdf, mtc-fko.tex,  
    mtc-fol.pdf, mtc-fol.tex, mtc-f02.pdf,  
    mtc-f02.tex, mtc-gap.pdf, mtc-gap.tex,  
    mtc-h11.pdf, mtc-h11.tex, mtc-h12.pdf,  
    mtc-h12.tex, mtc-hia.pdf, mtc-hia.tex,  
    mtc-hir.pdf, mtc-hir.tex, mtc-hop.pdf,  
    mtc-hop.tex, mtc-liv.pdf, mtc-liv.tex,  
    mtc-nem.pdf, mtc-nem.lan, mtc-nml.pdf,  
    mtc-nml.tex, mtc-mu.pdf, mtc-mu.tex,  
    mtc-nom.pdf, mtc-nom.lan, mtc-ocf.pdf,  
    mtc-ocf.tex, mtc-ofs.pdf, mtc-ofs.tex,  
    mtc-sbf.pdf, mtc-sbf.tex, mtc-scr.pdf,  
    mtc-scr.tex, mtc-syn.pdf, mtc-syn.tex,  
    mtc-tbi.pdf, mtc-tbi.lan, mtc-tlc.pdf,  
    mtc-tlc.tex, mtc-tlo.pdf, mtc-tlo.tex,  
    mtc-tsf.pdf, mtc-tsf.tex, mtc-vti.pdf,  
    mtc-vti.tex;  
(I) The list of the graphic and flag files is given in  
    tables 7.3 to 7.4 on page 244.
Chapter 8

Postface

Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.1 The origins</td>
<td>248</td>
</tr>
<tr>
<td>8.2 New design in 1993</td>
<td>249</td>
</tr>
<tr>
<td>8.3 Developments in 1994</td>
<td>249</td>
</tr>
<tr>
<td>8.4 No developments in 1995</td>
<td>249</td>
</tr>
<tr>
<td>8.5 Developments in 1996</td>
<td>250</td>
</tr>
<tr>
<td>8.6 Developments in 1997</td>
<td>250</td>
</tr>
<tr>
<td>8.7 Developments in 1998</td>
<td>250</td>
</tr>
<tr>
<td>8.8 Developments in 1999</td>
<td>250</td>
</tr>
<tr>
<td>8.9 Developments in 2000</td>
<td>250</td>
</tr>
<tr>
<td>8.10 Developments in 2001</td>
<td>251</td>
</tr>
<tr>
<td>8.11 Developments in 2002</td>
<td>251</td>
</tr>
<tr>
<td>8.12 Developments in 2003</td>
<td>251</td>
</tr>
<tr>
<td>8.13 Developments in 2004</td>
<td>251</td>
</tr>
<tr>
<td>8.14 Developments in 2005</td>
<td>252</td>
</tr>
<tr>
<td>8.15 Developments in 2006</td>
<td>254</td>
</tr>
<tr>
<td>8.16 Developments in 2007</td>
<td>254</td>
</tr>
<tr>
<td>8.17 Developments in 2008</td>
<td>258</td>
</tr>
</tbody>
</table>

This chapter summarizes the evolution of the minitoc package, year by year. A more detailed history is available in “Changes History”, page 597. Many minor changes are skipped here.

In fact, this chapter is for the average user of the package, who wants to have an overview of its evolution; the chapter “Changes History” is more oriented towards peoples interested in the code of the package and the problems encountered during its development.

8.1 The origins

The minitoc package was initially written by Nigel Ward in 1990 and 1991, with major contributions by Dan Jurafsky. But minitoc suffered of a major weakness: when the number of chapters exceeded 9 or 10, you got a rather mysterious error message:

No room for a new \write.

As I needed the basic fonctionnality of this package (printing mini-tables of contents for each chapter), I looked further in its code and finally found the culprit: minitoc used a \newwrite
command to create a new file for the contents table of each chapter, allocating a new file
descriptor each time. But the number of file descriptors for writing is limited to 16 under
\LaTeX (in fact, by the underlying \TeX program itself). As some descriptors are already used
by \LaTeX, writing more than 9 or 10 chapters was too much. Such errors are difficult to find
when testing on too small documents: with few chapters, everything goes fine. But on a real
document, with many chapters, the mysterious error happens.

8.2 New design in 1993

So I decided (June 1993, when I took the maintenance of the package) to change the allo-
cation method to always use the same file descriptor for all the mini-table of contents files.
Some major improvements happened in 1993: the addition of the \texttt{mtcoff} (\texttt{minitocoff} at this
time) package and a rewrite of \texttt{minitoc} to extract the data from the \texttt{document.toc} file, with a
selection mechanism. Then a first solution for the short extension problem was added (still
manual). An elementary system for the fonts in the mini-tables was added. In December
1993, the minilofs and minilots were added.

8.3 Developments in 1994

The first improvements in 1994 were about the formatting of the mini-tables: spacing was
improved and the position of the title became adjustable (with the optional parameter of
\texttt{\minitoc} or \texttt{\dominitoc}). But a major addition was done: part-level mini-tables (parttocs,
partlofs, partlots) and, for articles, section-level mini-tables (secttocs, sectlofs, sectlots).

With the emergence of \LaTeX\texttt{2e}, replacing the ageing \LaTeX\texttt{2.09}, some work was necessary
to support the compatibility with this new version. This was not easy, but Denis B. \textsc{Roegel}
and Frank \textsc{Mittelbach} gave me many helpful hints.

Another major addition is the language option feature, with the concept of the mini-
toc language definition file (or \texttt{.mld} file), coming from the \texttt{babel} package \cite{60,61}, by
Johannes L. \textsc{Braams}. But at this time, these files were still named as \texttt{.sty} files.

An important simplification is introduced by the notion of “absolute numbering”, with avoids
many problems when the chapters are not numbered the standard way (consecutively, starting
from 1, with arabic digits).

8.4 No developments in 1995

Sorry, I was busy with an other project.
8.5 Developments in 1996

The very annoying problem with the starred sectionning commands received some (rather primitive) solutions, but it is very complex, so manual interventions are often required.

The names of the minitoc language definition files take now the extension \texttt{.mld}, more specific, and english is the default language. Some new languages are added.

The work on the starred sectionning commands continues.

The \texttt{minitocoff} companion package is renamed \texttt{mtcoff} to keep its name short.

8.6 Developments in 1997

The problem of short extensions for files names under some operating systems is addressed via the new \texttt{shortext} package option and by the new autoconfiguration feature.

For the starred sectionning commands, the \texttt{\addstarredchapter} command is added (with analog commands for starred parts and sections).

8.7 Developments in 1998

The \texttt{tight} and \texttt{loose} package options are added to improve the line spacing in the mini-tables.

8.8 Developments in 1999

The \texttt{dotted} and \texttt{undotted} package options are added to add or suppress some lines of dots (leaders) in the mini-tables.

8.9 Developments in 2000

A major addition is the compatibility with the \texttt{hyperref} package \cite{hyperref}, and I ought to thank loudly Heiko Oberdiek, Didier Verna, Bernd Jaehne and A. J. “Tony” Roberts \cite{hyperref}.

Some corrections about the starred sectionning commands are added by Heiko Oberdiek.

The mini-tables features (\texttt{\befoerparttoc} and co.) commands are added.
Some adjustment commands, like `\mtcaddchapter`, are added, again about the problem with the starred sectionning commands.

The compatibility with the `tocbibind` package [472] is documented.

Aliases for some languages are added.

### 8.10 Developments in 2001

Added the `checkfiles` and `nocheckfiles` package options, to avoid the insertion of (ugly) empty mini-tables.

Added the `\mtcselectlanguage` command to change more easily the language of the mini-tables titles.

### 8.11 Developments in 2002

Correction of an interaction between `\tableofcontents` (creating a hidden `\chapter*` or `\section*` command) and the numbering of the mini-table files.

Added the `\mtcskip` and `\mtcskipamount` commands.

### 8.12 Developments in 2003

Added the `insection` package option (which was temporarily done by the `flsection` and `flsectionb` package options) to deal with floats drifting out of their section. The `placeins` package [15] (by Donald Arseneau) is used.

The font commands are made compatible with the `memoir` class [479, 481, 482].

Added compatibility with the `notoccite` package [14].

### 8.13 Developments in 2004

Added comments in the `.mld` files needing special fonts. Better documentation about languages.

Added an explanation about making a local table of contents for an appendix, eventually masking it in the main table of contents. Compatibility with the `appendix` package [471].
A major addition is the hints package option, to detect some programming and compatibility problems.

8.14 Developments in 2005

All messages are now written via the standard interface commands (\PackageInfo, \PackageWarning, and \PackageError), so the minitoc package is less verbose on the terminal.

Added the \mtcsetfont and \mtcsettitlefont commands (from a suggestion by Benjamin Bayart) to replace many font commands by only one command with a better user interface.

Comments about the AMS classes (some ones are incompatible with minitoc).

Added the \mtcsetformat and \mtcsettitle commands, again to have a simpler user interface.

Added various hints (insection package option, order of minitoc basic commands, short extensions).

Added the \mtcsetpagenumbers and \mtcsetrules commands, again to have a simpler user interface.

Added the mtchideinmaintoc environment, to hide a group of entries in the main table of contents; added also the mtchideinmainlof and mtchideinmainlot environments.

Added the \mtcfixindex and \mtcsettitle commands.

Added the description of the installation of the package (a new chapter and the file INSTALL).

Improved and added hints about consistency of \dominitoc/\minitoc and co.

Added the \mtcsetfeature command (very complex).

Added a hint about the abstract package [470].

The minitoc package is now written using the .dtx-.ins system. Some cleanup is done in the code.

Added the \mtcfixglossary command, like \mtcfixindex.

Some improvements are made to print the documentation.

Some new hints are added (sectsty package [319], empty mini-tables, obsolete commands).
Added the notion of depth for mini-tables of figures/tables. Added the `\mtcsetdepth` command.

The `hints` package option is now the default.

Added a method for making a bilingual documentation in one file (the `minitoc.dtx` file). This method could be used for more languages.

Added or improved some adjustment commands (`\adjustptc`, `\incrementptc`, `\decrementptc`, etc.).

Added the `k-tight` and `k-loose` package options, for the KOMA-Script [343, 344, 399] document classes.

Added a patch for the recent version of the `memoir` class [479, 481, 482].

Use `\mtcselectlanguage` in language options and in “secondary” `.mld` files.

Added the `\mtcloadmlo` command to be used in some `.mld` files to load a `.mlo` file. The extension `.mlo` means `minitoc` language object; such files contain characters not easily manipulated in a `.dtx` file.

The history of changes is now displayed in a much simpler way (using a glossary was too cumbersome).

Added the `listfiles` package option, to create a list of the minitoc auxiliary files, which can be removed after the `\LaTeX` compilation of the document. It is the `document.maf` file.

Added a remark in the FAQ chapter (and `minitoc.bug`) about precautions to take with the starred sectioning commands.

Added hints about the `caption`, `caption2`, `ccaption`, and `mcaption` packages (they must be loaded before `minitoc`).

A “Jargon” chapter is added. It will grow slowly.

Added a note about a problem with `minitoc`, `hyperref` and `memoir` used together.

Some bugs in the `\mtcset...` commands are fixed.

Added a hint about the `varsects` package [437].

Added a hint on the number of mini-tables when short extensions are used.

Added a chapter with all the (explained) messages.
8.15 Developments in 2006

Added the “*” keyword as first argument of the \mtcsetpagenumbers and \mtcsetrules command, to get an action on all kinds of mini-tables.

Corrections in the \mtcaddsection, \mtcfixglossary, and \mtcfixindex commands.

In the PDF documentation, the panel of bookmarks shows initially only the bookmarks for parts and chapters, but you can open them to show deeper entries.

Added a comment about the initialization of fonts in the FAQ (point 34). It is still an open domain and I am working on it.

Added a hint about the KOMA-Script classes [343, 344, 399], and an entry in the FAQ chapter (and in minitoc.bug).

Added the “Postface” chapter.

Added the \mtcprepare command.

Added an URL field in the bibliography (the styles are modified with the urlbst tool [196]).

Added the mtcmess package to add unique identifiers to the messages.

Suppressed the PostScript documentation files from the distribution (no more accepted on CTAN archives), but the scripts still creates them.

Corrections in the insection package option.

Reordering of the chapters in the user’s manual (part I).

8.16 Developments in 2007

Removed the preparation of documentation in PostScript format.

Added the \mkt script to convert the documentation from PDF format to PostScript format.

Added hints about the fnychap [301], quotchap [442], romannum [480], sfheaders [304], alnumsec [274], and captcont [131] packages.

Added FAQ 44 and the \mtcgapbeforeheads and \mtcgapafterheads formatting commands.

Added the chapter 4, “Examples of documents”, page 90.
Added FAQ 45 and the \kernafterparttoc and co. commands for the vertical space between a minitable and its bottom rule.

Increased the text width and adjusted the format of the entries in the TOC in the documentation.

Correction of the preamble in the generated files (spurious lines have been eliminated, at least).

Added devanagari.mld and hindi.mld. Added hindi-modern.mld.

The bibliographic styles plainurl.bst and frplain1.bst are renamed en-mtc.bst and fr-mtc.bst.

All example documents are renamed with names beginning with “mtc-”.

The “listfiles” package option is active by default.

Better error messages about undefined preparation and insertion commands.

Added japanese6.mld and japanese6.mlo.

Added a hint about the hangcaption package [250].

Added (in the memento) a table of the classes and packages which are incompatible or need precautions with minitoc.

Added a validation of the language options with the presence of the .mld and .mlo files.

Added the tmk script and a table describing a TDS-compliant structure for minitoc.

Updated the INSTALL file and the “Installation” chapter.

Added the file minitoc.tds.zip (a ZIP archive of a TDS-compliant hierarchy of all files of the package) to the distribution.

Improving the index (entries for: packages and classes, scripts, tools, names, examples, extensions, option, language options).

The names of some internal macros are shortened to fit into the margin.

Indexing the environments and the files.

Indexing the counters and depth counters.

The example files are in their own directory in the (proposed) TDS-compliant hierarchy.

Indexing referenced commands (begin).

Added mongolb.mld and mongolb.mlo.
Removed mongolb.mlo (new T2 and X2 cyrillic encodings in mongolb.mld).

Added the mtc-3co.tex example file.

Added mongolian.mld (loads mongolb.mld).

Improved the layout of the index.

Indexed the names of authors.

Added the mtc-fko.tex example file.

Bibliographic references for packages and classes in the index.

Features in the index.

Added the “open” and “close” features.

Added the mtc-ocf.tex example file.

Added the \mtcfixnomenclature command.

Added the mtc-nom.tex example file.

Corrected the last argument of \mtcsetfeature and siblings, using \mtc@toks.

Indexing the messages. Messages noted in the right margin.

Added latinc.mld and latinc2.mld for classical latin.

Added internal hyperlinks for messages.

Added \mtcoffset and co. for an horizontal offset of a mini-table.

Added \mtcsetoffset for an horizontal offset of a mini-table type.

Added the mtc-ofs.tex example file.

Added flagging of macros in example files.

Added a clickable table of all messages.

Added a local minitoc in the “Jargon” chapter.

Added lithuanian2.mld.

Added latvian2.mld and letton2.mld.

Added a hint (warning \texttt{W0097}) about the flowfram \cite{433,434} package (incompatible).
Added a hint (I0053) about the float [302], floatrow [285], trivfloat [484], and rotfloat [420] packages.

Regrouping some marginal notes about messages; improving their positions.

Corrected a bug about minitocs in appendices for the \texttt{memoir} class.

Changed the color of hyperlinks.

Revised the format of headers.

Corrected some \texttt{\mtcset...} commands to use \texttt{\edef} to correctly evaluate \texttt{\mtc@toks}.

Added \texttt{swahili.mld}.

Added stuff (files) for maps of areas of some languages.

Added the \texttt{bengali} language synonym of \texttt{bangla}.

Split the list of files into two tables (tables 7.1 to 7.2 on pages 243–244).

Replaced the \texttt{.mtc1} extension by \texttt{.mtc0} in the auto-configuration test (to avoid erasing \texttt{\jobname.mtc1} file).

Corrected a problem with \texttt{\nofiles} (Andreas Deininger).

The acknowledgements are moved to the “Complements” part.

Added a hint (warning \texttt{W0099}) about the \texttt{titlesec} [46] package.

Complete indexing of the messages.

Updated \texttt{lithuanian2.mld}.

Using the \texttt{chngpage} package [467] to make increase the width of the pages of the bibliography.

Renamed \texttt{\if@longextensions@} as \texttt{\if@mtc@longext@}.

Updated \texttt{czech.mld}.

Removed \texttt{\l@xsection}.

Graphic files are indexed separately.

Added a specific directory for image files in the TDS hierarchy.

Updated \texttt{galician.mld}, \texttt{lsorbian.mld}, \texttt{ukraineb.mld}, and \texttt{usorbian.mld}.

Added \texttt{malayalam-b.mld}, \texttt{malayalam-keli2.mld}, \texttt{malayalam-mr.mld}, and \texttt{malayalam-rachana3.mld}. 
Updated malayalam-omega.mlo.

Suppressed parasite entries from the index.

Added occitan.mld.

Updated croatian.mld, danish.mld, dutch.mld, galician.mld, germanb2.mld, greek.mld, icelandic.mld, interlingua.mld, polish.mld, scottish.mld, and turkish.mld.

8.17 Developments in 2008

Corrected polski.mld.

Added table 6.10 on page 231.

Added many maps about languages and dialects, etc.

Better captions for maps.

Added an entry for the Wikipedia in the jargon.

Split the TDS hierarchy into three tables 7.3 to 7.5 on pages 244–247.

The page numbers in the index are now hyperlinks (thanks to François PÉTIARD).

Colors added in figure 1.1 on page 31.

Corrected an error of message number.

Updated from the babel package version v3.8j of 2008/03/16.

Used \vrefrange to compress ranges of internal cross-references.

Added flags for many countries.

Added a figure about lusophonia.

Added a figure about germanophonics.

Added a figure about hispanophonics.

Added a figure about italophonics.

Added a minitoc in the index to make it easier to consult (not trivial).

Added figures about francophones countries.
Added a figure about swahili-speaking countries.

Added a figure about arabic-speaking countries.

Added a figure about russian-speaking countries.

Added a figure about english-speaking countries.

Added flags \ifinparttoc, \ifinpartlof, \ifinpartlot, \ifinminitoc, \ifinminilof, \ifinminilot, \ifinsecttoc, \ifinsectlof, and \ifinsectlot.

Added example document mtc-vti.tex, section 4.36 on page 148.

Added commands for polymorphic entries: \mtcpolymtoc, \mtcpolymlof, and \mtcpolymlot.

Added a figure about dutch-speaking countries.

Renamed fminitoc.dtx and consorts as minitoc-fr.dtx and consorts.

Added missing flag files (thanks to Morten Högholm).

Replaced many .pdf image files (most of them are flag files) by the original .png file because they were corrupted during the conversion by ImageMagick (xpdf didn’t see the problem but Acrobat Reader refuses to show the file); many thanks to Heiko Oberdiek and Staszek Wawrykiewicz.

Back to standard colors and default hyperref color options.
Part II

Implementation
Contents of the Second Part

9  Commented code of the minitoc package 262
10 Commented code of the mtcoff package 447
11 Commented code of the mtcmess package 464
12 Patch for the memoir class 465
13 Language definition (.mld) and object (.mlo) files 467
Chapter 9

Commented code of the minitoc package

Contents

9.1 Introduction ............................................ 266
9.2 Identification code ..................................... 266
9.3 A file descriptor to write ......................... 266
9.4 Indentation and skip ............................... 267
9.5 Tests and flags ........................................ 268
  9.5.1 Flags for the hints option ................. 269
  9.5.2 Use of section-level mini-lists of floats .... 269
  9.5.3 Presence of some packages and classes .... 269
  9.5.4 Flags for packages dealing with floats .... 270
  9.5.5 Insertion of empty mini-tables .............. 271
  9.5.6 Presence or absence of some sectioning commands .... 271
  9.5.7 Flags to check if some commands are used .... 272
  9.5.8 Check if the document has exactly 2 parts .... 273
9.6 Preparation for the notoccite option .......... 274
9.7 Preparation for the tight and k-tight options ... 274
9.8 Preparation to work with hyperref ............ 274
9.9 Checking the presence of some packages ...... 275
  9.9.1 Check if the sectsty package is loaded, and when .... 275
  9.9.2 Check if the varsect package is loaded, and when .... 275
  9.9.3 Check if the freychap package is loaded, and when .... 275
  9.9.4 Check if the hange caption package is loaded, and when .... 275
  9.9.5 Check if the quotchap package is loaded, and when .... 276
  9.9.6 Check if the romannum package is loaded, and when .... 276
  9.9.7 Check if the sfheaders package is loaded, and when .... 276
  9.9.8 Check if the alphanumeric package is loaded, and when .... 276
  9.9.9 Check if the captionref package is loaded, and when .... 277
  9.9.10 Check if the caption package is loaded, and when .... 277
  9.9.11 Check if the caption2 package is loaded, and when .... 277
  9.9.12 Check if the caption package is loaded, and when .... 277
  9.9.13 Check if the mcaption package is loaded, and when .... 278
  9.9.14 Check if the float package is loaded .... 278

262
<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.15</td>
<td>Check if the floatrow package is loaded</td>
</tr>
<tr>
<td>9.16</td>
<td>Check if the trivial float package is loaded</td>
</tr>
<tr>
<td>9.17</td>
<td>Check if the rolfloat package is loaded</td>
</tr>
<tr>
<td>9.10</td>
<td>Is the memoir class loaded?</td>
</tr>
<tr>
<td>9.11</td>
<td>Testing the emptiness of a file</td>
</tr>
<tr>
<td>9.12</td>
<td>Internal macros to decrement minitoc counters</td>
</tr>
<tr>
<td>9.13</td>
<td>Patching the \part command</td>
</tr>
<tr>
<td>9.14</td>
<td>Adding an entry in the TOC for a starred part</td>
</tr>
<tr>
<td>9.15</td>
<td>Section level macros</td>
</tr>
<tr>
<td>9.16</td>
<td>Corrections for numbering</td>
</tr>
<tr>
<td>9.17</td>
<td>Patching the \section command</td>
</tr>
<tr>
<td>9.18</td>
<td>Adding an entry in the TOC for a starred section</td>
</tr>
<tr>
<td>9.19</td>
<td>Chapter level macros</td>
</tr>
<tr>
<td>9.20</td>
<td>Patching the \chapter command</td>
</tr>
<tr>
<td>9.21</td>
<td>Adding an entry in the TOC for a starred chapter</td>
</tr>
<tr>
<td>9.22</td>
<td>Miscellaneous declarations</td>
</tr>
<tr>
<td>9.23</td>
<td>Autoconfiguration of extensions</td>
</tr>
<tr>
<td>9.24</td>
<td>Detecting obsolete versions of \LaTeX</td>
</tr>
<tr>
<td>9.25</td>
<td>Adding a TOC entry without leaders nor page numbers</td>
</tr>
<tr>
<td>9.26</td>
<td>Default values for the page-number customizations</td>
</tr>
<tr>
<td>9.27</td>
<td>“Features” for the mini-tables</td>
</tr>
<tr>
<td>9.28</td>
<td>Fake tables of contents</td>
</tr>
<tr>
<td>9.29</td>
<td>Depth counters for minilofs and minilot</td>
</tr>
<tr>
<td>9.30</td>
<td>Chapter level commands</td>
</tr>
<tr>
<td>9.31</td>
<td>Starred parts, chapters or sections</td>
</tr>
<tr>
<td>9.32</td>
<td>Font commands for the mini-tables</td>
</tr>
<tr>
<td>9.33</td>
<td>Internal commands to position the mini-table titles</td>
</tr>
<tr>
<td>9.34</td>
<td>The mtc@verse environment</td>
</tr>
<tr>
<td>9.35</td>
<td>The \minitoc, \minilof, and \minilot commands</td>
</tr>
<tr>
<td>9.35.1</td>
<td>The \minitoc command</td>
</tr>
<tr>
<td>9.35.2</td>
<td>The \minilof command</td>
</tr>
<tr>
<td>9.35.3</td>
<td>The \minilot command</td>
</tr>
<tr>
<td>9.36</td>
<td>Patching the \chapter command, continued</td>
</tr>
<tr>
<td>9.37</td>
<td>The addstarred... commands</td>
</tr>
<tr>
<td>9.38</td>
<td>TOC entries without leaders</td>
</tr>
<tr>
<td>9.39</td>
<td>Mini-tables with or without leaders</td>
</tr>
<tr>
<td>9.40</td>
<td>The \dominitoc command and its siblings</td>
</tr>
<tr>
<td>9.40.1</td>
<td>Analysis and splitting of the TOC file</td>
</tr>
<tr>
<td>9.41</td>
<td>Mini-lists of figures</td>
</tr>
<tr>
<td>9.41.1</td>
<td>Analysis and splitting of the list of figures file</td>
</tr>
<tr>
<td>9.42</td>
<td>Mini-lists of tables</td>
</tr>
<tr>
<td>9.42.1</td>
<td>Analysis and splitting of the list of tables file</td>
</tr>
<tr>
<td>9.43</td>
<td>Macro to write a contents line</td>
</tr>
<tr>
<td>9.44</td>
<td>Depth counters for partlofs and partlots</td>
</tr>
<tr>
<td>9.45</td>
<td>Part level commands</td>
</tr>
<tr>
<td>9.46</td>
<td>Fonts for the parttocs</td>
</tr>
<tr>
<td>9.47</td>
<td>Default titles for part-level mini-tables</td>
</tr>
<tr>
<td>9.48</td>
<td>The ptc@verse environment</td>
</tr>
<tr>
<td>9.49</td>
<td>The part level mini-tables: \parttoc, \partlof, and \partlot</td>
</tr>
<tr>
<td>9.49.1</td>
<td>The \parttoc command</td>
</tr>
<tr>
<td>9.49.2</td>
<td>The \partlof command</td>
</tr>
<tr>
<td>9.49.3</td>
<td>The \partlot command</td>
</tr>
<tr>
<td>9.50</td>
<td>Auxiliary commands for printing parttocs</td>
</tr>
</tbody>
</table>
9.51 Patching the `\part` command, continued ...................................... 338
9.52 The `\doparttoc` command and its siblings .................................... 338
  9.52.1 Processing macros for the parttos ........................................ 340
  9.52.2 Processing macros for the partlofs ..................................... 344
  9.52.3 Processing macros for the partlots ..................................... 346
9.53 Depth counters for sectlofs and sectlots ...................................... 349
9.54 Section-level commands ............................................................. 349
9.55 Fonts commands for secttocs and co. .......................................... 350
9.56 Internal macros for title positionning ........................................ 350
9.57 The `stc@verse` environment ...................................................... 351
9.58 The `\secttoc`, `\sectlof`, and `\sectlot` commands .......................... 351
   9.58.1 The `\secttoc` command .................................................. 351
   9.58.2 The `\sectlof` command .................................................. 354
   9.58.3 The `\sectlot` command .................................................. 357
9.59 Auxiliary internal commands, section level .................................... 359
9.60 Patching the `\section` command (continued) ................................ 360
9.61 The `\dosecttoc` command and siblings ........................................ 361
9.62 End of section-level commands ................................................... 371
9.63 The `\mtcprepare` command ....................................................... 371
9.64 Use with `\nofiles` ................................................................... 372
9.65 Necessary `\@...` commands ....................................................... 373
9.66 The horizontal rules and their default values .................................. 374
9.67 The `\mtcset...` commands .......................................................... 375
   9.67.1 Keywords for the `\mtcset...` commands ................................ 375
   9.67.2 The `\mtcsetfont` command ................................................. 377
   9.67.3 The `\mtcsettitlefont` command ......................................... 381
   9.67.4 The `\mtcsettitle` command ............................................... 382
   9.67.5 The `\mtcsetformat` command ............................................. 384
   9.67.6 The `\mtcsetpagenumbers` command .................................... 388
   9.67.7 The `\mtcsetrules` command .............................................. 391
   9.67.8 The `\mtcsetfeature` command ............................................ 393
   9.67.9 The `\mtcsetdepth` command .............................................. 395
   9.67.10 The `\mtcsetoffset` command ............................................ 397
9.68 Polymorphic entries ...................................................................... 398
9.69 The `\mtchideinmaintoc` environment and siblings ............................. 399
9.70 Fixing the "Glossary" entry in the TOC ........................................ 401
9.71 Fixing the "Index" entry in the TOC ............................................. 403
9.72 Fixing the "Nomenclature" entry in the TOC ................................... 404
9.73 The `\mtcselectlanguage` command ............................................. 405
9.74 The `\mtcloadmlo` internal command ............................................ 406
9.75 The "coffee breaks" ................................................................... 407
9.76 Initialization of counters ............................................................. 407
9.77 Declarations for simple options ..................................................... 407
   9.77.1 Options tight and loose, k-tight and k-loose .......................... 407
   9.77.2 Options checkfiles and nocheckfiles .................................... 408
   9.77.3 Options dotted and undotted .............................................. 408
   9.77.4 Option notoccite .............................................................. 408
   9.77.5 Option shortext ............................................................... 408
9.78 The `\mtc@insection` option .......................................................... 408
9.79 The `\listfiles` and `\nolistfiles` options ....................................... 409
9.80 Language options ......................................................................... 409
9.81 The `\mtc@hints@begindoc` option ............................................... 414
   9.81.1 First part: `\mtc@hints@begindoc` ....................................... 414
   9.81.1.1 Hint about the alphanum package ................................... 415
9.81.1.2 Hint about the appendix package .......................... 415
9.81.1.3 Hint about the tocbibind package .......................... 415
9.81.1.4 Hint about the KOMA-Script classes ........................ 415
9.81.1.5 Hint about the tocloft package ............................. 416
9.81.1.6 Hint about the titlessec package ............................ 416
9.81.1.7 Hint about the titletoc package .............................. 417
9.81.1.8 Hint about the placeins package ............................. 417
9.81.1.9 Hint about the memoir class ................................ 418
9.81.1.10 Hint about the amsart and amsproc classes ............... 419
9.81.1.11 Hint about the amsbook class ................................ 419
9.81.1.12 Hint about the abstract package ............................. 419
9.81.1.13 Hint about the jura class .................................. 420
9.81.1.14 Hint about the flowfram package ............................ 420
9.81.1.15 Hint about the alteration of the sectionning commands . . 420
  9.81.1.15.1 Alteration of \part ....................................... 421
  9.81.1.15.2 Alteration of \chapter ................................... 421
  9.81.1.15.3 Alteration of \section ................................... 422
9.81.1.16 Hint about the consistency of the calling sequences of the
  commands ........................................................................ 422
9.81.2 Final part: \mtc@hints@enddoc ................................. 423
  9.81.2.1 Hint about \sect-lof|lot and the insection option ......... 423
  9.81.2.2 Final part of the consistency tests .......................... 424
  9.81.2.3 Check if the main tables have been prepared (first part) . 426
  9.81.2.4 Check if the main tables have been prepared (second part) 428
  9.81.2.5 Check the number of mini-tables, in case of short extensions 430
  9.81.2.6 Final part of the hint about the sectsty package ........ 431
  9.81.2.7 Final part of the hint about the varsects package ......... 431
  9.81.2.8 Final part of the hint about the freychap package ........ 432
  9.81.2.9 Final part of the hint about the hangcaption package ..... 432
  9.81.2.10 Final part of the hint about the quotchap package ....... 432
  9.81.2.11 Final part of the hint about the romannum package ....... 433
  9.81.2.12 Final part of the hint about the sfheaders package ...... 433
  9.81.2.13 Final part of the hint about the alnumsec package ....... 434
  9.81.2.14 Final part of the hint about the captioncont package .... 434
  9.81.2.15 Final part of the hint about the caption package .......... 434
  9.81.2.16 Final part of the hint about the caption2 package ....... 435
  9.81.2.17 Final part of the hint about the caption2 package ....... 435
  9.81.2.18 Final part of the hint about the mcaption package ...... 435
  9.81.2.19 Final part of the hint about the float package ........... 436
  9.81.2.20 Final part of the hint about the floatrow package ...... 436
  9.81.2.21 Final part of the hint about the trivfloat package ...... 436
  9.81.2.22 Final part of the hint about the roff float package ...... 437
  9.81.2.23 Check if empty mini-tables have been detected .......... 437
  9.81.2.24 Check if obsolete commands have been used ............. 439
  9.81.2.25 Check if some hints have been written .................. 440
9.82 Processing of options ............................................. 440
  9.82.1 Processing the insection option ............................... 441
  9.82.2 Processing the notoccite option ............................... 442
  9.82.3 Processing the listfiles option ............................... 442
  9.82.4 Processing the hints option.................................... 443
  9.82.5 Saving the sectionning commands .............................. 443
9.83 Trapping the undefined preparation and insertion commands ...... 444
9.84 The minitoc-fr.dtx file .............................................. 445
9.1 Introduction

This very long chapter presents the code of the \texttt{minitoc} package and attempts to explain it. Some comments of the original source file\footnote{The source file of version \#42. Version \#43 includes the conversion of the package to \texttt{.dtx-\ins} format. Version \#42 has not been distributed because of that.} are skipped, like the history, because they do not need further examination (they will be put in the change history).

The code is split in sections to make the reading easier, and the sections are sometimes reordered to make the reading easier.

Most of the \texttt{minitoc} external commands have \texttt{mtc}, \texttt{ptc}, \texttt{stc}, or one of the mini-table names (\texttt{parttoc\ldots}, \texttt{sectlot}) in their names. Most of the \texttt{minitoc} internal commands have \texttt{@mtc}, \texttt{@ptc}, \texttt{@stc}, or \texttt{parttoc@\ldots}, \texttt{sectlot@} in their names, or a similar convention. The few exceptions should be explicit enough to not conflict with other packages.

9.2 Identification code

The code of \texttt{minitoc.sty} starts here:

\begin{verbatim}
\NeedsTeXFormat{LaTeX2e}[1996/06/02]
\ProvidesPackage{minitoc}[2018/07/12 v62 Package minitoc] % message I0000
\RequirePackage{mtcmess}[2006/03/14]
\mtcPackageInfo[I0001]{minitoc} % *** minitoc package, version 62 ***\@gobble
\end{verbatim}

9.3 A file descriptor to write

\begin{verbatim}
\newwrite\tf@mtc
\end{verbatim}
A file descriptor is needed to write the files containing the mini-tables, it is \texttt{\tf@mtc}. The \texttt{minitoc} package uses only one file descriptor for writing. See section 9.23 on page 286.

\begin{verbatim}
\newwrite\tf@mtc
\end{verbatim}
9.4 Indentation and skip

We define the indentation \mtcindent (both sides) of the mini-tables and the command \mtcskip to make a vertical skip before a mini-table, its value is \mtcskipamount (default: \bigskipamount).

\newlength{\mtcindent}
\newskip{\mtcskipamount}
\setlength{\mtcskipamount}{\bigskipamount}
\def{\mtcskip}{\parskip=\z@ \addvspace{\mtcskipamount}}

Note that \mtcskip uses a local group to avoid the influence of \parskip.

We define the default values for the vertical gaps before and after titles part level mini-tables.

\def{\mtcgapbeforeheads}{50\p@}
\def{\mtcgapafterheads}{40\p@}

We define the vertical kernings between the minitables and their before the bottom rule. The names of these macros is rather explicit. The values are empirical and can be changed via \renewcommand.

\ifdefined{part}{}{%
  \def{\kernafterpartttoc}{\kern-1.\baselineskip\kern.5ex}%
  \def{\kernafterparttlof}{\kern-1.\baselineskip\kern.5ex}%
  \def{\kernafterpartlot}{\kern-1.\baselineskip\kern.5ex}%
}\ifdefined{chapter}{}{%
  \ifdefined{section}{}{%
    \def{\kernaftersectttoctoc}{\kern-1.\baselineskip\kern.5ex}%
    \def{\kernaftersectttoctlof}{\kern-1.\baselineskip\kern.5ex}%
    \def{\kernaftersectttoclot}{\kern-1.\baselineskip\kern.5ex}%
  }%}
%}
}\ifdefined{minitoc}{}{%
  \def{\kernafterminitoc}{\kern-.5\baselineskip\kern0.5ex}%
  \def{\kernafterminitetoc}{\kern-1.\baselineskip\kern0.5ex}%
  \def{\kernafterminitetlot}{\kern-1.\baselineskip\kern0.5ex}%
}%
We defines horizontal offsets by which the margins in the mini-tables are corrected (added to the right margin and subtracted from the left margin). These are commands, not lengths\(^3\), to be redefined by `\renewcommand`.

\begin{verbatim}
\@ifundefined{part}{}{\
  \def\ptcoffset{0pt}\
  \def\plfoffset{0pt}\
}
\@ifundefined{chapter}{}{\
  \def\stcoffset{0pt}\
  \def\slfoffset{0pt}\
}
\@ifundefined{section}{}{\
  \def\mtcoffset{0pt}\
  \def\mlfoffset{0pt}\
}
\ifinparttoc\ifinpartlof\ifinpartlot\ifinminitoc\ifinminilof\ifinminilot\ifinsecttoc\ifinsectlof\else\fi
\newif\ifinparttoc\inparttocfalse\
\newif\ifinpartloid\inpartloffalse\newif\ifinpartlotid\inpartlotfalse\
\newif\ifinminitoc\inminitocfalse\newif\ifinminilof\inminiloffalse\newif\ifinminilot\inminilotfalse\newif\ifinsecttoc\insecttocfalse\
\newif\ifinsectlof\insectloffalse\newif\ifinsectlot\insectlotfalse
\end{verbatim}

\subsection*{9.5 Tests and flags}

We need to declare some flags\(^3\) (via `\newif`) to detect the loading of some packages or classes and the availability of some commands (this will be used by the `\texttt{hints}` option (section 9.81 on page 414) or to allow the definition of some `\texttt{minitoc}` commands).

\footnote{We avoid to allocate a precious length register.}
\footnote{Not so many years ago, some authors had a preference for using counters rather than flags, because a flag costs 3 control sequences (`\iffoo`, `\foofalse` and `\footrue`), which use memory. But the number of count registers is limited to 256 in the native \LaTeX{} engine (much more with \LaTeXe{} \cite{LaTeXe}, but still limited in number), while memory has become rather cheap today. And a code programmed with flags (`\iffoo \else \fi`) is easier to structure and debug than a code programmed with counters, as we.}
9.5.1 Flags for the hints option

But first, we define some flags for the hints option:

- The flag `\if@mtc@hints@` is true if the hints option is required (default).
- The flag `\if@mtc@hints@w@` is set true if we detect that some sectioning commands have been altered since the loading of the document class.
- The flag `\if@mtc@hints@given@` is set true if the hints option detects something curious and writes messages in the `document.log` file. It will be used at the end of the document to signal that you should look for hints in the `document.log` file.

9.5.2 Use of section-level mini-lists of floats

We will check if the commands `\dosectlof` and `\dosectlot` are used:

9.5.3 Presence of some packages and classes

We will check if the `placeins` package is loaded, then if the `memoir` is loaded (and if it is a recent enough version), then if the `sectsty` package is loaded (before or after `minitoc`).

We do the same for some caption-related packages:
9.5.4 Flags for packages dealing with floats

We must warn about a limitation with the float [302], floatrow [285], trivfloat [484], and rotfloat [420] packages.
9.5.5 Insertion of empty mini-tables

We will check if you have attempted to insert some empty mini-tables:
\if@mtc@empty@parttoc@ \if@mtc@empty@partlot@ \if@mtc@empty@minitoc@ \if@mtc@empty@minilof@ \if@mtc@empty@minilot@ \if@mtc@empty@secttoc@ \if@mtc@empty@sectlof@ \if@mtc@empty@sectlot@

9.5.6 Presence or absence of some sectioning commands

We define and set flags about the presence of the sectioning commands (in fact, the counters
associated with these commands).
\if@mtc@part@def@ The part counter:
\if@mtc@chapter@def@ The chapter counter:
\if@mtc@section@def@ The section counter:

We define and set flags about the absence of the sectioning commands.
\if@mtc@part@undef@ The part counter:
9.5.7 Flags to check if some commands are used

We define a pair of flags for each mini-table type: one for the command itself and one for the preparation command (\do...). These flags will be used by the hints package option (section 9.81 on page 414).
We also detect the use of some obsolete commands:

\newif\if@firstpartis\used@ \global\@firstpartis\used@false
\newif\if@firstchapteris\used@ \global\@firstchapteris\used@false
\newif\if@firstsectionis\used@ \global\@firstsectionis\used@false

9.5.8 Check if the document has exactly 2 parts

In French, the ordinal adjective is “deuxième” if the second object is not the last object, but “second” (masculine) or “seconde” (feminine) when it is also the last one (see [251, page 204]). So we define a specific flag:

\newif\ifmtcsecondpart \mtcsecondpartfalse

At the beginning of the document, we test this flag and make it global:
\AtBeginDocument
\ifmtcsecondpart
\AtBeginDocument{\%
\ifmtcsecondpart
\global\mtcsecondparttrue
\else
\global\mtcsecondpartfalse
\fi
\}
\AtEndDocument
\ifmtcsecondpart
\@mainaux
\AtEndDocument{\%
\ifnum\value{part}=2\relax
\mtcsecondparttrue
\else
\mtcsecondpartfalse
\fi
\if@filesw
\ifmtcsecondpart
\immediate\write\@mainaux{\string\global\string\mtcsecondparttrue}%
\else
\immediate\write\@mainaux{\string\global\string\mtcsecondpartfalse}%
\fi
\fi
\}

So we need two \TeX runs to get a correct result. The french2.mld language definition file (see section 13.62 on page 498) uses this trick to form the titles of part level mini-tables. See the mtc-2nd.tex example file in section 4.2 on page 92.
9.6 Preparation for the notoccite option

We declare a flag for the presence of this option and the new internal “hook” command (redefinable command) `\mtc@hook@beforeinputfile`, used by this option (this has been requested by Donald Arseneau for his notoccite package [14]). See section 1.6 on page 52.

\begin{verbatim}
\newif\if@mtc@notoccite@ \@mtc@notoccite@false
\@ifundefined{mtc@hook@beforeinputfile}{\let\mtc@hook@beforeinputfile\relax}{}}
\end{verbatim}

9.7 Preparation for the tight and k-tight options

We just declare a flag for each of these options; they are set false by default (loose and k-loose options):

\begin{verbatim}
\newif\iftightmtc \tightmtcfalse
\newif\ifktightmtc \ktightmtcfalse
\end{verbatim}

9.8 Preparation to work with hyperref

This code prepares the interface with the hyperref package [390]. A flag is defined, then this preparation is performed in an `\AtBeginDocument` block if this package is loaded. This action defines some commands for the hyperref package.

\begin{verbatim}
\AtBeginDocument
\if@mtc@hyper@used@ \@ifpackageloaded{hyperref}{\mtcPackageInfo{minitoc}{\text{compatible with hyperref@gobble}}\newif\if@mtc@hyper@used@ \global@mtc@hyper@used@false \AtBeginDocument{% \@ifpackageloaded{hyperref}{% \global@mtc@hyper@used@true \\def\toclevel@xpart{1000}\% \\def\toclevel@xchapter{1000}\% \\def\toclevel@xsect{1000}\% \\def\toclevel@starpart\toclevel@part \\def\toclevel@starchapter\toclevel@chapter \\def\toclevel@starsection\toclevel@section \\def\toclevel@starsubsection\toclevel@subsection \\def\toclevel@starparagraph\toclevel@paragraph \\def\toclevel@starsubparagraph\toclevel@subparagraph
}\}}%}
\end{verbatim}
9.9 Checking the presence of some packages

9.9.1 Check if the sectsty package is loaded, and when

\AtBeginDocument
\if@mtc@sectstyLoaded@
\if@mtc@sectstyLoaded@a@
\@ifpackageloaded\@mtc@sectstyLoaded@true\%
\AtBeginDocument\@ifpackageloaded\@mtc@sectstyLoaded@true\%
\else\fi\fi\fi

We must test if the sectsty package [319] is loaded before or after minitoc, so we test when minitoc is loaded and also in an \AtBeginDocument block, when all packages have been loaded. See section 9.81.2.6 on page 431.

9.9.2 Check if the varsects package is loaded, and when

\@ifpackageloaded\varsects\%
\@ifpackageloaded\varsects\%
\AtBeginDocument\@ifpackageloaded\varsects\%
\@ifpackageloaded\varsects\%

We must test if the varsects package [437] is loaded before or after minitoc, so we test when minitoc is loaded and also in an \AtBeginDocument block, when all packages have been loaded. See section 9.81.2.7 on page 431.

9.9.3 Check if the fncychap package is loaded, and when

\@ifpackageloaded\fncychap\%
\@ifpackageloaded\fncychap\%
\AtBeginDocument\@ifpackageloaded\fncychap\%
\@ifpackageloaded\fncychap\%

We must test if the fncychap package [301] is loaded before or after minitoc, so we test when minitoc is loaded and also in an \AtBeginDocument block, when all packages have been loaded. See section 9.81.2.8 on page 432.

9.9.4 Check if the hangcaption package is loaded, and when

\@ifpackageloaded\hangcaption\%
\@ifpackageloaded\hangcaption\%
\AtBeginDocument\@ifpackageloaded\hangcaption\%
\@ifpackageloaded\hangcaption\%

We must test if the hangcaption package [250] is loaded before or after minitoc, so we test when minitoc is loaded and also in an \AtBeginDocument block, when all packages have been loaded. See section 9.81.2.9 on page 432.
9.9.5 Check if the quotchap package is loaded, and when

\ifpackageloaded
\AtBeginDocument
\if@mtc@quotchapLoaded@\if@mtc@quotchapLoaded@a@\We must test if the quotchap package [442] is loaded before or after minitoc, so we test when minitoc is loaded and also in an \AtBeginDocument block, when all packages have been loaded. See section 9.81.2.10 on page 433.\% \fi\fi\% 2657 \ifpackageloaded{quotchap}{\@mtc@quotchapLoaded@true}{\%2658 \AtBeginDocument{\ifpackageloaded{quotchap}{\@mtc@quotchapLoaded@a@true}{\%} 2659 \ifpackageloaded{romannum}{\@mtc@romannumLoaded@true}{\%} 2660 \AtBeginDocument{\ifpackageloaded{romannum}{\@mtc@romannumLoaded@a@true}{\%}} 2661 \ifpackageloaded{sfheaders}{\@mtc@sfheadersLoaded@true}{\%} 2662 \AtBeginDocument{\ifpackageloaded{sfheaders}{\@mtc@sfheadersLoaded@a@true}{\%}} 2663 \ifpackageloaded{alnumsec}{\@mtc@alnumsecLoaded@true}{\%} 2664 \AtBeginDocument{\ifpackageloaded{alnumsec}{\@mtc@alnumsecLoaded@a@true}{\%}}
9.9.9 Check if the \texttt{captcont} package is loaded, and when

\begin{verbatim}
\@ifpackageloaded{captcont}{\@mtc@captcontLoaded@true}{}
\AtBeginDocument{%\@ifpackageloaded{captcont}{\@mtc@captcontLoaded@a@true}{}}
\end{verbatim}

We must test if the \texttt{captcont} package \cite{131} is loaded before or after \texttt{minitoc}, so we test when \texttt{minitoc} is loaded and also in an \texttt{AtBeginDocument} block, when all packages have been loaded. See section 9.81.2.14 on page 434.

9.9.10 Check if the \texttt{caption} package is loaded, and when

\begin{verbatim}
\@ifpackageloaded{caption}{\@mtc@captionLoaded@true}{}
\AtBeginDocument{%\@ifpackageloaded{caption}{\@mtc@captionLoaded@a@true}{}}
\end{verbatim}

We must test if the \texttt{caption} package \cite{421, 422, 424} is loaded before or after \texttt{minitoc}, so we test when \texttt{minitoc} is loaded and also in an \texttt{AtBeginDocument} block, when all packages have been loaded. See section 9.81.2.15 on page 434.

9.9.11 Check if the \texttt{caption2} package is loaded, and when

\begin{verbatim}
\@ifpackageloaded{caption2}{\@mtc@captionIILoaded@true}{}
\AtBeginDocument{%\@ifpackageloaded{caption2}{\@mtc@captionIILoaded@a@true}{}}
\end{verbatim}

We must test if the \texttt{caption2} package \cite{423} is loaded before or after \texttt{minitoc}, so we test when \texttt{minitoc} is loaded and also in an \texttt{AtBeginDocument} block, when all packages have been loaded. See section 9.81.2.16 on page 435.

9.9.12 Check if the \texttt{ccaption} package is loaded, and when

\begin{verbatim}
\@ifpackageloaded{ccaption}{\@mtc@ccaptionLoaded@true}{}
\AtBeginDocument{%\@ifpackageloaded{ccaption}{\@mtc@ccaptionLoaded@a@true}{}}
\end{verbatim}

We must test if the \texttt{ccaption} package \cite{474} is loaded before or after \texttt{minitoc}, so we test when \texttt{minitoc} is loaded and also in an \texttt{AtBeginDocument} block, when all packages have been loaded. See section 9.81.2.17 on page 435.
9.9.13 Check if the \texttt{mcaption} package is loaded, and when

\begin{verbatim}
\@ifpackageloaded 
\AtBeginDocument 
\if@mtc@mcaptionLoaded@ 
\if@mtc@mcaptionLoaded@a@ 
\end{verbatim}

We must test if the \texttt{mcaption} package is loaded before or after \texttt{minitoc}, so we test when \texttt{minitoc} is loaded and also in an \texttt{\AtBeginDocument} block, when all packages have been loaded. See section 9.81.2.18 on page 435.

\begin{verbatim}
2673 \@ifpackageloaded{mcaption}{\@mtc@mcaptionLoaded@true}{}% 
2674 \AtBeginDocument{\@ifpackageloaded{mcaption}{\@mtc@mcaptionLoaded@true}{}}% 
\end{verbatim}

9.9.14 Check if the \texttt{float} package is loaded

\begin{verbatim}
\@ifpackageloaded 
\AtBeginDocument 
\if@mtc@floatLoaded@ 
\end{verbatim}

We must test if the \texttt{float} package is loaded in the preamble, so we use an \texttt{\AtBeginDocument} block. See section 9.81.2.19 on page 436.

\begin{verbatim}
2675 \AtBeginDocument{\@ifpackageloaded{float}{\@mtc@floatLoaded@true}{}}% 
\end{verbatim}

9.9.15 Check if the \texttt{floatrow} package is loaded

\begin{verbatim}
\@ifpackageloaded 
\AtBeginDocument 
\if@mtc@floatrowLoaded@ 
\end{verbatim}

We must test if the \texttt{floatrow} package is loaded in the preamble, so we use an \texttt{\AtBeginDocument} block. See section 9.81.2.20 on page 436.

\begin{verbatim}
2676 \AtBeginDocument{\@ifpackageloaded{floatrow}{\@mtc@floatrowLoaded@true}{}}% 
\end{verbatim}

9.9.16 Check if the \texttt{trivfloat} package is loaded

\begin{verbatim}
\@ifpackageloaded 
\AtBeginDocument 
\if@mtc@trivfloatLoaded@ 
\end{verbatim}

We must test if the \texttt{trivfloat} package is loaded in the preamble, so we use an \texttt{\AtBeginDocument} block. See section 9.81.2.21 on page 436.

\begin{verbatim}
2677 \AtBeginDocument{\@ifpackageloaded{trivfloat}{\@mtc@trivfloatLoaded@true}{}}% 
\end{verbatim}

9.9.17 Check if the \texttt{rotfloat} package is loaded

\begin{verbatim}
\@ifpackageloaded 
\AtBeginDocument 
\if@mtc@rotfloatLoaded@ 
\end{verbatim}

We must test if the \texttt{rotfloat} package is loaded in the preamble, so we use an \texttt{\AtBeginDocument} block. See section 9.81.2.22 on page 437.

\begin{verbatim}
2678 \AtBeginDocument{\@ifpackageloaded{rotfloat}{\@mtc@rotfloatLoaded@true}{}}% 
\end{verbatim}
9.10 Is the memoir class loaded?

We test if the memoir [479, 481, 482] class is loaded. This class needs some compatibility adjustments or may be incompatible if too recent. In the later case, a patch is inserted (see chapter 12 on page 465). This correction is no more necessary after the 2005/09/25 version of memoir.cls.

```
\@ifclassloaded
\if@mtc@memoirLoaded@
  \if@mtc@memoirnew@
    \if@mtcpatchmemoir@
      We test if the memoir [479, 481, 482] class is loaded. This class needs some compatibility adjustments or may be incompatible if too recent. In the later case, a patch is inserted (see chapter 12 on page 465). This correction is no more necessary after the 2005/09/25 version of memoir.cls.
    \else
      \MessageBreak
      \if@mtcpatchmemoir@
        And now the patch:
        \IfFileExists{mtcpatchmem.sty}{\@ifclasslater{memoir}{2005/09/25}{}{\RequirePackage{mtcpatchmem}}}{\mtcPackageError{minitoc}{Unable to patch the memoir class}{So it remains incompatible. Sorry.}}
        \fi
      \fi
    \fi
  \fi
\fi
```

9.11 Testing the emptiness of a file

Some macros for testing if an argument of a macro is empty (taken from the package ifmtarg [483], by Peter R. Wilson and Donald Arseneau, and from while.tip, by
COMMENTED CODE OF THE MINITOC PACKAGE

Stephan P. von Bechtolsheim [460, Vol III, page 408]). The group is necessary to keep local the catcode change of “Q”, hence a \gdef is needed for \mtc@ifmtarg.

\begin{verbatim}
\begingroup
\catcode\'Q=3
\long\gdef\mtc@ifmtarg#1{\mtc@@ifmtarg#1QQ\@secondoftwo\@firstoftwo\@nil}
\long\gdef\mtc@@ifmtarg#1#2Q#3#4#5\@nil{#4}
\endgroup
\let\mtc@EndWhile = \fi
\def\mtc@While #1#2#3\mtc@EndWhile{\def\mtc@WhilePreCondition{#1}\def\mtc@WhileCondition{#2}\def\mtc@WhileBody{#3}\mtc@@While}
\def\mtc@@While{\mtc@WhilePreCondition\mtc@WhileCondition\let\mtc@WhileNext=\mtc@WhileBody\mtc@@While}
\else\def\mtc@WhileNext{}\fi\mtc@WhileNext
\end{verbatim}

Some macros to test if a file is empty or not: \mtc@CkFile{file} returns \@mtc@FEtrue if the file is empty, \@mtc@FEfalse if the is file not empty. An inexistent file is empty. A file full of white space (space, tabulation, newline) is empty. Comments are empty.

Note: on a big empty file, the \mtc@While loop may be time consuming, but not an eternity (33 s for 10^6 lines on my computer), and the first non-empty line stops the loop. \jobname.mtc is used as scratch file. Its contents is erased after use.

\ifmtc@checkfiles\ifmtc@FE\ifmtc@LI\mtc@While\mtc@Body\mtc@EndWhile\mtc@CkFile\mtc@CkStr\mtc@Rline\tf@mtc\@inputcheck
\newif\ifmtc@OLI\@mtc@OLITrue
\newif\ifmtc@FE\@mtc@FEtrue
\newif\ifmtc@checkfiles\@mtc@checkfilestrue
\def\mtc@Body{\immediate\read\@inputcheck to \mtc@Rline\relax
\ifeof\@inputcheck\relax\@mtc@LIFalse\fi\expandafter\ifx\mtc@Rline\par\relax\def\mtc@Rline{}\else\ifeof\@inputcheck\relax\global\@mtc@LIFalse\fi\mtc@ifmtarg{\mtc@Rline}{\relax}{\@mtc@FEfalse\@mtc@LIFalse}\fi}
\def\mtc@CkFile#1{\@mtc@LItrue\@mtc@FEtrue\if@mtc@checkfiles
\newif\ifmtc@OLI\@mtc@OLITrue\newif\ifmtc@FE\@mtc@FEtrue\newif\ifmtc@checkfiles\@mtc@checkfilestrue
\def\mtc@Body{\immediate\read\@inputcheck to \mtc@Rline\relax\ifeof\@inputcheck\relax\@mtc@OLIfalse\fi\expandafter\ifx\mtc@Rline\par\relax\def\mtc@Rline{}\else\ifeof\@inputcheck\relax\global\@mtc@OLIfalse\fi\mtc@ifmtarg{\mtc@Rline}{\relax}{\@mtc@FEfalse\@mtc@OLIfalse}\fi\fi}
\def\mtc@checkfiles{\@mtc@LItrue\@mtc@FEtrue\if@mtc@checkfiles
9.12 Internal macros to decrement minitoc counters

\texttt{\mtc@onebackpart} \texttt{\mtc@onebackchapter} \texttt{\mtc@onebacksection} \texttt{\addtocounter}

It is sometimes necessary to decrement a minitoc counter (ptc, mtc or stc) by one. These macros are:

\begin{verbatim}
\def\mtc@onebackpart{\addtocounter{ptc}{-1}}
\def\mtc@onebackchap{\addtocounter{mtc}{-1}}
\def\mtc@onebacksect{\addtocounter{stc}{-1}}
\end{verbatim}

9.13 Patching the \texttt{\part} command

\texttt{\part} \texttt{\mtc@svspart} \texttt{\mtc@svpart} \texttt{\@part} \texttt{\stepcounter}

If the \texttt{\part} command is not defined (by the document class, usually), we cannot patch it and a warning is displayed \footnote{Document classes with sectioning commands but no \texttt{\part} command are likely non standard, hence the warning displayed on the terminal.}. Else, we patch its two branches, \texttt{\@part} (for the unstarred version) or \texttt{\@spart} (for the starred version, \texttt{\part*}): we add \texttt{\stepcounter{ptc}} to increment the parttoc counter ptc. See also section 9.51 on page 338.

\begin{verbatim}
\@ifundefined{part}{% \mtcPackageWarningNoLine[W0018]{minitoc}{part level macros NOT available}}{\mtcPackageInfo[I0023]{minitoc}{part level macros available\@gobble}}
\let\mtc@svspart\@spart
\end{verbatim}

The code of the next section (section 9.14 on the following page) is also skipped if \texttt{\part} is not defined.

\begin{verbatim}
\@ifundefined{part}{% \mtcPackageWarningNoLine[W0018]{minitoc}{part level macros NOT available}}{\mtcPackageInfo[I0023]{minitoc}{part level macros available\@gobble}}
\let\mtc@svspart\@spart
\end{verbatim}
9.14 Adding an entry in the TOC for a starred part

To add an entry in the TOC for a starred part, we need the \mtcaddpart macro, which has an optional argument, the title of the part as it should appear in the TOC.

By default, this argument is empty. If it is empty (tested via \mtc@ifmtarg) or omitted, we add a \contentsline{xpart}{}... line in the .toc file. If it is not empty, we add a \contentsline{part}{title...}... line in the .toc file. We always add a \contentsline{xpart}{}... line in the .lof and .lot files. Then we increment the ptc counter, via \adjustptc (defined in section 9.45 on page 325). Using xpart as first argument of \contentsline means that \l@xpart will be invoked in place of \l@part to print the entry in the TOC, but \l@xpart uses a huge depth (10000) for this entry, hence it will never be really printed (except if you cheat).

This code terminates (temporarily) the part level commands.

9.15 Section level macros

The section level macros are defined if \chapter is not defined and \section defined, i.e., in document classes like article, but not in document classes like book or report. So we test if \chapter is defined and if \section is defined, with adequate warnings. If neither are defined, you are in big trouble to use the minitoc package with the class of your document.
9.16 Corrections for numbering

As the TOC, the LOF and the LOT are considered as (starred) sections, we must decrement the secttoc counter (stc) via \mtc@onebacksect when the corresponding commands are executed. Hence we patch these commands.

\let\mtcsv@tableofcontents\tableofcontents
\let\mtcsv@listoffigures\listoffigures
\let\mtcsv@listoftables\listoftables
\def\tableofcontents{\mtcsv@tableofcontents\mtc@onebacksect}
\def\listoffigures{\mtcsv@listoffigures\mtc@onebacksect}
\def\listoftables{\mtcsv@listoftables\mtc@onebacksect}

9.17 Patching the \section command

If the \section command is not defined (by the document class, usually), we cannot patch it and a warning is displayed. Else, we patch its two branches, \@sect (for the unstarred version) or \@ssect (for the starred version, \section*): we add \stepcounter{stc} to increment the secttoc counter stc, only in the unstarred case.

\let\mtc@svsection\section
\def\section{\stepcounter{stc}\mtc@svsection}
\let\mtc@svss\@ssect
\def\@sect{\stepcounter{stc}\mtc@svss\@ssect}

9.18 Adding an entry in the TOC for a starred section

To add an entry in the TOC for a starred section, we need the \mtcaddsection macro, which has an optional argument, the title of the section as it should appear in the TOC. By default, this argument is empty. If it is empty (tested via \mtc@iffmtarg) or omitted, we add a \contentsline{xsect}{...} line in the .toc file. If it is not empty, we add \contentsline{section}{title...} line in the .toc file. We always add a \contentsline{xsect}{...} line in the .lof and .lot files. Then we increment the stc counter, via \adjuststc (this command is defined in section 9.54 on page 349). Using xsect as first argument of \contentsline means that \@xsect will be invoked in place.

\footnote{Version #25 has removed a spurious decrementation of this counter.}
of \@section to print the entry in the TOC, but \@xsect uses a huge depth (10000) for this entry, hence it will never be really printed (except if you cheat).

\newcommand{\mtcaddsection}{% \mtc@ifmtarg{#1}{\addcontentsline{toc}{xsect}{}}{% \addcontentsline{toc}{section}{#1}}% \addcontentsline{lof}{xsect}{} \addcontentsline{lot}{xsect}{% \adjuststc}

This code terminates (temporarily) the section level commands, and we continue with chapter level macros.

}}%}{

9.19 Chapter level macros

The chapter level macros are defined if \chapter is defined, i.e., in document classes like book or report. So we test if \chapter is defined, with adequate warnings. The test is already done above, we are in the “else” branch of @ifundefined{chapter}.

\mtcPackageInfo{minitoc}{chapter level macros available@gobble}

9.20 Patching the \chapter command

The \chapter command is defined (by the document class, usually). We patch its two branches, \@chapter (for the unstarred version) or \@schapter (for the starred version, \chapter*): we add call to \stepcounter{mtc} to increment the minitoc counter mtc. Only the unstarred branch (\@chapter) is patched here. The other branch is patched later (section 9.36 on page 308).

\let\mtc@svchapter@chapter \def\@chapter{\stepcounter{mtc}\mtc@svchapter}

9.21 Adding an entry in the TOC for a starred chapter

To add an entry in the TOC for a starred chapter, we need the \mtcaddchapter macro, which has an optional argument, the title of the chapter as if should appear in the TOC. By default, this argument is empty. If it is empty (tested via \mtc@ifmtarg) or omitted, we add a \contentsline{xchapter}{... line in the .toc file. If it is not empty, we
add a `\contentsline{chapter}{title...}` line in the `.toc` file. We always add a `\contentsline{xchapter}{...} line in the `.lof` and `.lot` files. Then we increment the mtc counter, via `\adjustmtc` (defined in section 9.31 on page 295). Using `xchapter` as first argument of `\contentsline` means that `\l@xchapter` will be invoked in place of `\l@chapter` to print the entry in the TOC, but `\l@xchapter` uses a huge depth (10000) for this entry, hence it will never be really printed (except if you cheat).

```latex
\newcommand{\mtcaddchapter}[1][1]{% 
  \mtc@ifmtarg{#1}{\addcontentsline{toc}{xchapter}{}}% 
  {\addcontentsline{toc}{chapter}{#1}}% 
  \addcontentsline{lof}{xchapter}{}% 
  \addcontentsline{lot}{xchapter}{}% 
  \adjustmtc}% 
}\mtcaddchapter
```

This code terminates (temporarily) the chapter level commands, i.e., terminates the `\ifundefined{chapter}` at the beginning of section 9.15 on page 282.

9.22 Miscellaneous declarations

\newread \newtoks \mtc@toks \mtc@string \mtc@strut \mtc@strutbox \mtc@hstrutbox \mtc@zrule \mtc@BBR

The `\newread` command must be redeclared as being `\outer` (as Donald Arseneau told me). We need a token register (`\mtc@toks`), a temporary string (`\mtc@string`), struts (two kinds, each one using a box containing an invisible vertical rule), a rule with all dimensions equal to zero (`\mtc@zrule`) and a command discouraging page breaks (`\mtc@BBR`, for "bad break").

For the struts, which are boxes containing an invisible vertical rule, we use "ex" units, to follow the current font.

```latex
\def\newread{\alloc@6\read\chardef\sixt@@n}
\def\mtc@strut{\relax\ifmmode\copy\mtc@strutbox\else\unhcopy\mtc@strutbox\fi}
\def\mtc@hstrut{\relax\ifmmode\copy\mtc@hstrutbox\else\unhcopy\mtc@hstrutbox\fi}
```

```latex
\def\mtc@string{\relax}
\def\mtc@zrule{\leavevmode\mtc@strut}
\def\mtc@BBR{\unpenalty\nopagebreak[4]}
```
### 9.23 Autoconfiguration of extensions

This code is a trick to determine if the operating system is able or unable to use long extensions (> 3 characters) in file names. We define a file descriptor ($\texttt{tf@mtc}$) to write files\(^6\). This code is verbose if long extensions cannot be used, else the messages are only written in the `document.log` file. The sequencing of these operations is vital. The table 9.1 shows this sequence. A star (\*) denotes which file is read in phase 3.

\begin{verbatim}
\newif\if@mtc@longext\@mtc@longext@false
\immediate\openout\@mtc@longext@true
\immediate\write\@mtc@longext@true
\immediate\closeout\@mtc@longext@true
\immediate\openout\@mtc@longext@false
\immediate\write\@mtc@longext@false
\immediate\closeout\@mtc@longext@false
\input{\jobname.mtc}
\end{verbatim}

\(*$\backslash$*)

\begin{table}[h]
\centering
\begin{tabular}{|l|c|c|c|}
\hline
Phase (time runs from left to right): & 1 & 2 & 3 \\
\hline
OS with long extensions & $\jobname.mtc$ & TRUE & TRUE & \* \\
& $\jobname.mtc$ & & FALSE & \\
\hline
OS with short extensions & $\jobname.mtc(0)$ & TRUE & FALSE & \* \ \\
\hline
\end{tabular}
\caption{Trick to detect the limitation to short extensions}
\end{table}

---

\[^6\text{It is the only new file descriptor created by the minitoc package. All files written by minitoc use this descriptor, or one of the standard descriptors, e.g., for the \texttt{document.log} file. In fact, minitoc writes also in the .toc, .lof and .lot files, but via file descriptors already used by standard commands like }\backslash\texttt{tableofcontents}, \backslash\texttt{listoffigures} \text{ and } \backslash\texttt{listoftables}. \text{We can conclude that minitoc itself uses only one file descriptor (or write stream). Some other attempts to make per chapter TOCs have failed by quickly leading to exhaustion of file descriptors (\LaTeX only 16 file descriptors for writing), because they called the standard internal \backslash\texttt{starttoc} macro, which invokes \backslash\texttt{newwrite}, for each mini-table. As minitoc writes into only one file at a time (and in the \texttt{document.log} file, and in the standard contents files, of course), we can reuse the same file descriptor and avoid this serious problem (which was present in the original version of the package). The minitoc package writes in the contents files when it encounters a major sectioning command (\backslash\texttt{part}, \backslash\texttt{chapter}, or \backslash\texttt{section}), if necessary. It writes into the mini-table auxiliary files only via the mini-table preparing commands (\backslash\texttt{doparttoc}, ..., \backslash\texttt{dosectlot}), once at a time. You do not need a new hammer for each nail.}\]
Commented code of the \texttt{minitoc} package

The text and the severity of the messages are different.

\begin{verbatim}
\jobname.mtc. The text and the severity of the messages are different.
2842 \if@mtc@longext@
2843 \mtcPackageInfo[I0012]{minitoc}\
2844 {Long extensions (Unix-like) will be used}@gobble}
2845 \mtcPackageInfo[I0031]{minitoc}\
2846 {==> this version is configured for UNIX-like
2847 \MessageBreak
2848 \space\space\space\space(long extensions) file names}@gobble%
2849 \else
2850 \mtcPackageWarningNoLine[W0019]{minitoc}\
2851 {Short extensions (MSDOS-like) will be used
2852 \MessageBreak
2853 ==> this version is configured for MSDOS-like
2854 \MessageBreak
2855 \space\space\space\space(8+3) file names}
2856 \fi

(5) We erase the contents of the two files (because \jobname.mtc is also used later as a scratch file, see section 9.11 on page 279).
2857 \immediate\openout\tf@mtc \jobname.mtc
2858 \immediate\closeout\tf@mtc
2859 \immediate\openout\tf@mtc \jobname.mtc0
2860 \immediate\closeout\tf@mtc

\section{Detecting obsolete versions of \LaTeX}

\texttt{@inputcheck} \texttt{\reset@font} This code detects old versions of the \LaTeX kernel that are no more supported and with which the \texttt{minitoc} package can hardly work. The trick is to detect the absence of some internal \LaTeX commands, \texttt{@inputcheck} and \texttt{\reset@font}. If you get one of these messages, you are in bad luck and should \textit{urgently} update your \LaTeX installation, which is just rusting since... a lot of years!

\begin{verbatim}
\@ifundefined{@inputcheck}\
{\mtcPackageWarningNoLine[W0021]{minitoc}\
 {Your version of latex.tex is obsolete.
2864 \MessageBreak
2865 Trying to continue..\}\newread{@inputcheck}{relax}{}\}
\@ifundefined{reset@font}\
{\mtcPackageWarningNoLine[W0022]{minitoc}\
 {Your version of latex.tex is very obsolete.
2869 \MessageBreak
2870 Trying to continue... crossing fingers}%
2871 \let\reset@font\{relax}{}
\end{verbatim}
9.25 Adding a TOC entry without leaders nor page numbers

\@undottedtocline The (internal) macro \@undottedtocline is a modified version of the standard command \@dottedtocline. It will be used in customization macros.

\newif\ifundottedmtc
\undottedmtcfalse

\ifnum #1\s@tocdepth\relax \else \vskip \z@ plus.2\p@ \fi}

9.26 Default values for the page-number customizations

\if@mtc@memoirLoaded@ This section defines some customization macros for the presence or absence of page numbers in the mini-tables. But if the memoir class \cite{memoir} is loaded, it does the job. So, we test first \if@mtc@memoirLoaded@ to use the commands of memoir when they are available.

\mtcpagenumbers For entries in minitocs:
\nomtcpagenumbers

\def\mtcpagenumbers{%
  \cftpagenumberson{section}
  \cftpagenumberson{subsection}
  \cftpagenumberson{subsubsection}
  \cftpagenumberson{paragraph}
  \cftpagenumberson{subparagraph}}
\def\nomtcpagenumbers{%
  \cftpagenumbersoff{section}
  \cftpagenumbersoff{subsection}
  \cftpagenumbersoff{subsubsection}
  \cftpagenumbersoff{paragraph}
  \cftpagenumbersoff{subparagraph}}
\stcpagenumbers For entries in secttocs:
\nostcpagenumbers
\def\stcpagenumbers{%
  \cftpagenumberson{subsection}
  \cftpagenumberson{subsubsection}
  \cftpagenumberson{paragraph}
  \cftpagenumberson{subparagraph}}
\def\nostcpagenumbers{%
  \cftpagenumbersoff{subsection}
  \cftpagenumbersoff{subsubsection}
  \cftpagenumbersoff{paragraph}
  \cftpagenumbersoff{subparagraph}}

\ptcpagenumbers For entries in parttocs:
\noptcpagenumbers
\def\ptcpagenumbers{%
  \cftpagenumberson{chapter}
  \cftpagenumberson{section}
  \cftpagenumberson{subsection}
  \cftpagenumberson{subsubsection}
  \cftpagenumberson{paragraph}
  \cftpagenumberson{subparagraph}}
\def\noptcpagenumbers{%
  \cftpagenumbersoff{chapter}
  \cftpagenumbersoff{section}
  \cftpagenumbersoff{subsection}
  \cftpagenumbersoff{subsubsection}
  \cftpagenumbersoff{paragraph}
  \cftpagenumbersoff{subparagraph}}

\mlfpagenumbers For entries in minilofs, sectlofs, and partlofs:
\nomlfpagenumbers
\mlfpagenumbers
\slfpagenumbers
\oslfpagenumbers
\plfpagenumbers
\oplfpagenumbers
\def\mlfpagenumbers{%
  \cftpagenumberson{figure}}
\def\nomlfpagenumbers{%
  \cftpagenumbersoff{figure}}
\def\slfpagenumbers{%
  \cftpagenumberson{figure}}
\def\oslfpagenumbers{%
  \cftpagenumbersoff{figure}}
\def\plfpagenumbers{%
  \cftpagenumberson{figure}}
\def\oplfpagenumbers{%
  \cftpagenumbersoff{figure}}

\mltpagenumbers For entries in minilots, sectlots, and partlots:
\nomltpagenumbers
\mltpagenumbers
\sltpagenumbers
\osltpagenumbers
\pltpagenumbers
\opltpagenumbers
\def\mltpagenumbers{%
  \cftpagenumberson{table}}
\def\nomltpagenumbers{%
  \cftpagenumbersoff{table}}
\def\sltpagenumbers{%
  \cftpagenumberson{table}}
\def\osltpagenumbers{%
  \cftpagenumbersoff{table}}
\def\pltpagenumbers{%
  \cftpagenumberson{table}}
\def\opltpagenumbers{%
  \cftpagenumbersoff{table}}
Else, minitoc will use its own commands.

\else

\mtcpagenumbers \nomtcpagenumbers \mlfpagenumbers \nomlfpagenumbers \mltpagenumbers \nomltpagenumbers
First, for minitocs, secttocs and parttocs:
\def\mtcpagenumbers{\let\mtc@pgno\null}
\def\nomtcpagenumbers{\let\mtc@pgno\relax}
\def\stcpagenumbers{\let\stc@pgno\null}
\def\nostcpagenumbers{\let\stc@pgno\relax}
\def\ptcpagenumbers{\let\ptc@pgno\null}
\def\noptcpagenumbers{\let\ptc@pgno\relax}
\fi

Then, for minilofs, sectlofs and partlofs:
\def\mlfpagenumbers{\let\mlf@pgno\null}
\def\nomlfpagenumbers{\let\mlf@pgno\relax}
\def\slfpagenumbers{\let\slf@pgno\null}
\def\oslfpagenumbers{\let\slf@pgno\relax}
\def\plfpagenumbers{\let\plf@pgno\null}
\def\oplfpagenumbers{\let\plf@pgno\relax}

Then, for minilots, sectlots and partlots:
\def\mltpagenumbers{\let\mlt@pgno\null}
\def\nomltpagenumbers{\let\mlt@pgno\relax}
\def\sltpagenumbers{\let\slt@pgno\null}
\def\osltpagenumbers{\let\slt@pgno\relax}
\def\pltpagenumbers{\let\plt@pgno\null}
\def\opltpagenumbers{\let\plt@pgno\relax}
\fi

Then the default values are set; page numbers are present:
9.27 “Features” for the mini-tables

Each kind of mini-table has five “features”: a “before” feature, an “after” feature, an “open” feature, an “close” feature, and a “pagestyle” feature.

A “before” feature is defined by a macro like \beforeparttoc which contains code to be executed before any mini-table of a given type: \beforeparttoc is executed before each parttoc. Usually such features contain only trivial commands like \clear[double]page, or \empty.

An “after” feature is analog but its code is executed after each mini-table of a given type.

An “open” feature contains code to be executed just before the insertion of the file containing the mini-table. Usually such features either do nothing, either prepare some basic formatting (like multi-column). It does not concern the title of the mini-table or the decorative rules.

An “close” feature contains code to be executed just after the insertion of the file containing the mini-table. Usually such features either do nothing, either finish some basic formatting (like multi-column). It does not concern the title of the mini-table or the decorative rules.

A “pagestyle” feature is defined by a macro like \thispageparttocstyle which contains code to define the page style implied by mini-tables of a given type: the command \thispageparttocstyle can be defined as \thispagestyle{...}. Usually, the “pagestyle” feature is only defined for part-level mini-tables, which use page breaks in their before and after features. For chapter- and section-level mini-tables, the “pagestyle” feature is usually defined as \empty.

We set the default values for the part-level features depending on the presence of the \chapter command, as article-like documents are different from the book- or report-like documents for the layout of part-level mini-tables.

\chapter \beforeparttoc \beforepartlof \beforepartlot
If \chapter is not defined, the part level mini-tables have no “before” feature (by default):

\cleardoublepage But if \chapter is defined, they have a \cleardoublepage as default “before” feature:
Chapter level mini-tables have no "before" feature (by default):
\beforeminitoc \empty
\beforeminilof \empty
\beforeminilot \empty

Section level mini-tables have no "before" feature (by default):
\beforesecttoc \empty
\beforesectlof \empty
\beforesectlot \empty

If \chapter is not defined, the part level mini-tables have no "after" feature (by default):
\chapter
\afterparttoc \empty
\afterpartlof \empty
\afterpartlot \empty

But if \chapter is defined, they have a \cleardoublepage as default "after" feature:
\cleardoublepage
\afterparttoc \cleardoublepage
\afterpartlof \cleardoublepage
\afterpartlot \cleardoublepage

Chapter level mini-tables have no "after" feature (by default):
\afterminitoc \empty
\afterminilof \empty
\afterminilot \empty

Section level mini-tables have no "after" feature (by default):
\aftersecttoc \empty
\aftersectlof \empty
\aftersectlot \empty
By default, the "open" features do nothing:
\let\openparttoc\empty
\let\openpartlof\empty
\let\openminitoc\empty
\let\openminilof\empty
\let\openminilot\empty
\let\opensecttoc\empty
\let\opensectlof\empty
\let\opensectlot\empty

By default, "close" features do nothing:
\let\closeparttoc\empty
\let\closepartlof\empty
\let\closeminitoc\empty
\let\closeminilof\empty
\let\closeminilot\empty
\let\closesecttoc\empty
\let\closesectlof\empty
\let\closesectlot\empty

By default, all the "pagemode" features (at part level) use the empty page style. It affects only the first page of the mini-table. If \chapter is not defined, there is no default "pagemode" features at the part level.
\ifundefined{chapter}{%  
\def\thispageparttocstyle{\empty}
\def\thispagepartlofstyle{\empty}
\def\thispagepartlotstyle{\empty}
\def\thispageminitocstyle{\empty}
\def\thispageminilofstyle{\empty}
\def\thispageminilotstyle{\empty}
\def\thispagesecttocstyle{\empty}
\def\thispagesectlofstyle{\empty}
\def\thispagesectlotstyle{\empty}
\}

In section 9.67.8 on page 393, we will define the \mtcsetfeature macro which is a much easier user interface to set the mini-tables "features".
9.28 Fake tables of contents

If you don’t want a table of contents, but want minitocs, you need to create the .toc file, without inserting it into your document. This \faketableofcontents command is a stripped off version of the standard command \tableofcontents. We define in the same way the analogous commands \fakelistoffigures and \fakelistoftables, using in fact just a stripped version \fake@starttoc of \@starttoc. But it is nice to reset to zero the ptc, mtc, and stc counters now, if they are defined.\footnote{Remember the infamous “stc0” bug.}

\def\faketableofcontents{\fake@starttoc{toc}\
%\if@filesw \expandafter\newwrite\csname tf@#1\endcsname\
%\immediate\openout \csname tf@#1\endcsname \jobname.#1\relax \fi
%\global\@nobreakfalse \endgroup}
\def\fakelistoffigures{\fake@starttoc{lof}}
\def\fakelistoftables{\fake@starttoc{lot}}
\def\fake@starttoc#1{\begingroup \makeatletter
%\if@filesw \expandafter\newwrite\csname tf@#1\endcsname
%\immediate\openout \csname tf@#1\endcsname \jobname.#1\relax \fi
%\global\@nobreakfalse \endgroup}

This code uses the same file descriptors (for writing) than the original commands.

9.29 Depth counters for minilofs and minilots

If the counters lofdepth and lotdepth are defined, we create the necessary new counters: minilofdepth and minilotdepth. These counters are initialized to 2. This is done after the loading of the packages, in an \AtBeginDocument block:

\AtBeginDocument{\%\ifdefined{c@lofdepth}\%\setcounter{minilofdepth}{2}\%\fi}\%\ifdefined{c@lotdepth}\%\setcounter{minilotdepth}{2}\%\fi}\%}

9.30 Chapter level commands

From here, we define the chapter-level commands.
First, we memorize the marks (not used today, but...):

\global\let\mtc@markboth\markboth
\global\let\@mkboth\markboth

\addcontentsline
\stepcounter
\c@ptc
\c@mtc
\c@stc

We define commands to manage the starred sectioning commands: \part*, \chapter* and \section*. The section-level is different depending on the presence of the \chapter command. Eventually, a counter is incremented. A contents line is added in the .toc file, with the right depth to print it (see \l@star... later, in section 9.65 on page 373).

\def\addstarredsection#1{
  \addst@rred{section}{#1}
}

If \chapter is not defined, we just define \addstarredsection:

\addstarredsection
\chapter
\addst@rred

\def\addstarredsection#1{\addst@rred{section}{#1}}

\def\addstarredsection#1{\addst@rred{section}{#1}}
Else we begin to define the stuff for chapter-level commands (the “else” branch of \@ifundefined{chapter}):

\begin{verbatim}
3065 \%
\begin{verbatim}
\The@mtc
\firstchapteris
\if@firstchapteris@used@
\newcounter
\setcounter
\adjustmtc
\decrementmtc
\incrementmtc
\themtc
\columnwidth
We define now: the internal format of the mtc counter (\The@mtc), the obsolete command \firstchapteris (it just emits a harmless warning), the mtc counter (initialized to 0), the \adjustmtc command (increments the mtc counter by 1 by default), the \decrementmtc command (decrements the mtc counter by 1), the \incrementmtc command (increments the mtc counter by 1), the format of the mtc counter (\themtc), the counter minitocdepth, initialized to 2, for the depth of a minitoc (analog to the standard tocdepth counter).
\end{verbatim}
\end{verbatim}

\begin{verbatim}
3066 \def\The@mtc{\arabic{mtc}}
3067 \def\firstchapteris#1\{
3068 \mtcPackageWarning[W0003]{minitoc}{\string\firstchapteris \space is an obsolete (ignored)
3069 \MessageBreak
3070 \@firstchapteris@used@true}
3071 \newcounter{mtc}
3072 \setcounter{mtc}{0}
3073 \newcommand{\adjustmtc}[1]{\addtocounter{mtc}{#1}}
3074 \def\decrementmtc{\addtocounter{mtc}{-1}}
3075 \def\incrementmtc{\addtocounter{mtc}{+1}}
3076 \gdef\themtc{\arabic{mtc}}
3077 \newcounter{minitocdepth}
3078 \setcounter{minitocdepth}{2}
\end{verbatim}

\begin{verbatim}
3079 \mtc@rule
3080 \mlf@rule
3081 \mlt@rule
3082 \plf@rule
3083 \plt@rule
3084 \slf@rule
3085 \slt@rule
3086 \mtcindent=24\p@
\end{verbatim}

We define the horizontal rules to draw before and after minitocs (\mtc@rule), and we copy that definition into analog macros for other kinds of mini-tables. We also set the default value (24pt) of \mtcindent, the indentation for minitocs (both sides). The rules are 0.4pt thick. They are defined via \hrule to stay in vertical mode for the final \kern.
\end{verbatim}

\begin{verbatim}
3087 \def\mtc@rule{\kern-3\p@ \hrule \@width\columnwidth \kern2.6\p@}
3088 \let\mlf@rule=\mtc@rule
3089 \let\mlt@rule=\mtc@rule
3090 \let\plf@rule=\mtc@rule
3091 \let\plt@rule=\mtc@rule
3092 \let\slf@rule=\mtc@rule
3093 \let\slt@rule=\mtc@rule
3094 \mtcindent=24\p@
\end{verbatim}
9.32 Font commands for the mini-tables

We define these commands with full NFSS [291] descriptions. These definitions are effective if \chapter is defined. The fonts for titles are also defined here. See also the \mtcsetfont macro (section 9.67.2 on page 377) and the \mtcsettitlefont macro later (section 9.67.3 on page 381).

\mtcfont \mtcSfont \mtcSSfont \mtcSSSfont \mtcPfont \mtcSPfont \mlffont \mlfSfont \mltfont \mltSfont \mtifont

3089 \def\mtcfont{\small\rmfamily\upshape\mdseries}
3090 \def\mtcSfont{\small\rmfamily\upshape\bfseries}
3091 \let\mtcSSfont\mtcfont \let\mtcSSSfont\mtcfont
3092 \let\mtcPfont\mtcfont \let\mtcSPfont\mtcfont
3093 \let\mlffont\mtcfont \let\mlfSfont\mtcfont
3094 \let\mltfont\mtcfont \let\mltSfont\mtcfont
3095 \def\mtifont{\large\rmfamily\upshape\bfseries}
3096 \def\coffeefont{\small\rmfamily\slshape\mdseries}

9.33 Internal commands to position the mini-table titles

\df@mtitc \df@mtilf \df@mtilt \do@mtitc \do@mtilf \do@mtilt

The commands \miniXXX and \dominiXXX accept an optional argument to left justify, center, right justify or omit the title of the chapter-level mini-tables. By default, these titles are left justified. The choice made in a \dominiXXX command is global and memorized in \df@mtitc, \df@mtilf or \df@mtilt; the choice made in a \miniXXX command is local and stored in \do@mtitc, \do@mtilf or \do@mtilt. See the \minitoc@ macro later (section 9.35.1 on the following page). An empty title needs a vertical correction (Frank Mittelbach).

\c@mti \l@mti \r@mti \e@mti \n@mti

Centering, flushleft, flushright or empty titles:
3097 \def\c@mti#1{\null\hfill #1\hfill\null}
3098 \def\l@mti#1{\null #1\hfill\null}
3099 \def\r@mti#1{\null\hfill #1\null}
3100 \def\e@mti#1{\vspace{-\baselineskip}}
3101 \def\n@mti#1{\vspace{-\baselineskip}}

3102 \let\do@mtitc\l@mti
3103 \let\df@mtitc\l@mti
3104 \let\do@mtilf\l@mti
3105 \let\df@mtilf\l@mti
3106 \let\do@mtilt\l@mti
3107 \let\df@mtilt\l@mti

Default: titles on left:
Each minitoc is placed inside a \texttt{mtc@verse} environment. This environment is analog to the standard \texttt{verse} environment and hence defined via two commands: \texttt{mtc@verse} and \texttt{endmtc@verse}. As it is a list environment, we first define (in a local way) \texttt{\list{}} and set some dimensions like \texttt{\itemsep}, \texttt{\itemindent}, \texttt{\listparindent}, \texttt{\topsep}. \texttt{\parsep} is set to zero if the \texttt{tight} option is active (to reduce the spacing of the lines). \texttt{\parskip} is set to zero if the \texttt{k-tight} option is active (to reduce the spacing of the lines). Both margins are set to \texttt{\mtcindent}. \texttt{\endmtc@verse} terminates the list and discourages a page break. The \texttt{mtc@verse} environment has an argument which is an horizontal offset (a command like \texttt{\mtoffset}).

9.35 The \texttt{minitoc}, \texttt{minilof}, and \texttt{minilot} commands

These three commands are very similar, with only cosmetic differences.

9.35.1 The \texttt{minitoc} command

The \texttt{minitoc} command must be used after \texttt{\chapter} if you need a minitoc (no automatic minitoc).

This command accepts an optional argument, whose default value has eventually been set earlier by a \texttt{\dominitoc} command. The letter "d" represents this default value. \texttt{\dominitoc} has itself an optional argument which sets the default value of the optional argument of \texttt{\minitoc}. 
The default value of the optional argument of the \dominitoc command is “1”. It seems tortuous, but it is simple to use: we have a default behaviour (1) which can be altered globally via the optional argument of \dominitoc, or locally via the optional argument of \minitoc.

\minitoc \minitoc@ \@ifnextchar So we define \minitoc with an optional argument and its (current) default value, and call the true code in the \minitoc@ macro (which has one delimited argument); we use the \@ifnextchar trick to detect a left bracket for the optional argument:

\def\minitoc{\@ifnextchar[#1]{\minitoc@}{\minitoc@[d]}}

The real code of \minitoc is in \minitoc@, which has a mandatory argument (delimited by brackets) specifying the position of the title.

First, we set the global flag \@minitoc@used@true to note that \minitoc has been called (this will be used by a hint later, section 9.81.2.2 on page 424).

\def\minitoc@[#1]{% \global\@minitoc@used@true

The name of the file containing the minitoc is constructed from \jobname and a suffix \@tocfile, which is .mtc (long extensions) or .M (short extensions) followed by the absolute number of the minitoc.

\def\@tocfile{mtc\The@mtc}% \else \def\@tocfile{M\The@mtc}% \fi

Then we test (via \mtc@CkFile) the emptiness of this file. A warning is given if the file is empty and a flag is set (a hint will signal that an empty minitoc has been requested). We call \thispageminitocstyle to set the page style (by default, this does nothing because, by default, there is no page break before a minitoc). The marks are not treated, because usually there is no new page for a minitoc.
We call \beforeminitoc, then begin a samepage environment (to try to discourage page breaks in a minitoc) and look at the position of the title. If the title is empty, the layout is corrected. We print the title with its font (\mtifont), then the top rule of the minitoc (if rules are present), using a tabular environment (to inhibit a page break between the title and the top rule). The font is set to \mtcfont.

\begin{tabular}{@{}p{\columnwidth}@{}}
\reset@font\mtifont\do@mtitc{\mtc@v\mtctitle}\
\end{tabular}

We forbid a page break after the title and the top rule, then set some layout parameters and begin an mtc@verse environment:

\begin{mtc@verse}{\mtcoffset}
We force the effective depth of the mini-table ($\texttt{\c@tocdepth}$) to the required depth ($\texttt{\c@minitocdepth}$), so the printing is done inside the $\texttt{mtc@verse}$ environment, where $\texttt{\c@tocdepth}$ has been forced to $\texttt{\c@minitocdepth}$, to print only the entries whose level is low enough, then inhibit a page break. The blank line is necessary to avoid a parasite negative indentation.

\begin{verbatim}
\c@tocdepth=\c@minitocdepth
\leavevmode\mtc@BBR
\c@pgno\@dottedtocline\@undottedtocline\mtc@hook@beforeinputfile\mtc@setform\openminitoc\ifinminitoc\closeminitoc\mtcsetformat\mtc@strut\global\@nobreakfalse\endgroup
\end{verbatim}

We test the presence of leaders and page numbers, then print the minitoc by inputing the minitoc file. But before reading the minitoc file, we must call the hook macro (asked for by Donald Arseneau for his notoccite package [14]) $\texttt{\mtc@hook@beforeinputfile}$ and the macro $\texttt{\mtc@setform}$ which adjusts some layout parameters (defined by the user via some $\texttt{\mtcsetformat}$ commands). We work in a group to keep local some macro redifinitions. The "open" and "close" features are called just before and after the insertion of the mini-table file.

\begin{verbatim}
\begingroup
\makeatletter
\ifx\mtc@pgno\relax\relax\else\vskip -2.5ex\rule[2.4\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}\fi
\end{verbatim}

We close the $\texttt{mtc@verse}$ environment, add the bottomrule (while preventing a page break), then close the $\texttt{samepage}$ environment, and call $\texttt{\afterminitoc}$. The blank line (\texttt{\textbackslash{}}) is essential.

\begin{verbatim}
\end{verbatim}

And we define the bottom rule for a minitoc, with some space under the minitoc:

\begin{verbatim}
\def\mtc@bottom@rule{% 
\texttt{\ifx\mtc@rule\relax\relax\else\vskip -2.5ex\rule[2.4\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}\fi}
\end{verbatim}
9.35.2 The \minilof command

The \minilof command is very similar to the \minitoc command.

The \minilof command must be used after \chapter if you need a minilof (no automatic minilof).

This command accepts an optional argument, whose default value has eventually been set earlier by a \dominilof command. The letter "d" represents this default value. \dominilof has itself an optional argument which sets the default value of the optional argument of \minilof. The default value of the optional argument of the \dominilof command is “l”. It seems tortuous, but it is simple to use: we have a default behaviour (l) which can be altered globally via the optional argument of \dominilof, or locally via the optional argument of \minilof.

So we define \minilof with an optional argument and its (current) default value, and call the true code in the \minilof@ macro (which has one delimited argument); we use the \@ifnextchar trick to detect a left bracket for the optional argument:

\def\minilof{\@ifnextchar[{\minilof@}{\minilof@[d]}}

The real code of \minilof is in \minilof@, which has a mandatory argument (delimited by brackets) specifying the position of the title.

First, we set the global flag \@minilof@used@true to note that \minilof has been called (this will be used by a hint later, section 9.81.2.2 on page 424).

The name of the file containing the minilof is constructed from \jobname and a suffix \@tocfile, which is .mlf (long extensions) or .F (short extensions) followed by the absolute number of the minilof.
Then we test (via \texttt{mtc@CkFile}) the emptiness of this file. A warning is given if the file is empty and a flag is set (a hint will signal that an empty minilof has been requested).

\begin{verbatim}
\texttt{mtc@CkFile} \texttt{\jobname.\@tocfile}
\if@mtc@FE
\mtcPackageInfo[Hallo06]{minitoc}
% \jobname.@tocfile\space is empty
\@mtc@empty@minilof\true
\else
\thispageminilofstyle
\fi
\end{verbatim}

We call \texttt{\thispageminilofstyle} to set the page style (by default, this does nothing because, by default, there is no page break before a minilof). The marks are not treated, because usually there is no new page for a minilof.

\begin{verbatim}
\beforeminilof
\samepage
\do@mtitc
\e@mti
\n@mti
\c@mti
\l@mti
\r@mti
\df@mtic
\mtc@CkStr
\mtctitle
\if@mtc@FE
\mlffont
\mtifont
\mlf@rule
\columnwidth
\tabular
\\beforeminilof
\relax\begin{samepage}\
\if #1e\let\do@mtilf\e@mti
\else\if #1n\let\do@mtilf\n@mti
\else\if #1c\let\do@mtilf\c@mti
\else\if #1l\let\do@mtilf\l@mti
\else\if #1r\let\do@mtilf\r@mti
\else\if #1d\let\do@mtilf\df@mtilf
\fi\fi\fi\fi\fi\fi
\mtc@CkStr{\mlftitle}\if\mtc@FE \let\do@mtilf\e@mti,relax\fi
\raggedright
\par
\noindent
\if\mlf@rule
\begin{tabular}{@{}p{\columnwidth}@{}}
\reset@font\mtifont\do@mtilf{\mtc@v\mlftitle}\
\end{tabular}\
\else
\begin{tabular}{@{}p{\columnwidth}@{}}
\reset@font\mtifont\do@mtilf{\mtc@v\mlftitle}\
\hline
\end{tabular}\
\fi
\end{samepage}\
\reset@font\mlffont
\parindent=\z@\noindent\noindent\kern-0.8\baselineskip\nopagebreak[4]\%
\par\noindent
\if\mlf@rule\relax
\begin{tabular}{p{\columnwidth}@{}}
\reset@font\mtifont\do@mtilf{\mtc@v\mlftitle}\
\end{tabular}\
\else
\begin{tabular}{p{\columnwidth}@{}}
\reset@font\mtifont\do@mtilf{\mtc@v\mlftitle}\
\hline
\end{tabular}\
\fi
We forbid a page break after the title and the top rule, then set some layout parameters and begin an\texttt{mtc@verse} environment:

\begin{verbatim}
\nopagebreak[4]\null\leavevmode\mtc@zrule\\mtc@BBR
\leftmargin\mtcindent \rightmargin\mtcindent
\itemindent=\z@\labelwidth=\z@%
\labelsep=\z@\listparindent=\z@%
\begin{mtc@verse}{\mlfoffset}\c@lofdepth
\c@minilofdepth\\\mtc@BBR
\c@lofdepth \c@minilofdepth
\end{mtc@verse}\mtc@bottom@rule
\samepage
\afterminilof
\end{verbatim}

We force the effective depth of the mini-table (\texttt{\c@tocdepth}) to the required depth (\texttt{\c@minilofdepth}), so the printing is done inside the \texttt{mtc@verse} environment, where \texttt{tocdepth} has been forced to \texttt{minilofdepth}, to print only the entries whose level is low enough, then inhibit a page break. The blank line is necessary to avoid a parasite negative indentation.

\begin{verbatim}
\@ifundefined{c@lofdepth}{}\c@lofdepth=\c@minilofdepth
\ifnum\c@lofdepth<1\relax\c@lofdepth=1\fi\leavevmode\mtc@BBR
\vskip -.5\baselineskip
\mtc@pgno\@dottedtocline\@undottedtocline\mtc@hook@beforeinputfile\mlf@setform
\ifinminilof\openminilof\closeminilof\mtcsetformat\mtc@strut\beginingroup
\ifundef{mlf@pgno}{\let@dottedtocline\@undottedtocline}{}
\@fileswfalse\mtc@hook@beforeinputfile\mlf@setform\global\openminilof\inminiloftrue
\@input{\jobname.@tocfile}\global\inminiloffalse\closeminilof
\vspace{-\baselineskip} \leavevmode\mtc@strut
\endgroup
\end{verbatim}

We test the presence of leaders and page numbers, then print the minilof by inputing the minilof file. But before reading the minilof file, we must call the hook macro (asked for by Donald Arsenau for his \texttt{notoccite} package [14]) \texttt{mtc@hook@beforeinputfile}, and the macro \texttt{mlf@setform} which adjusts some layout parameters (defined by the user via some \texttt{mtcsetformat} commands). We work in a group to keep local some macro redefinitions. The “open” and “close” features are called just before and after the insertion of the mini-table file.

\begin{verbatim}
\begingroup
\makeatletter
\@ifundefined{mlf@pgno}{\let\dottdottedtocline\undottdottedtocline}{}, {}
\@fileswfalse\mtc@hook@beforeinputfile\mlf@setform\global\openminilof\inminiloftrue\mtc@setformat
\begin{verbatim}
\global\closeminilof\mtc@strut\begin{verbatim}
\global\@nobreakfalse\endgroup
\end{verbatim}
\endgroup
\end{verbatim}

We close the \texttt{mtc@verse} environment, add the bottomrule (while preventing a page break), then close the \texttt{samepage} environment, and call \texttt{\afterminilof}. The blank line (\texttt{\null}) is essential.

\begin{verbatim}
\end{mtc@verse}\
\end{verbatim}
9.35.3 The \minilot command

\minilot The \minilot command is absolutely similar to the \minilof command:

\minilot \chapter The \minilot command must be used after \chapter if you need a minilot (no automatic minilot).

\minilot This command accepts an optional argument, whose default value has eventually been set earlier by a \dominilot command. The letter "d" represents this default value. \dominilot has itself an optional argument which sets the default value of the optional argument of \minilot. The default value of the optional argument of the \dominilot command is “1”. It seems tortuous, but it is simple to use: we have a default behaviour (1) which can be altered globally via the optional argument of \minilot, or locally via the optional argument of \minilot.

\minilot So we define \minilot with an optional argument and its (current) default value, and call the true code in the \minilot@ macro (which has one delimited argument); we use the \@ifnextchar trick to detect a left bracket for the optional argument:

\def\minilot{\@ifnextchar[\minilot@}{\minilot@[d]}}

The real code of \minilot is in \minilot@, which has a mandatory argument (delimited by brackets) specifying the position of the title.

\ifminilotused@ First, we set the global flag \@minilotused@true to note that \minilot has been called (this will be used by a hint later, section 9.81.2.2 on page 424).

\def\minilot@[#1]{%}
\global\@minilotused@true

\@tocfile \ifmtclongext@ The name of the file containing the minilot is constructed from \jobname and a suffix \@tocfile, which is .mlt (long extensions) or .T (short extensions) followed by the absolute number of the minilot.

\ifmtclongext@%
\def\@tocfile{mlt\The@mtc}%
Then we test (via `\mtc@CkFile`) the emptiness of this file. A warning is given if the file is empty and a flag is set (a hint will signal that an empty minilot has been requested).

```latex
\mtc@CkFile
\if\mtc@FE
\if\mtc@empty@minilot
Then we test (via `\mtc@CkFile`) the emptiness of this file. A warning is given if the file is empty and a flag is set (a hint will signal that an empty minilot has been requested).

\mtc@CkFile{\jobname.\@tocfile}
\if\mtc@FE
\mtcPackageInfo{\jobname.\@tocfile\space is empty}
\@mtc@empty@minilot@true
\else
We call `\thispageminilotstyle` to set the page style (by default, this does nothing because, by default, there is no page break before a minilot). The marks are not treated, because usually there is no new page for a minilot.

\thispageminilotstyle
\beforeminilot
\relax\begin{samepage}%
\if #1e\let\do@mtilt\e@mti
\else\if #1n\let\do@mtilt\n@mti
\else\if #1c\let\do@mtilt\c@mti
\else\if #1l\let\do@mtilt\l@mti
\else\if #1r\let\do@mtilt\r@mti
\else\if #1d\let\do@mtilt\df@mtic
\fi\fi\fi\fi\fi\fi
\mtc@CkStr{\mlttitle}\if\mtc@FE \let\do@mtilt\e@mti\relax\fi
\raggedright
\parskip=\z@%
\reset@font\mltfont%
\parindent=\z@%
\nopagebreak[4]%
\kern-0.8\baselineskip\nopagebreak[4]%
```

We call `\beforeminilot`, then begin a `\samepage` environment (to try to discourage page breaks in a minilot) and look at the position of the title. If the title is empty, the layout is corrected. We print the title with its font (`\mtifont`), then the top rule of the minilot (if rules are present), using a `\tabular` environment (to inhibit a page break between the title and the top rule). The font is set to `\mltfont`.

We call `\beforeminilot`, then begin a `\samepage` environment (to try to discourage page breaks in a minilot) and look at the position of the title. If the title is empty, the layout is corrected. We print the title with its font (`\mtifont`), then the top rule of the minilot (if rules are present), using a `\tabular` environment (to inhibit a page break between the title and the top rule). The font is set to `\mltfont`.

```latex
\beforeminilot
\relax\begin{samepage}%
\if #1e\let\do@mtilt\e@mti
\else\if #1n\let\do@mtilt\n@mti
\else\if #1c\let\do@mtilt\c@mti
\else\if #1l\let\do@mtilt\l@mti
\else\if #1r\let\do@mtilt\r@mti
\else\if #1d\let\do@mtilt\df@mtilt
\fi\fi\fi\fi\fi\fi
\mtc@CkStr{\mlttitle}\if\mtc@FE \let\do@mtilt\e@mti\relax\fi
\raggedright
\parskip=\z@%
\reset@font\mltfont%
\parindent=\z@%
\nopagebreak[4]%
\kern-0.8\baselineskip\nopagebreak[4]%
```
We forbid a page break after the title and the top rule, then set some layout parameters and begin an \texttt{mtc@verse} environment:

\begin{tabular}{@{}p{\columnwidth}@{}}
\reset@font\mtifont\do@mtilt{\mtc@v\mlttitle}\\
\hline
\end{tabular}

\begin{itemize}
\item \texttt{nopagebreak[4]}\null \leavevmode \mtc@zrule
\item \leftmargin=\mtcindent \rightmargin=\mtcindent
\item \itemindent=\z@ \labelwidth=\z@ \labelsep=\z@ \listparindent=\z@
\item begin(\texttt{mtc@verse})\{\texttt{mltoffset}\}
\end{itemize}

We force the effective depth of the mini-table (\texttt{c@lotdepth}) to the required depth (\texttt{c@minilotdepth}), so the printing is done inside the \texttt{mtc@verse} environment, where \texttt{lotdepth} has been forced to \texttt{minilotdepth}, to print only the entries whose level is low enough, then inhibit a page break. The blank line is necessary to avoid a parasite negative indentation.

\begin{verbatim}
\@ifundefined{c@lotdepth}{}{\c@lotdepth=\c@minilotdepth}
\ifnum\c@lotdepth<1\relax\c@lotdepth=1\fi
\leavevmode
\mtc@BBR
\vskip -.5\baselineskip
\end{verbatim}

We test the presence of leaders and page numbers, then print the minilot by inputing the minilot file. But before reading the minilot file, we must call the hook macro (asked for by Donald Arseneau for his \texttt{notoccite} package \cite{notoccite}) \texttt{mtc@hook@beforeinputfile} and the macro \texttt{mlt@setform} which adjusts some layout parameters (defined by the user via some \texttt{mtcsetformat} commands). We work in a group to keep local some macro redefinitions. The “open” and “close” features are called just before and after the insertion of the mini-table file.

\begin{verbatim}
\begingroup
\makeatletter
\@ifundefined{mlt@pgno}{\let\@dottedtocline\@undottedtocline}{}
\@fileswfalse
\mtc@hook@beforeinputfile
\mtcsetformat
\mlt@setform
\ifinminilot
\openminilot
\closeminilot
\mtc@strut
\mtc@zrule
\mtc@BBR
\texttt{\begin{itemize}}
\item \texttt{begingroup}
\item \texttt{\makeatletter}
\item \texttt{\@ifundefined{mlt@pgno}{\let\@dottedtocline\@undottedtocline}{}}
\item \texttt{\filesfalse\mtc@hook@beforeinputfile}
\item \texttt{\mlt@setform}
\item \texttt{\global\openminilot\imminilottrue}
\item \texttt{\input{\jobname..tocfile}}
\item \texttt{\global\imminilotfalse\closeminilot}
\item \texttt{\vspace{-.lex} \vspace{-.baselineskip}}
\item \texttt{\leavevmode\mtc@strut}
\end{verbatim}

We test the presence of leaders and page numbers, then print the minilot by inputing the minilot file. But before reading the minilot file, we must call the hook macro (asked for by Donald Arseneau for his \texttt{notoccite} package \cite{notoccite}) \texttt{mtc@hook@beforeinputfile} and the macro \texttt{mlt@setform} which adjusts some layout parameters (defined by the user via some \texttt{mtcsetformat} commands). We work in a group to keep local some macro redefinitions. The “open” and “close” features are called just before and after the insertion of the mini-table file.

\begin{verbatim}
\begingroup
\makeatletter
\@ifundefined{mlt@pgno}{\let\@dottedtocline\@undottedtocline}{}
\@fileswfalse
\mtc@hook@beforeinputfile
\mtcsetformat
\mlt@setform
\ifinminilot
\openminilot
\closeminilot
\mtc@strut
\mtc@zrule
\mtc@BBR
\texttt{\begin{itemize}}
\item \texttt{begingroup}
\item \texttt{\makeatletter}
\item \texttt{\@ifundefined{mlt@pgno}{\let\@dottedtocline\@undottedtocline}{}}
\item \texttt{\filesfalse\mtc@hook@beforeinputfile}
\item \texttt{\mlt@setform}
\item \texttt{\global\openminilot\imminilottrue}
\item \texttt{\input{\jobname..tocfile}}
\item \texttt{\global\imminilotfalse\closeminilot}
\item \texttt{\vspace{-.lex} \vspace{-.baselineskip}}
\item \texttt{\leavevmode\mtc@strut}
\end{verbatim}

We test the presence of leaders and page numbers, then print the minilot by inputing the minilot file. But before reading the minilot file, we must call the hook macro (asked for by Donald Arseneau for his \texttt{notoccite} package \cite{notoccite}) \texttt{mtc@hook@beforeinputfile} and the macro \texttt{mlt@setform} which adjusts some layout parameters (defined by the user via some \texttt{mtcsetformat} commands). We work in a group to keep local some macro redefinitions. The “open” and “close” features are called just before and after the insertion of the mini-table file.

\begin{verbatim}
\begingroup
\makeatletter
\@ifundefined{mlt@pgno}{\let\@dottedtocline\@undottedtocline}{}
\@fileswfalse
\mtc@hook@beforeinputfile
\mtcsetformat
\mlt@setform
\ifinminilot
\openminilot
\closeminilot
\mtc@strut
\mtc@zrule
\mtc@BBR
\texttt{\begin{itemize}}
\item \texttt{begingroup}
\item \texttt{\makeatletter}
\item \texttt{\@ifundefined{mlt@pgno}{\let\@dottedtocline\@undottedtocline}{}}
\item \texttt{\filesfalse\mtc@hook@beforeinputfile}
\item \texttt{\mlt@setform}
\item \texttt{\global\openminilot\imminilottrue}
\item \texttt{\input{\jobname..tocfile}}
\item \texttt{\global\imminilotfalse\closeminilot}
\item \texttt{\vspace{-.lex} \vspace{-.baselineskip}}
\item \texttt{\leavevmode\mtc@strut}
\end{verbatim}
We close the mtc@verse environment, add the bottomrule (while preventing a page break), then close the samepage environment, and call \afterminilot. The blank line (\hspace{1em}) is essential.

\end{mtc@verse}
\kernafterminilot
\nopagebreak[4]\mlt@rule\null\leavevmode\%
\vskip-1.0\baselineskip\mtc@zrule\end{samepage}%
\par\pagebreak[1]\vspace*{-1ex}\afterminilot\fi}

9.36 Patching the \chapter command, continued

First, we define \l@xchapter which is like \l@chapter, but with a huge depth, to inhibit its printing (except if you cheat):
\def\l@xchapter{@dottedtocline{\@M}{1em}{2.3em}}
\def\xchapter{xchapter}
\sv@chapter
\addcontentsline{lof}{xchapter}{#1}\
\addcontentsline{lot}{xchapter}{#1}\
\ignorespaces

\let\mtc@schapter\@schapter
\def\@schapter{\addtocontents{toc}{\protect\chapterend}}
\let\chapterbegin\relax
\let\chapterend\relax
\pagebreak[1]
\vspace*{-1ex}
\afterminilot\fi}

We also patch \@schapter (the starred branch of \chapter) to add marks in the TOC to delimit chapters; these marks will be used by the \dominiXXX commands to take slices from the LOF and the LOT, as they are defined as \relax, they should not perturbate other packages.
9.37 The \addstarred... commands

If the command \chapter is undefined, we define the command \addstarredsection (only if \section is defined). If the command \chapter is defined, we define the command \addstarredchapter. If the command \part is defined, we define the command \addstarredpart. We use the utility command \addst@rred defined in section 9.31 on page 295.

\addstarredsection
\addstarredchapter
\addstarredpart
\chapter
\section
\part
\addst@rred

9.38 TOC entries without leaders

\@Undottedtocline We define two internal macros to format TOC entries without leaders. The macro \@Undottedtocline prints no page number, but \@Undottedtoclinep prints it.

\@Undottedtocline
\coffeefont

\@Undottedtoclinep The same but with the page number:

\@Undottedtoclinep
9.39 Mini-tables with or without leaders

This code sets the flag to false, then patches each mini-table command (its internal part). We alter the commands \minitoc, \minilof, etc., to test the flag \ifundottedmtc and, if true, replace locally \dottedtocline by its dotless version \Undottedtocline. Of course, we must also test the availability of the \chapter, \part and \section commands, to avoid to define many unnecessary commands.

For the part level:
\sv@parttoc\sv@partlof\sv@partlot
\ift\sv@parttoc\parttoc\sv@partlof\sv@partlot
\let\sv@parttoc@\parttoc@\sv@partlof@\sv@partlot@
\def\parttoc@[#1]{\ifundottedmtc\let\dottedtocline\Undottedtocline\fi\sv@parttoc@[#1]}%
\let\sv@partlof@\partlof@\sv@partlot@
\def\partlof@[#1]{\ifundottedmtc\let\dottedtocline\Undottedtocline\fi\sv@partlof@[#1]}%
\let\sv@partlot@\partlot@\sv@partlot@
\def\partlot@[#1]{\ifundottedmtc\let\dottedtocline\Undottedtocline\fi\sv@partlot@[#1]}

For the section level:
\sv@secttoc\sv@sectlof\sv@sectlot
\ift\sv@secttoc\secttoc\sv@sectlof\sv@sectlot@
\def\secttoc@[#1]{\ifundottedmtc\let\dottedtocline\Undottedtocline\fi\sv@secttoc@[#1]}%
\let\sv@sectlof@\sectlof@\sv@sectlot@
\def\sectlof@[#1]{\ifundottedmtc\let\dottedtocline\Undottedtocline\fi\sv@sectlof@[#1]}%
\let\sv@sectlot@\sectlot@\sv@sectlot@
\def\sectlot@[#1]{\ifundottedmtc\let\dottedtocline\Undottedtocline\fi\sv@sectlot@[#1]}

\minitoc\minilof\minilot
\ift\minitoc\\minilof\\minilot@\dottedtocline\Undottedtocline
\let\sv@minitoc@\minitoc@\sv@minilof@\sv@minilot@
\def\minitoc@[#1]{\ifundottedmtc\let\dottedtocline\Undottedtocline\fi\sv@minitoc@[#1]}%
\def\minilof@[#1]{\ifundottedmtc\let\dottedtocline\Undottedtocline\fi\sv@minilof@[#1]}%
\def\minilot@[#1]{\ifundottedmtc\let\dottedtocline\Undottedtocline\fi\sv@minilot@[#1]}

For the chapter level:
\sv@chaptoc\sv@chapelof\sv@chaplot
\ift\sv@chaptoc\chaptoc\sv@chapelof\sv@chaplot@
\def\chaptoc@[#1]{\ifundottedmtc\let\dottedtocline\Undottedtocline\fi\sv@chaptoc@[#1]}%
\def\chapelof@[#1]{\ifundottedmtc\let\dottedtocline\Undottedtocline\fi\sv@chapelof@[#1]}%
\def\chaplot@[#1]{\ifundottedmtc\let\dottedtocline\Undottedtocline\fi\sv@chaplot@[#1]}

9.40 The \dominitoc command and its siblings

The three commands \dominitoc, \dominilof and \dominilot are, of course, very similar. They take the \jobname.toc file (resp. the \jobname.lof and \jobname.lot files) produced by the previous \LaTeX run and cut it in slices (one slice per chapter or starred chapter) into the \jobname.mtc\langle N\rangle files (resp. the \jobname.mlf\langle N\rangle and \jobname.mlt\langle N\rangle files), using specific lines in the \jobname.toc (resp. \jobname.lof and \jobname.lot) file. These lines are essentially chapter-level entry commands (like \contentsline{chapter}..., \contentsline{xchapter}..., \chapbegin and \starchapter) delimiting chapters in the TOC (or in the LOF or the LOT). Analog part-level lines delimit parts, hence also chapters.

As \dominitoc has an optional argument, whose default value is "l" (left), it calls \dominitoc@ with a argument delimited by brackets. The macros are \dominitoc (user interface), which calls \dominitoc@[l] (or with the optional argument of \dominitoc). Then \dominitoc@[l] processes its argument and calls \@dominitoc. \@dominitoc calls \dominitoc (passing \jobname as argument) then close the minitoc file written. \@dominitoc reset to zero the counter of mini-tables, calls \@ifnextchar\MTC@next\jobname.toc (where \jobname is the value of \jobname), then reset again to zero the counter of mini-tables. Each call to \dominitoc@ (i.e., to \dominitoc) sets the flag \@dominitoc@used@true. This will be used later for a hint (which detects that you have correctly called \minitoc after \dominitoc and that both or neither have been called). See section 9.81.2.2 on page 424. The code is similar for \dominilof and \dominilot.

The \dominitoc command extracts information from the \jobname.toc file and create the minitocs files, with the adequate extension.

The \dominilof command extracts information from the \jobname.lof file and create the minilofs files, with the adequate extension.
\makeatletter
\setcounter{mtc}{0}
\MLF@next#1.lof\relax\}
\setcounter{mtc}{0}\}
def\dominilof\{\@inextchar\{\{\dominilof\}\{\dominilof\[l\]}}
\dominilot The \dominilot command extracts information from the .lot file and create the minilots files, with the adequate extension.
\def\@dominilot#1{{% 
\makeatletter 
\setcounter{mtc}{0} 
\MLT@next#1.lot\relax\}
\setcounter{mtc}{0} 
\def\dominilot\{\@ifnextchar[\{\dominilot@}{\dominilot@[l]} 
\if@dominitoc@used@ 
\if@mtc@hints@ 
\@mtc@hints@given@true 
\df@mtitc 
\e@mti 
\n@mti 
\c@mti 
\l@mti 
\r@mti 
\@@dominitoc 
\if@dominitoc@used@ 
\if@mtc@hints@ 
\if@dominitoc@used@ 
\mtcPackageInfo[10045]{minitoc(hints)} 
\MessageBreak 
\if #1e\let\df@mtitc\e@mti% 
\else\if #1n\let\df@mtitc\n@mti% 
\else\if #1c\let\df@mtitc\c@mti% 
\else\if #1l\let\df@mtitc\l@mti% 
\else\if #1r\let\df@mtitc\r@mti% 
\fi\fi\fi\fi\fi% 
\@@dominitoc 
\if@dominilof@used@ 
\if@mtc@hints@ 
\if@dominilof@used@ 
\mtcPackageInfo[10045]{minitoc(hints)} 
\MessageBreak 
\if #1e\let\df@mtitc\e@mti% 
\else\if #1n\let\df@mtitc\n@mti% 
\else\if #1c\let\df@mtitc\c@mti% 
\else\if #1l\let\df@mtitc\l@mti% 
\else\if #1r\let\df@mtitc\r@mti% 
\fi\fi\fi\fi\fi% 
\@@dominilof
Some code to flag the use of the command and manage the position of the minilot title; a hint detects any spurious invocation.

\def\dominilot@[#1]{% 
  \if@mtc@hints@ 
    \if@dominilot@used@ 
      \mtcPackageInfo[I0045]{minitoc(hints)}% 
      \MessageBreak 
      {The \string\dominilot{ #1} \space command \MessageBreak 
      \MessageBreak 
      has been invoked more than once \MessageBreak} 
      \global@mtc@hints@given@true 
    \fi 
  \fi 
  \global@dominilot@used@true 
  \if #1e\let\df@mtilt\e@mti\% 
  \else\if #1n\let\df@mtilt\n@mti\% 
  \else\if #1c\let\df@mtilt\c@mti\% 
  \else\if #1l\let\df@mtilt\l@mti\% 
  \else\if #1r\let\df@mtilt\r@mti\% 
  \fi\fi\fi\fi\fi\% 
  \@@dominilot} 

These macros invoke the \dominitoc... macros to create the mini-table file, then close the file descriptor.

\def\@@dominitoc{\@dominitoc{\jobname}\immediate\closeout\tf@mtc} 
\def\@@dominilof{\@dominilof{\jobname}\immediate\closeout\tf@mtc} 
\def\@@dominilot{\@dominilot{\jobname}\immediate\closeout\tf@mtc}
9.4.0.1 Analysis and splitting of the TOC file

This is done via a loop managed by the following macros\textsuperscript{8}:
\begin{verbatim}
\MTC@next\MTC@list\MTC@loop\def\MTC@next#1\relax#2\{}\edef\MTC@list{#2}\MTC@loop{#1}\}
\MTC@toc\MTC@list\MTC@explist\def\MTC@toc{\ifx\MTC@list\@empty\else\expandafter\MTC@explist\fi}
\MTC@contentsline\arabic\chapter\themtc\tf@mtc\def\MTC@contentsline#1#2#3#4{\gdef\themtc\arabic{mtc}\expandafter\ifx\csname #1\endcsname\chapter\stepcounter{mtc}\
\if@mtc@longext@\themtc\mtcname\tf@mtc\closeout\openout\immediate\closeout\tf@mtc\immediate\openout\tf@mtc=\mtcname\fi}
\if@mtc@longext@\themtc\mtcname\tf@mtc\closeout\openout\immediate\closeout\tf@mtc\immediate\openout\tf@mtc=\mtcname\fi
\end{verbatim}

This code is derived from the \texttt{xr} package \cite{114}, by David P. Carlisle, with his permission. Some modifications were made by Heiko Oberdiek, Didier Vaerna, and Bernd Jahn for the support of hyperref, essentially by adding an argument to some macros, to use the hyperlink argument in the contents lines.
We need a similar code to detect TOC entries for appendices in the `memoir` class:

```latex
\if@mtc@longext@
\themtc
\mtcname
\stepcounter{mtc}\
\if@mtc@longext@
\mtcPackageInfo[30033]{minitoc}{Writing \jobname.mtc \themtc @gobble}
\def\mtcname{\jobname.mtc \themtc}
\else
\mtcPackageInfo[30033]{minitoc}{Writing \jobname.M \themtc @gobble}
\def\mtcname{\jobname.M \themtc}
\fi
\immediate\closeout\tf@mtc
\immediate\openout\tf@mtc=\mtcname
\fi
```

Now, we filter the relevant contents lines, the token register `\mtc@toks` is used as a verbatim memory.

```latex
\mtc@toks{\noexpand\leavevmode #2}\
```

Each interesting contents line is copied, with a font command added before it. We begin with the standard sectioning commands, below `\chapter`:

```latex
\expandafter\ifx\csname #1\endcsname\section
\MTCWriteContentsline{#1}{mtcS}{#3}{#4}
\fi
\expandafter\ifx\csname #1\endcsname\subsection
\MTCWriteContentsline{#1}{mtcSS}{#3}{#4}
\fi
\expandafter\ifx\csname #1\endcsname\subsubsection
\MTCWriteContentsline{#1}{mtcSSS}{#3}{#4}
\fi
\expandafter\ifx\csname #1\endcsname\paragraph
\MTCWriteContentsline{#1}{mtcP}{#3}{#4}
\fi
\expandafter\ifx\csname #1\endcsname\subparagraph
\MTCWriteContentsline{#1}{mtcSP}{#3}{#4}
\fi
```

A coffee break contents line is written for `\coffee`:

```latex
\MTCWriteCoffeeLine
```

---

9 Tim Arnold has signaled the problem; thanks!
\starchapter
\stepcounter{mtc}
\if\@mtc@longext@
  \mtcname
  \themtc
  \tf@mtc
  \closeout
  \openout
\fi

If it is \starchapter (for a starred chapter), we increment the mtc counter, build a new minitoc file name, close the file descriptor and open it with this new file.

\expandafter\ifx\csname #1\endcsname\starchapter
  \stepcounter{mtc}\
  \if\@mtc@longext@
    \mtcPackageInfo[I0033]{minitoc}{Writing\space\jobname.mtc\themtc\@gobble}\
    \def\mtcname{\jobname.mtc\themtc}\
  \else
    \mtcPackageInfo[I0033]{minitoc}{Writing\space\jobname.M\themtc\@gobble}\
    \def\mtcname{\jobname.M\themtc}\
  \fi
  \immediate\closeout\tf@mtc\
  \immediate\openout\tf@mtc=\mtcname
\fi

\starsection
\MTC@WriteContentsline
\starsubsection
\starsubsubsection
\starparagraph
\starsubparagraph

For starred sectionning commands lower than \chapter, a contents line is written into the minitoc file, with a font command added:

\expandafter\ifx\csname #1\endcsname\starsection
  \MTC@WriteContentsline{#1}{mtcS}{#3}{#4}\
\fi
\expandafter\ifx\csname #1\endcsname\starsubsection
  \MTC@WriteContentsline{#1}{mtcSS}{#3}{#4}\
\fi
\expandafter\ifx\csname #1\endcsname\starsubsubsection
  \MTC@WriteContentsline{#1}{mtcSSS}{#3}{#4}\
\fi
\expandafter\ifx\csname #1\endcsname\starparagraph
  \MTC@WriteContentsline{#1}{mtcP}{#3}{#4}\
\fi
\expandafter\ifx\csname #1\endcsname\starsubparagraph
  \MTC@WriteContentsline{#1}{mtcSP}{#3}{#4}\
\fi

\MTC@explist

The loop to read the lines of the TOC file; it expands the list of entries and call \MTC@next to process the first one:

\def\MTC@explist{\expandafter\MTC@next\MTC@list\}
Read the next entry of the .toc file.
\read\@inputcheck to\MTC@line
\MTC@test
The ..... make sure that \MTC@test has enough arguments:
\MTC@contentsline The \MTC@test macro finds the “interesting” commands in the TOC file, mainly to delimit chapters.
\MTC@test Look at the first token of the line. If it is an interesting entry, process it. If it is \@input, add the file to the list. Otherwise ignore. Go around the loop if not at end of file. Finally process the next file in the list.
\MTC@test
\MTC@contentsline
\contentsline
\mct@string
\@input
\MTC@list
\chapterend
\closeout
\tf@mct
\openout
\chapterbegin
\addtocounter
\MTC@toc
\MTC@read
\addtocounter{mct}{-1}%
\fi\fi\fi

\textsuperscript{10}The macro \MTC@test has been patched to call \MTC@contentsline with four parameters instead of three (thanks to Heiko Oberdiek, Didier Verna, Bernd Jähne and A. J. “Tony” Roberts). The same remark applies to similar macros.
9.41 Mini-lists of figures

The code is similar to the code for mini-tables of contents, but with less commands to recognize.

9.41.1 Analysis and splitting of the list of figures file

\MLF@next This is done via a loop managed by the following macros:
\MLF@list\MLF@loop Processes the next entry in the list and removes it from the head of the list:

\MLF@lof Checks if the list is empty:
\MLF@list\MLF@explist

\MLF@contentsline The macro \MLF@contentsline analyses the lines read from the LOF file and detects interesting keywords. If \xchapter is found, the counter mtc is incremented and a new minilof file is created.

\if@mtc@longext\mtcPackageInfo{I0033}{minitoc} The name of the minilof file is built from \jobname and a long or short extension:
The token register \mtc@toks is used to pass the entry to \MTC@WriteContentsline. If we found a \figure entry, we copy it into the minilof file:

\expandafter\if\csname #1\endcsname\figure
\mtc@toks{\noexpand\leavevmode#2}\MTC@WriteContentsline{#1}{mlf}{#3}{#4}
\fi
\expandafter\if\csname #1\endcsname\subfigure
\mtc@toks{\noexpand\leavevmode#2}\MTC@WriteContentsline{#1}{mlfS}{#3}{#4}
\fi

The loop to read the LOF file; it expands the list of entries and calls \MLF@next to process the first one:

\MLF@explist
\MLF@next
\MLF@list
\MLF@loop
And now, we scan the .lof file:
9.42 Mini-lists of tables

The code is similar to the code for mini-tables of contents, but with less commands to recognize.
9.42.1 Analysis and splitting of the list of tables file

This is done via a loop managed by the following macros:

- \MLT@next
- \MLT@list
- \MLT@loop

Processes the next entry in the list and removes it from the head of the list:

\MLT@lot
\MLT@list
\MLT@explist

Checks if the list is empty:

\MLT@contentsline
\arabic
\xchapter

The macro \MLT@contentsline analyses the lines read from the LOT file and detects interesting keywords. If \xchapter is found, the mtc counter is incremented and a new minilot file is created.

The name of the minilot file it build from \jobname and a long or short extension:

The token register \mtc@toks is used to pass the entry to \MTC@WriteContentsline. If we found a \table entry, we copy it into the minilot file:
The loop to read the LOT file; it expands the list of entries and calls \MLT@next to process the first one:

\MLT@explist The loop to read the LOT file; it expands the list of entries and calls \MLT@next to process the first one:

\MLT@list

\MLT@loop And now, we scan the .lot file:

\MLT@loop And now, we scan the .lot file:

\MLT@read Read the next entry in the .lot file:

\MLT@read Read the next entry in the .lot file:

\MLT@line The .... make sure that \MLT@test has enough arguments:

\MLT@test The .... make sure that \MLT@test has enough arguments:

\MLT@test The \MLT@test macro finds the “interesting” commands in the LOT file, mainly to delimit chapters.
Look at the first token of the line. If it is an interesting entry, process it. If it is \@input, add the file to the list. Otherwise ignore. Go around the loop if not at end of file. Finally process the next file in the list.

Note that we terminate with a closing brace to end the chapter-level macros (end of the else branch of a \@ifundefined{chapter} alternative).

\begin{macro}{\MTC@WriteContentsline}

The \MTC@WriteContentsline macro makes the definition of \MTC@contentsline shorter. An extra \edef level is removed (Heiko Oberdiek):

The arguments of \MTC@WriteContentsline are:

- #1: the #1 argument of \MTC@contentsline;
- #2: font shorthand ⇒ \csname #2font\endcsname;
- #3: the #3 argument of \MTC@contentsline;
- #4: the #4 argument of \MTC@contentsline (hyperlink).

The token register \mtc@toks is used to pass the entry to \MTC@WriteContentsline.
And the same for a "coffee" line. The arguments of the macro \MTC@WriteCoffeeline are:
#1: the #1 argument of \MTC@contentsline;
#2: the #3 argument of \MTC@contentsline.

The token register \mtc@toks is used to pass the entry to \MTC@WriteCoffeeline. Le registre token \mtc@toks est utilisé pour passer l’entrée à \MTC@WriteCoffeeline.

If the counters lofdepth and lotdepth are defined, we create the corresponding new counters: partlofdepth and partlotdepth. These counters are initialized to 2. This is done after the loading of the packages, in an \AtBeginDocument block:
9.45 Part level commands

If \part is defined, we define some utility commands, a counter (ptc) for the parttocs and related commands (\theptc, \Thethepart, \adjustptc, \decrementptc, \incrementptc), the obsolete command \firstpartis, and the depth counter parttocdepth.

\if@firstpartis@used@% \newcounter{partlotdepth} \setcounter{partlotdepth}{2}%
\newcommand{\firstpartis}{% \mtcPackageWarning{W0004}{minitoc}{% \string\firstpartis \space is an obsolete (ignored) command} %\@firstpartis@used@true}
\newcounter{ptc}
\setcounter{ptc}{0}
\newcommand{\adjustptc}[1]{\addtocounter{ptc}{#1}}
\def{\decrementptc}{\addtocounter{ptc}{-1}}
\def{\incrementptc}{\addtocounter{ptc}{+1}}
\def{\theptc}{\arabic{ptc}}
\newcounter{parttocdepth}
\setcounter{parttocdepth}{2}

But, sometimes, we need to make a difference between book/report and article classes (is \chapter defined?), to have a different layout: the definition of \ptc@rule is empty except if \chapter is undefined. By default, there is no rule before/after parttocs, parttols, and partlots for books. You should redeclare \ptc@rule if you want these rules.

\if@undefined{chapter} %\ptc@rule \columnwidth \newcommand{\ptc@rule}{% \if@undefined{chapter} %\newcommand{\ptc@rule}{% \def{\ptc@rule}{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}} %\let{\ptc@rule}{\relax}
\newlength{ptcindent}
\if@undefined{chapter} \setlength{ptcindent}{24\p@}\setlength{ptcindent}{2\p@}

And we declare the default indentation (both sides) of the parttocs:
9.46 Fonts for the parttocs

\ptcfont We define the fonts for the parttocs. Note that they are larger if \chapter is defined (book/report-like document classes) than when it is not (article-like document classes):
\ptcfont
\ptcSfont
\ptcSSfont 3777 \ifnum\catcode`\mathchar`\relax=% \def\ptcfont{\small\rmfamily\upshape\mdseries} % the parttoc
\ptcSPfont 3779 \def\ptcSfont{\small\rmfamily\upshape\bfseries}% (sections)
\pltfont 3780 \let\ptcSSfont\ptcfont % (subsections)
\pltSfont 3781 \let\ptcSSSfont\ptcfont % (subsubsections)
\ptifont 3782 \let\ptcPfont\ptcfont % (paragraphs)
\ptSPfont 3783 \let\ptcSPfont\ptcfont % (subparagraphs)
\plffont 3784 \let\plffont\ptcfont % (figures)
\plfSfont 3785 \let\plfSfont\ptcfont % (subfigures)
\pltfont 3786 \let\pltfont\ptcfont % (tables)
\pltSfont 3787 \let\pltSfont\ptcfont % (subtables)
\ptifont 3788 \def\ptifont{\Large\rmfamily\upshape\bfseries}% titles
3789 \%
\ptcfont If \chapter is defined, the fonts are larger and \ptcCfont must be defined:
\ptcfont
\ptcCfont 3790 \%
\ptcSfont 3791 \def\ptcfont{\normalsize\rmfamily\upshape\mdseries} % the parttoc
\ptcSSfont 3792 \def\ptcSfont{\normalsize\rmfamily\upshape\bfseries}% (sections)
\ptcSPfont 3793 \def\ptcSPfont{\normalsize\rmfamily\upshape\mdseries}% (subsections)
\pltfont 3794 \let\ptcSSfont\ptcfont % (subsections)
\pltSfont 3795 \let\ptcSSSfont\ptcfont % (subsubsections)
\ptifont 3796 \let\ptcPfont\ptcfont % (paragraphs)
\ptSPfont 3797 \let\ptcSPfont\ptcfont % (subparagraphs)
\plffont 3798 \let\plffont\ptcfont % (figures)
\plfSfont 3799 \let\plfSfont\ptcfont % (subfigures)
\pltfont 3800 \let\pltfont\ptcfont % (tables)
\pltSfont 3801 \let\pltSfont\ptcfont % (subtables)
\ptifont 3802 \def\ptifont{\LARGE\rmfamily\upshape\bfseries}% titles
3803 }

9.47 Default titles for part-level mini-tables

\parttoc We define the default position, the fonts and the layout for titles of the part-level mini-tables (\parttoc, \partlof and \partlot). This formating is different if \chapter is defined or undefined.
If \texttt{\chapter} is undefined, the definitions are very simple, for centered, flushleft, flushright or empty titles. Here, empty titles need a vertical correction (Frank Mittelbach).

But, if \texttt{\chapter} is defined, we must simulate the formatting of a chapter head, which is more complex. Here, empty titles need a vertical correction (Frank Mittelbach).

For a centered title, we must also test if the main text is on two columns:

\begin{verbatim}
\if@twocolumn
\@topnewpage \@afterheading \@makephead@c{#1}\@afterheading \fi
\parindent \z@ \centering \ptifont #1\par \nobreak \vskip \mtcgapafterheads
\end{verbatim}
9.48 The \texttt{ptc@verse} environment

Each parttoc is placed inside a \texttt{ptc@verse} environment. This environment is analogous to the standard \texttt{verse} environment and hence defined via two commands: \texttt{ptc@verse} and \texttt{endptc@verse}. As it is a list environment, we first define (in a local way) \texttt{	extbackslash list}, then call \texttt{	extbackslash list{}}, and set some dimensions like \texttt{\itemsep}, \texttt{\itemindent}, \texttt{\listparindent}, \texttt{\topsep}, \texttt{\parskip}, \texttt{\partopsep}, \texttt{\topsep}, \texttt{\parsep} is set to zero if the \texttt{\\k-tight} option is active (this reduces the spacing between the lines). \texttt{\parskip} is set to zero if the \texttt{\k-tight} option is active (this reduces the spacing between the lines). Both margins are set to \texttt{\ptcindent}. \texttt{\endptc@verse} terminates the list and discourages a page break. The \texttt{ptc@verse} environment has an argument which is an horizontal offset (a command like \texttt{\ptcoffset}).

9.49 The part level mini-tables: \texttt{parttoc}, \texttt{partlof}, and \texttt{partlot}

These commands are essentially similar to the \texttt{minitoc} command, except that they should be placed after a \texttt{\part} command to produce a parttoc, a partlof or a partlot, and the formatting is different and depends on the availability of the \texttt{\chapter} command (for the fonts and the horizontal rules). The code is very similar. The \texttt{\partlof} and \texttt{\partlot} commands are siblings of the \texttt{\parttoc} command. Note that \texttt{\parttoc}, \texttt{\partlof} and \texttt{\partlot} use page styles, because \texttt{\beforepart...} and \texttt{\afterpart...} commands imply usually a \texttt{\clear[double]page} command, and hence \texttt{\markboth{...}{...}} must be called.
9.49.1 The \parttoc command

\parttoc\parttoc@\@ifnextchar The \parttoc command must be used after \part if you need a parttoc (no automatic parttoc). First, \parttoc detects the presence of its optional argument, and uses its default value, d, if it is missing. Then, \parttoc@ is called with the effective position as argument:

\def\parttoc{\@ifnextchar[{{\parttoc}[d]}}

\parttoc\parttoc@\@ifnextchar The \parttoc@ macro does the real work. It first sets the flag \if@parttoc@used@ (for a consistency hint) and checks if long extensions are used or not (to create the name of the parttoc file):

\def\parttoc@[#1]{{%\g@addto@macro\parttoc@{\@tocfile{ptc\Thepart}}}%\g@addto@macro\parttoc@{\@tocfile{P\Thepart}}%\fi%}

\begin{verbatim}
\if@parttoc@used@
\if@mtc@longext@
\def\@tocfile{ptc\Thepart}\
\else
\def\@tocfile{P\Thepart}\
\fi
\mtc@CkFile{jobname.\@tocfile}
\if@mtc@FE
\mtcPackageInfo{I0006}{minitoc}{jobname.\@tocfile\ is empty}
\@mtc@empty@parttoc@true
\else
\beforeparttoc
\mtc@markboth\@mkboth\thispageparttocstyle\MakeUppercase\ptctitle
\ifdefined{chapter}{{%\global\let\mtc@markboth\markboth \global\let\@mkboth\markboth
\thispageparttocstyle\MakeUppercase\ptctitle}
\mtc@markboth{\MakeUppercase{\ptctitle}}{\MakeUppercase{\ptctitle}}%}
\end{verbatim}

\mtc@CkFile{jobname.\@tocfile}
\if@mtc@FE
\mtcPackageInfo{I0006}{minitoc}{jobname.\@tocfile\ is empty}
\@mtc@empty@parttoc@true
\else
\beforeparttoc
\mtc@markboth\@mkboth\thispageparttocstyle\MakeUppercase\ptctitle
\ifdefined{chapter}{{%\global\let\mtc@markboth\markboth \global\let\@mkboth\markboth
\thispageparttocstyle\MakeUppercase\ptctitle}
\mtc@markboth{\MakeUppercase{\ptctitle}}{\MakeUppercase{\ptctitle}}%}
We enter in a \texttt{ptc@verse} environment to format the parttoc. The toc depth is forced (locally) to \texttt{parttocdepth}. A little trick is necessary to adjust the position. A blank line is necessary to avoid a negative indentation.
If the contents lines must have no numbers, we replace the macro \@dottedtocline with its undotted version. For chapter-level entries, we must invoke \l@chapter ignoring the page number argument. A hook (redefinissable command) is added, and the formatting settings coming from \mtcsetformat are activated via \ptc@setform. Then the parttoc file is inserted, followed by a strut, and the ptc@verse environment is terminated. The "open" and "close" features are called just before and after the insertion of the mini-table file.

\begingroup
\makeatletter
\@ifundefined{ptc@pgno}\let\@dottedtocline\@undottedtocline\{}
\@ifundefined{ptc@pgno}\let\l@chapter@SVPN\l@chapter\def\l@chapter##1##2{\l@chapter@SVPN{##1}{\hbox{}}}}{}
\@fileswfalse\mtc@hook@beforeinputfile
\ptc@setform
\openparttoc\global\inparttoctrue
\@input{\jobname.\@tocfile}\global\inparttocfalse\closeparttoc
\vspace{-1ex}\vspace{-1\baselineskip}\leavevmode\mtc@strut\global\@nobreakfalse\endgroup
\end{ptc@verse}

The final part is just to add the bottom rule, if necessary, a possible page break (if \chapter is not defined), and \afterparttoc.

\partlof@@ifnextchar[@ifnextchar]
\partlof\@ifnextchar\partlof@\partlof@{d}

This command must be used after \part if you need a partlof (no automatic partlof). First, \partlof detects the presence of its optional argument, and uses its default value, d, if it is missing. Then, \partlof@ is called with the effective position as argument:

\def\partlof{[\@ifnextchar[[@partlof@\partlof@{d}]}
The \partlof@ macro does the real work. It first sets the flag \if@partlof@used@ (for a consistency hint) and checks if long extensions are used or not (to create the name of the partlof file):

\partlof@ 3916 \def\partlof@[#1]{% 3917 \global@partlof@used@true 3918 \if@mtc@longext@% 3919 \def\@tocfile{plf\Thepart} 3920 \else 3921 \def\@tocfile{G\Thepart} 3922 \fi

Then, we check the presence of the partlof file and give a warning if it is not here:

\@tocfile 3923 \mtc@CkFile\jobname.\@tocfile 3924 \if@mtc@FE 3925 \mtcPackageInfo[I0006]{minitoc}{\jobname.\@tocfile\space is empty} 3926 \@mtc@empty@partlof@true 3927 \else 3928 \fi

If the partlof file is present, we can insert it, but we must add some presentation code: first, \beforepartlof, of course:

\beforepartlof 3929 \beforepartlof

If \chapter is defined, we just set the page marks with the partlof title and set the page style:

\chapter 3930 \@ifundefined{chapter}{}% 3931 \@ifundefined{chapter}{\global\let\mtc@markboth\markboth 3932 \global\let\@mkboth\markboth 3933 \thispagepartlofstyle 3934 \mtc@markboth{\MakeUppercase{plftitle}}{\MakeUppercase{plftitle}}% 3935

A samepage environment is begun, then the argument is treated to set the position of the partlof title. If the title string is empty, this forces the positionning.

\force@ptilf 3936 \relax\begin{samepage}\% 3937 \if #1e\let\do@ptilf\e@pti 3938 \else\if #1n\let\do@ptilf\n@pti 3939 \else\if #1c\let\do@ptilf\c@pti 3940 \else\if #1l\let\do@ptilf\l@pti 3941 \else\if #1r\let\do@ptilf\r@pti 3942 \else\if #1d\let\do@ptilf\df@ptilf 3943 \fi\fi\fi\fi\fi\fi

A samepage environment is begun, then the argument is treated to set the position of the partlof title. If the title string is empty, this forces the positionning.
We adjust some formatting parameters and avoid a page break between the title and the parttoc, then we set the font:

\raggedright \parskip=\z@\%
\reset@font \plffont%

\plf@rule The parttoc title is set in a \texttt{tabular} environment (to inhibit a page break between the title and the top rule), with a rule at its bottom if necessary. This rule is an \texttt{\hline}. It is the top rule of the partlof.
\texttt{\columnwidth} \texttt{\plftitle} \texttt{\hline}
\texttt{\do@ptilf} \texttt{\mtc@v \plftitle}\texttt{\hline} \texttt{\mtc@hstrut}\texttt{\end{tabular}}%
\fi

\plfoffset We enter in a \texttt{ptc@verse} environment to format the partlof. If necessary, the toc depth is forced (locally) to \texttt{partlofdepth}. A little trick is necessary to adjust the position. A blank line is necessary to avoid a negative indentation.

\leavevmode\mtc@BBR \vskip -.5\baselineskip
If the contents lines must have no numbers, we replace the macro \@dottedtocline with its undotted version. A hook is added, and the formatting settings coming from \mtcsetformat are activated via \plf@setform. Then the partlof file is inserted, followed by a strut, and the ptc@verse environment is terminated. The “open” and “close” features are called just before and after the insertion of the mini-table file.

The final part is just to add the bottom rule, if necessary, a possible page break (if \chapter is not defined), and \afterpartlof. The blank line (\) is essential.

9.49.3 The \partlot command

This command must be used after \part if you need a partlot (no automatic partlot). First, \partlot detects the presence of its optional argument, and uses its default value, d, if it is missing. Then, \partlot@ is called with the effective position as argument:

The \partlot@ macro does the real work. It first sets the flag \if@partlot@used@ (for a consistency hint) and checks if long extensions are used or not (to create the name of the partlot file):
Then, we check the presence of the partlot file and give a warning if it is not here:

\begin{samepage}
\if#1e\let\do@ptilt\e@pti
\else\if#1n\let\do@ptilt\n@pti
\else\if#1c\let\do@ptilt\c@pti
\else\if#1l\let\do@ptilt\l@pti
\else\if#1r\let\do@ptilt\r@pti
\else\if#1d\let\do@ptilt\df@pti
\fi\fi\fi\fi\fi\fi
\mtc@CkStr\plttitle\if@mtc@FE \let\do@ptilt\e@pti\relax\fi
\end{samepage}
We adjust some formatting parameters and avoid a page break between the title and the partlot, then we set the font:

\begin{tabular}{@{}p{\columnwidth}@{}}
\reset@font\ptifont\do@ptilt{\mtc@v\plttitle}\\
\end{tabular}

The partlot title is set in a tabular environment (to inhibit a page break between the title and the top rule), with a rule at its bottom if necessary. This rule is an \textbackslash hline. It is the top rule of the partlot.

Then, we adjust the position under the top rule and set the indentation and some formatting parameters:

\begin{ptc@verse}{\pltoffset}\
\@ifundefined{c@lotdepth}{}\
{\c@lotdepth=\c@partlotdepth}\\
{\ifnum\c@lotdepth<1\relax\c@lotdepth=1\fi}\\
{\vskip -.5\baselineskip}
\end{ptc@verse}

We enter in a ptc@verse environment to format the parttoc. If necessary, the toc depth is forced (locally) to partlotdepth. A little trick is necessary to adjust the position.
If the contents lines must have no numbers, we replace the macro `\@dottedtocline` with its undotted version. A hook is added, and the formatting settings coming from `\mtcsetformat` are activated via `\plt@setform`. Then the partlot file is inserted, followed by a strut, and the `ptc@verse` environment is terminated. The “open” and “close” features are called just before and after the insertion of the mini-table file.

The final part is just to add the bottom rule, if necessary, a possible page break (if `\chapter` is not defined), and `\afterpartlot`. The blank line (`\`) is essential.

The following auxiliary commands are used in the printing of parttocs. Note that `\l@xpart` uses a huge depth to inhibit the printing of its contents line (except if you cheat). These commands are similar to `\l@subsection`, only the arguments have been altered:

```latex
\begin{verbatim}
\def\l@xpart{\@dottedtocline{\@M}{1.0em}{2.3em}}
\def\l@pchapter{\@dottedtocline{1}{1.0em}{2.3em}}
\def\l@psect{\@dottedtocline{2}{1.0em}{2.3em}}
\end{verbatim}
```

**9.50 Auxiliary commands for printing parttocs**
9.51 Patching the \part command, continued

We patch both branches of the \part command: \@part (unstarred \part) and \@spart (\part*). We add the incrementation of the ptc counter to both branches. In the unstarred branch, we add xpart entries in the TOC, the LOF and the LOT. In the starred branch, we add a \partbegin line in the TOC. This command is just a marker and does nothing real (relax).

\let\sv@part\mtc@svpart
\def\@part[#1]{\sv@part[#1]{#2}\relax
\addcontentsline{lof}{xpart}{#1}\
\addcontentsline{lot}{xpart}{#1}\
\addcontentsline{toc}{xpart}{#1}\
\stepcounter{ptc}}

\let\sv@spart\@spart
\def\@spart{\stepcounter{ptc}\sv@spart}
\def\@spart{\addtocontents{toc}{\protect\partend}\ptc@spart}
\def\@spart{\addtocontents{toc}{\protect\partbegin}\ptc@spart}
\let\partend\relax
\let\partbegin\relax

9.52 The \doparttoc command and its siblings

The \doparttoc command works like the \dominitoc command, \dopartlof like \dominilof and \dopartlot like \dominilot.

\def\@doparttoc#1{{\makeatletter\setcounter{ptc}{0}\PTC@next#1.toc\relax\setcounter{ptc}{0}}}

\def\@dopartlof#1{{\makeatletter\setcounter{ptc}{0}\PLF@next#1.lof\relax\setcounter{ptc}{0}}}

The \doparttoc command extracts information from the .toc file and creates the .ptc⟨N⟩ files (.ptc becomes .P on MS-DOS).

\def\@doparttoc#1{{%\makeatletter\setcounter{ptc}{0}\PTC@next#1.toc\relax\setcounter{ptc}{0}}}

\def\@dopartlof#1{{%\makeatletter\setcounter{ptc}{0}\PLF@next#1.lof\relax\setcounter{ptc}{0}}}

The \dopartlof command extracts information from the .lof file and creates the .plf⟨N⟩ files (.plf becomes .G on MS-DOS).
The \dopartlot command extracts information from the .lot file and creates the .plt files (.plt becomes .U on MS-DOS).

4090 \def\@dopartlot#1{% 4091 \makeatletter 4092 \setcounter{ptc}{0} 4093 \PLT@next#1.lot\relax\}\setcounter{ptc}{0}}%

\doparttoc\dopartlot\@ifnextchar

We define the user macros, who detect the optional argument:

4040 \def\doparttoc\{\@ifnextchar[\{\doparttoc@\}{\doparttoc@[l]}}
4045 \def\dopartlof\{\@ifnextchar[\{\dopartlof@\}{\dopartlof@[l]}}
4046 \def\dopartlot\{\@ifnextchar[\{\dopartlot@\}{\dopartlot@[l]}}

\doparttoc@\if@mtc@hints@\if@doparttoc@used@
\mtcPackageInfo[I0045]{minitoc(hints)}%
\doparttoc@
\MessageBreak
\MessageBreak
\global\@mtc@hints@given@true
\def\ptitc\e@pti%
\else\if #1n\let\ptitc\n@pti%
\else\if #1c\let\ptitc\c@pti%
\else\if #1l\let\ptitc\l@pti%
\else\if #1r\let\ptitc\r@pti%
\fi\fi\fi\fi%
\@@doparttoc
\fi\fi\fi\fi\fi%

\dopartlof@
\if@dopartlof@used@
\mtcPackageInfo[I0045]{minitoc(hints)}%
\dopartlof@
\MessageBreak
\MessageBreak
\global\@mtc@hints@given@true
\fi\fi\fi\fi\fi\fi\fi\fi

\dopartlot@
\if@dopartlot@used@
\mtcPackageInfo[I0045]{minitoc(hints)}%
\dopartlot@
\MessageBreak
\MessageBreak
\global\@mtc@hints@given@true
\fi\fi\fi\fi\fi\fi\fi\fi

We treat the optional argument of \dopartlot (it becomes the default position for titles of parttocs) and flag this macro as used; a hint detects any spurious invocation.

\dopartlof@
\if@dopartlof@used@
\mtcPackageInfo[I0045]{minitoc(hints)}%
\dopartlof@
\MessageBreak
\MessageBreak
\global\@mtc@hints@given@true
\fi\fi\fi\fi\fi\fi\fi\fi

We treat the optional argument of \dopartlof (it becomes the default position for titles of partlofs) and flag this macro as used; a hint detects any spurious invocation.
Commented code of the minitoc package

We treat the optional argument of \dopartlot (it becomes the default position for titles of partlofs) and flag this macro as used; a hint detects any spurious invocation.

These macros invoke the @dopart... commands to create the mini-table file, then close the file descriptor.

9.52.1 Processing macros for the parttocs

Processing the next entry in the list and remove it from the head of the list:
Commented code of the minitoc package

```
4157 \def\PTC@next#1\relax#2\{
4158 \edef\PTC@list{#2}\{
4159 \PTC@loop[#1]}%

\PTC@toc Check if the list is empty:
\PTC@list
\PTC@explist
4160 \def\PTC@toc{%
4161 \ifx\PTC@list\@empty\else\expandafter\PTC@explist\fi}

\PTC@contentsline
\part
\theptc
\tf@mtc
\ptcname
\MTC@WriteContentsLine The macro \PTC@contentsline analyses the lines read from the TOC file and detects interesting keywords. If \part is found, the ptc counter is incremented and a new partlof file is created.
4162 \def\PTC@contentsline#1#2#3#4{%
4163 \if\csname #1\endcsname\part
4164 \stepcounter{ptc}%
4165 \if\@mtc@longext@%
4166 \mtcPackageInfo[I0033]{minitoc}%
4167 {Writing\space\jobname.ptc\theptc@gobble}%
4168 \def\ptcname{\jobname.ptc\theptc}%
4169 \else
4170 \mtcPackageInfo[I0033]{minitoc}%
4171 {Writing\space\jobname.P\theptc@gobble}%
4172 \def\ptcname{\jobname.P\theptc}%
4173 \fi
4174 \immediate\closeout\tf@mtc
4175 \immediate\openout\tf@mtc=\ptcname
4176 \fi
4177 \expandafter\if\csname #1\endcsname\starpart\relax
4178 \stepcounter{ptc}%

\if\@mtc@longext@
\ptcname
We test if long or short extensions are used, to build the name of the mini-table file, then open it:
4179 \if\@mtc@longext@
4180 \mtcPackageInfo[I0033]{minitoc}%
4181 {Writing\space\jobname.ptc\theptc@gobble}%
4182 \def\ptcname{\jobname.ptc\theptc}%
4183 \else
4184 \mtcPackageInfo[I0033]{minitoc}%
4185 {Writing\space\jobname.P\theptc@gobble}%
4186 \def\ptcname{\jobname.P\theptc}%
4187 \fi
4188 \immediate\closeout\tf@mtc
4189 \immediate\openout\tf@mtc=\ptcname
4190 \fi
```
The token register \mtc@toks is used to pass the entry to \MTC@WriteContentsline:

\mtc@toks\(\text{\noexpand \leavevmode #2}\)\%

Now, we filter the relevant contents lines; this code extracts and writes info for chapters, sections, etc.:

\texttt{4191} \expandafter\ifx\csname #1\endcsname\chapter 
\MTC@WriteContentsline{#1}{ptcC}{#3}{#4} \fi
\texttt{4192} \expandafter\ifx\csname #1\endcsname\appendix 
\MTC@WriteContentsline{#1}{ptcC}{#3}{#4} \fi
\texttt{4193} \expandafter\ifx\csname #1\endcsname\pchapter 
\MTC@WriteContentsline{#1}{ptcC}{#3}{#4} \fi
\texttt{4194} \expandafter\ifx\csname #1\endcsname\section 
\MTC@WriteContentsline{#1}{ptcS}{#3}{#4} \fi
\texttt{4195} \expandafter\ifx\csname #1\endcsname\coffee 
\MTC@WriteContentsline{#1}{ptcS}{#3}{#4} \fi
\texttt{4196} \expandafter\ifx\csname #1\endcsname\subsection 
\MTC@WriteContentsline{#1}{ptcSS}{#3}{#4} \fi
\texttt{4197} \expandafter\ifx\csname #1\endcsname\subsubsection 
\MTC@WriteContentsline{#1}{ptcSSS}{#3}{#4} \fi
\texttt{4198} \expandafter\ifx\csname #1\endcsname\paragraph 
\MTC@WriteContentsline{#1}{ptcP}{#3}{#4} \fi
\texttt{4199} \expandafter\ifx\csname #1\endcsname\subparagraph 
\MTC@WriteContentsline{#1}{ptcSP}{#3}{#4} \fi

And for the starred sectioning commands:

\texttt{4200} \expandafter\ifx\csname #1\endcsname\starchapter 
\MTC@WriteContentsline{#1}{ptcC}{#3}{#4} \fi
\texttt{4201} \expandafter\ifx\csname #1\endcsname\starsection 
\MTC@WriteContentsline{#1}{ptcS}{#3}{#4} \fi
\texttt{4202} \expandafter\ifx\csname #1\endcsname\starsubsection 
\MTC@WriteContentsline{#1}{ptcSS}{#3}{#4} \fi
\texttt{4203} \expandafter\ifx\csname #1\endcsname\starsubsubsection 
\MTC@WriteContentsline{#1}{ptcSSS}{#3}{#4} \fi
\texttt{4204} \expandafter\ifx\csname #1\endcsname\starparagraph 
\MTC@WriteContentsline{#1}{ptcP}{#3}{#4} \fi
\texttt{4205} \expandafter\ifx\csname #1\endcsname\starsubparagraph 
\MTC@WriteContentsline{#1}{ptcSP}{#3}{#4} \fi
4232 \MTC@WriteContentsline{#1}{ptcP}{#3}{#4}\
4233 \fi
4234 \expandafter\ifx\csname #1\endcsname\starsubparagraph
4235 \MTC@WriteContentsline{#1}{ptcSP}{#3}{#4}\
4236 \fi
4237 }

\PTC@explist The loop to read the lines of the TOC file; expands the list of entries and call \PTC@next to process the first one:

\PTC@loop If an entry is found, loop through line by line, looking for interesting entries. Otherwise, process the next entry in the list.

\PTC@read Read the next entry of the .toc file.

\PTC@test The ..... make sure that \PTC@test has enough arguments:

\PTC@test The \PTC@test macro finds the “interesting” commands in the TOC file, mainly to delimit parts:
9.52.2 Processing macros for the partlofs

\PLF@next \PLF@lof \PLF@explist

Check if the list is empty:
\PLF@lof \PLF@list \PLF@explist

\PLF@lof \PLF@list \PLF@explist

The macro \PLF@contentsline analyses the lines read from the LOF file and detects interesting keywords. If \part is found, the ptc counter is incremented and a new partlof file is created.
\PLF@contentsline \part \thepart \tf@mtc \plfname \PLF@WriteContentsLine

\PLF@written\PLF@lof\empty else expandafter\PLF@explist\fi
We test if long or short extensions are used, to build the name of the mini-table file, then open it:

\if@mtc@longext@
  \plfname
\else
  \mtcPackageInfo[10033]{minitoc}%
  \{Writing\space jobname.plf\theptc%
  \def\plfname{jobname.plf\theptc}%
\fi
\immediate\closeout\tf@mtc
\immediate\openout\tf@mtc=\plfname
\fi

The token register \mtc@toks is used to pass the entry to \MTC@WriteContentsline. Now, we filter the relevant contents lines:

\expandafter\ifx\csname #1\endcsname\figure
  \mtc@toks{\noexpand\leavevmode#2}%
  \MTC@WriteContentsline{#1}{plf}{#3}{#4}%
\fi
\expandafter\ifx\csname #1\endcsname\subfigure
  \mtc@toks{\noexpand\leavevmode#2}%
  \MTC@WriteContentsline{#1}{plfS}{#3}{#4}%
\fi}

The loop to read the lines of the LOF file; expands the list of entries and call \PLF@next to process the first one:

\def\PLF@explist{\expandafter\PLF@next\PLF@list\}

If an entry is found, loop through line by line, looking for interesting entries. Otherwise, process the next entry in the list.

\def\PLF@loop#1{\openin\@inputcheck#1\relax
  \ifeof\@inputcheck
    \mtcPackageWarning[W0011]{minitoc}%
    {No file #1 \MessageBreak
     PARTLOFS NOT PREPARED}%
    \expandafter\PLF@lof
  \else
    \mtcPackageInfo[10035]{minitoc}%
    \{PREPARING PARTLOFS FROM #1}%
    \expandafter\PLF@read
  \fi}

Read the next entry of the .lof file.

\read@inputcheck to PLF@line

The ... make sure that PLF@test has enough arguments:

\expandafter PLF@test \PLF@line.....\PLF@%

The PLF@test macro finds the “interesting” commands in the LOF file, mainly to delimit parts:

Look at the first token of the line. If it is an interesting entry, process it. If it is @input, add the file to the list. Otherwise ignore. Go around the loop if not at end of file. Finally process the next file in the list.

\expandafter PLF@lof

9.52.3 Processing macros for the partlots

Processing the next entry in the list and remove it from the head of the list:

\immediate\closeout\tf@mtc
\immediate\openout\tf@mtc=\jobname.mtc
\addtocounter{ptc}{-1}%
\fi\fi\fi\fi
\ifeof\@inputcheck\expandafter\PLF@lof
\else\expandafter\PLF@read\fi%
Check if the list is empty:

```latex
\def\PLT@lot{% 
    \ifx\PLT@list\@empty\else\expandafter\PLT@explist\fi}
```

The macro `\PLT@contentsline` analyses the lines read from the LOT file and detects interesting keywords. If `\part` is found, the `ptc` counter is incremented and a new partlot file is created.

```latex
\def\PLT@contentsline#1#2#3#4{\expandafter\ifx\csname #1\endcsname\xpart\stepcounter{ptc}\fi}
```

We test if long or short extensions are used, to build the name of the mini-table file, then open it:

```latex
\if@mtc@longext@\pltname
```

The token register `\mtc@toks` is used to pass the entry to `\MTC@WriteContentsLine`. Now, we filter the relevant contents lines:

```latex
\expandafter\ifx\csname #1\endcsname\table\mtc@toks{\noexpand\leavevmode#2}\MTC@WriteContentsLine{#1}{plt}{#3}{#4}\fi
\expandafter\ifx\csname #1\endcsname\subtable\mtc@toks{\noexpand\leavevmode#2}\MTC@WriteContentsLine{#1}{pltS}{#3}{#4}\fi
\fi
\immediate\closeout\tf@mtc
\immediate\openout\tf@mtc=\pltname
\fi
```
The loop to read the lines of the LOT file; expands the list of entries and call \PLT@next to process the first one:

\def\PLT@explist{\expandafter\PLT@next\PLT@list}

If an entry is found, loop through line by line, looking for interesting entries. Otherwise, process the next entry in the list.

\def\PLT@loop#1{\openin\@inputcheck#1\relax
\ifeof\@inputcheck
\mtcPackageWarning[W0012]{minitoc}{No file #1}
\MessageBreak
\expandafter\PLT@lot
\else
\mtcPackageInfo[I0038]{minitoc}{PREPARING PARTLOTS FROM #1}
\expandafter\PLT@read\fi}

Read the next entry of the .lot file.

\def\PLT@read{\read\@inputcheck to\PLT@line}

The ..... make sure that \PLT@test has enough arguments:

\long\def\PLT@test#1#2#3#4#5#6\PLT@{\ifx#1\contentsline
\let\mtc@string\string\PLT@contentsline{#2}{#3}{#4}{#5}\let\mtc@string\relax
\else\ifx#1\@input

Look at the first token of the line. If it is an interesting entry, process it. If it is \@input, add the file to the list. Otherwise ignore. Go around the loop if not at end of file. Finally process the next file in the list.

\long\def\PLT@test#1#2#3#4#5#6\PLT@{\ifx#1\contentsline
\let\mtc@string\string\PLT@contentsline{#2}{#3}{#4}{#5}\let\mtc@string\relax
\else\ifx#1\@input
End of the part level stuff (begun in section 9.45 on page 325):

9.53 Depth counters for sectlofs and sectlots

If the counters lofdepth and lotdepth are defined, we create new counters for the depths of the corresponding mini-tables: sectlofdepth and sectlotdepth. These counters are initialized to 2. This is done after the loading of the packages, in an \AtBeginDocument block:

\AtBeginDocument{\c@lofdepth\c@lotdepth\newcounter{sectlofdepth}\setcounter{sectlofdepth}{2}}
\AtBeginDocument{\c@lofdepth\c@lotdepth\newcounter{sectlotdepth}\setcounter{sectlotdepth}{2}}

9.54 Section-level commands

The section-level commands are defined only if \chapter is not defined, hence in article-like document classes, and only if \section is defined:

\if@mtc@chapter@undef@ \if@mtc@section@def@

\firstsectionis\adjuststc\decrementstc\incrementstc\stc@rule\stcindent\columnwidth

We define the obsolete command \firstsectionis (with its harmless warning), the counter stc of secttocs, the \adjuststc, \decrementstc and \incrementstc commands, the depth counter secttocdepth and its default value 2 (to include at least the subsections), the horizontal rule \stc@rule (rule before/after secttoc/sectlof/sectlot), the indentation (both sides) \stcindent for the secttocs (with its default values).


9.55 Fonts commands for secttocs and co.

\stcfont We define the fonts commands for the sectocs, sectlofs and sectlots and their titles:
\stcSSfont\def\stcfont{\small\rmfamily\upshape\mdseries} % secttoc
\stcSSSfont\def\stcSSfont{\small\rmfamily\upshape\bfseries} % (subsections)
\stcPfont\let\stcSSSfont\stcfont % (subsubsections)
\stcSPfont\let\stcPfont\stcfont % (paragraphs)
\slffont\let\stcSPfont\stcfont % (subparagraphs)
\slSfont\let\stcPfont\stcfont % sectlof (figures)
\sltfont\let\slSfont\stcfont % sectlof (subfigures)
\stifont\let\sltfont\stcfont % sectlot (tables)
\stiltfont\let\stifont\stcfont % sectlot (subtables)
\stiltfont\def\stifont{\large\rmfamily\upshape\bfseries} % titles

9.56 Internal macros for title positionning

\l@sti Some internal macros for title positionning, from the optional arguments of \dosecttoc and
\c@sti \secttoc commands (and siblings). Centering, flushleft, flushright or empty titles (with a
\r@sti vertical correction for empty titles, from Frank MITTELBACH):
\e@sti \n@sti\def\c@sti#1{\null\hfill #1\hfill\null}
\r@sti\def\l@sti#1{\null\hfill #1\hfill\null}
\b@sti\def\r@sti#1{\null\hfill #1\null}
\b@sti\def\e@sti#1{\vspace{-\baselineskip}}
\b@sti\def\n@sti#1{\vspace{-\baselineskip}}
\do@stitc By default, titles are flushleft.
\df@stitc
\do@stilf\let\do@stitc\l@sti
\df@stilf\let\df@stitc\l@sti
\do@stilt
\df@stilt
\l@sti
9.57 The \textit{stc@verse} environment

The \textit{stc@verse} environment is a very simple list environment, analog to the standard \textit{verse} environment. Some formatting parameters are adjusted. The tight/loose and k-tight/k-loose package options are honored. The \textit{stc@verse} environment has an argument which is an horizontal offset (a command like \texttt{stcoffset}).

\begin{verbatim}
def\stc@verse#1{\let\\@centercr\list{}{\itemsep=\z@ \itemindent=\z@ \topsep=1ex \listparindent=\itemindent \partopsep=\z@ \iftightmtc \parsep=\z@ \fi \iftightmtc \parskip=\z@ \fi \leftmargin=\stcindent \rightmargin=\leftmargin \addtolength{\leftmargin}{+#1}\addtolength{\rightmargin}{-#1}}\item[]}\end{verbatim}

9.58 The \texttt{secttoc}, \texttt{sectlof}, and \texttt{sectlot} commands

These three commands are very similar.

9.58.1 The \texttt{secttoc} command

The \texttt{secttoc} command must be used after \texttt{section} if you need a secttoc (no automatic secttoc). Its code is similar to the code of \texttt{\texttt{minitoc}} (but simpler). First, \texttt{secttoc} detects the presence of its optional argument, and uses its default value, d, if it is missing. Then, \texttt{secttoc@} is called with the effective position as argument:

\begin{verbatim}
def\secttoc{@\ifnextchar[{{\secttoc@}[d]}}
\end{verbatim}
The \secttoc@ macro does the real work. It first sets the flag \if@sectiontoc@used@ (for a consistency hint) and checks if long extensions are used or not (to create the name of the secttoc file):

\begin{verbatim}
\def\secttoc@[#1]{\global\@secttoc@used@true
  \if@mtc@longext@
    \def\@tocfile{stc\thestc}%
  \else
    \def\@tocfile{S\thestc}%
  \fi
\end{verbatim}

Then, we check the presence and emptiness of the secttoc file and give a warning if it is not here or is empty:

\begin{verbatim}
\mtc@CkFile{\jobname.\@tocfile}
\if@mtc@FE
\mtcPackageInfo[10006]{minitoc}{\jobname.\@tocfile space is empty}
\@mtc@empty@secttoc@true
\else
\beforesecttoc
\thispagesecttocstyle
\do@stitc
\c@stitc
\l@stitc
\r@stitc
\df@stitc
\mtc@CkStr{\stctitle}
\if@mtc@FE \let\do@stitc\e@stitc \relax\fi
\end{verbatim}

If the secttoc file is present and not empty, we can insert it, but we must add some presentation code: first, \beforesecttoc, of course, and the page style feature:

\begin{verbatim}
\beforesecttoc
\thispagesecttocstyle
\end{verbatim}

We begin a samepage environment, then treat the positionning argument. If the title is empty, we simulate the "e" positionning.

\begin{verbatim}
\if#1e\let\do@stitc\e@stitc
\fi
\relax\begin{samepage}%
\if #1n\let\do@stitc\n@stitc
\else
\if #1c\let\do@stitc\c@stitc
\else
\if #1l\let\do@stitc\l@stitc
\else
\if #1r\let\do@stitc\r@stitc
\else
\if #1d\let\do@stitc\df@stitc
\fi\fi\fi\fi\fi\fi
\fi
\mtc@CkStr{\stctitle}\if@mtc@FE \let\do@stitc\e@stitc \relax\fi
\end{verbatim}
We adjust some formatting parameters and avoid a page break between the title and the secttoc, then we set the font:

\raggedright
\parskip=\z@%
\reset@font\stcfont%
\parindent=\z@%
\nopagebreak[4]%

The secttoc title is set in a \texttt{tabular} environment (to inhibit a page break between the title and the top rule), with a rule at its bottom if necessary. This rule is an \texttt{\hline}. It is the top rule of the secttoc.

\raggedright
\parskip=\z@%
\reset@font\stcfont%
\parindent=\z@%
\nopagebreak[4]%
\begin{tabular}{@{}p{\columnwidth}@{}}
\reset@font\sticfont\do@stitc{\mtc@v\stctitle}\
\end{tabular}\
\begin{tabular}{@{}p{\columnwidth}@{}}
\reset@font\sticfont\do@stitc{\mtc@v\stctitle}\hline
\end{tabular}
\fi

Then, we adjust the position under the top rule and set the indentation and some formatting parameters:

\kern-0.8\baselineskip\nopagebreak[4]%
\null\leavevmode\mtc@zrule
\mtc@BBR
\leftmargin\stcindent \rightmargin\stcindent
\itemindent=\z@\labelwidth=\z@
\labelsep=\z@\listparindent=\z@
\mtc@BBR
\begin{stc@verse}{\stcoffset}\c@tocdepth=\c@secttocdepth%
\leavevmode\null\mtc@BBR\vskip -.5\baselineskip
\mtc@BBR

If the contents lines must have no numbers, we replace the macro \texttt{@dottedtocline} with its undotted version. A hook is added, and the formatting settings coming from \texttt{\mtcsetformat} are activated via \texttt{\stc@setform}. Then the secttoc file is inserted, followed by a strut, and the \texttt{\stc@verse} environment is terminated. The “open” and “close” features are called just before and after the insertion of the mini-table file.
The final part is just to add the bottom rule, if necessary, a possible page break and
\aftersecttoc.

\stc@rule \mtc@zrule \samepage \aftersecttoc
\aftersecttoc \begin{stc@verse}
\kernaftersecttoc
\nopagebreak[4]\stc@rule\null\leavevmode\%
\vskip-1.0\baselineskip\mtc@zrule\end{samepage}\%
\par\pagebreak[1]\vspace*{-1ex}\aftersecttoc\fi%

9.58.2 The $\textbackslash$sectlof command

\sectlof $\textbackslash$sectlof$@$\@ifnextchar
$\textbackslash$sectlof$@$\{\@ifnextchar[\{\sectlof$@$\{\sectlof$[@d]\}\}
\sectlof$@$\@ifnextchar[\{\sectlof$@$\{\sectlof$@$\}
\sectlof$@$\if@sectlof$@$used$@$\@ifmtc@longext$@$\@otocfile
\thesctc
\sectlof$[@d]\{\%
\global\@sectlof$@$used$@$true
\ifmtc@longext$@$\%
\def@otocfile\slf\thesctc$%$
\else
\def@otocfile\H\thesctc$%$
\fi
\fi
Then, we check the presence and the emptiness of the sectlof file and give a warning if it is not here or is empty:

```
\mtc@CkFile\jobname.\@tocfile
\if@mtc@FE
\mtcPackageInfo[I0006]{minitoc}%
\jobname.\@tocfile\space is empty
\@mtc@empty@sectlof@true
\else
beforesectlof
\thispagesectlofstyle
\if@mtc@FE\let\do@stilf\e@sti\relax\fi
\raggedright
\parskip\z@%
\reset@font\slffont%
\parindent\z@%
\nopagebreak[4]%
\slf@rule
\tablular
\stifont
\columnwidth
\do@stilf
\mtc@v
\slftitle
\hline
```

If the sectlof file is present and not empty, we can insert it, but we must add some presentation code: first, \beforesectlof, of course, and the page style feature:

```
\beforesectlof
\thispagesectlofstyle
\mtc@markboth{\MakeUppercase{\slftitle}}{\MakeUppercase{\slftitle}}%
\beforesectlof
```

We begin a \samepage environment, then treat the positionning argument. If the title is empty, we simulate the “e” positionning.

```
\begin{samepage}\%
\if #1e\let\do@stilf\e@sti
\else\if #1n\let\do@stilf\n@sti
\else\if #1c\let\do@stilf\c@sti
\else\if #1l\let\do@stilf\l@sti
\else\if #1r\let\do@stilf\r@sti
\else\if #1d\let\do@stilf\df@stilf
\fi\fi\fi\fi\fi\fi\fi\fi\fi\fi\fi\fi\fi\fi\fi\fi
\mtc@CkStr{\slftitle}\if@mtc@FE\let\do@stilf\e@sti\relax\fi
```

We adjust some formatting parameters and avoid a page break between the title and the sectlof, then we set the font:

```
\raggedright
\parskip\z@%
\reset@font\slffont%
```

The sectlof title is set in a \tablular environment (to inhibit a page break between the title and the top rule), with a rule at its bottom if necessary. This rule is an \hline. It is the top rule of the sectlof.
Then, we adjust the position under the top rule and set the indentation and some formatting parameters:
\nopagebreak[4]\null\leavevmode\mtc@zrule\
\mtc@BBR\stcindent

We enter in a stc@verse environment to format the sectlof. The toc depth is forced (locally) to sectlofdepth. A little trick is necessary to adjust the position.

If the contents lines must have no numbers, we replace the macro \@dottedtocline with its undotted version. A hook is added, and the formatting settings coming from \mtcsetformat are activated via \slf@setform. Then the sectlof file is inserted, followed by a strut, and the stc@verse environment is terminated. The "open" and "close" features are called just before and after the insertion of the mini-table file.
The final part is just to add the bottom rule, if necessary, a possible page break and `\aftersectlof`. The blank line (`\n\n`) is essential.

The `\sectlot` command must be used after `\section` if you need a sectlot (no automatic sectlot). Its code is similar to the code of `\minilot` (but simpler). First, `\sectlot` detects the presence of its optional argument, and uses its default value, d, if it is missing. Then, `\sectlot@` is called with the effective position as argument:

\begin{verbatim}
\def\sectlot{
  \@ifnextchar[\{
    \sectlot@
  }\{\sectlot@d}\}
\end{verbatim}

The `\sectlot@` macro does the real work. It first sets the flag `\if@sectlot@used@` (for a consistency hint) and checks if long extensions are used or not (to create the name of the sectlot file):

\begin{verbatim}
\def\sectlot@[#1]{%  
  \global\@sectlot@used@true
  \if@mtc@longext@%    \def\@tocfile{slt\thestc}%
  \else%               \def\@tocfile{I\thestc}%
  \fi
\end{verbatim}

Then, we check the presence and the emptiness of the sectlot file and give a warning if it is not here or is empty:

\begin{verbatim}
\if@mtc@FE  
\@tocfile%  \mtc@CheckFile{\jobname.\@tocfile}
\if@mtc@FE%  \mtpacketinfo[10006]{minitoc}{\jobname.\@tocfile \space is empty}
\else%      \\empty@sectlot@true
\fi
\beforesectlot  \thispagesectlotstyle
\end{verbatim}

If the sectlot file is present and not empty, we can insert it, but we must add some presentation code: first, `\beforesectlot`, of course, and the page style feature:

\begin{verbatim}
\beforesectlot  \thispagesectlotstyle
\mct@markboth{\MakeUppercase{\slttitle}}{\MakeUppercase{\slttitle}}%  \beforesectlot
\end{verbatim}
We begin a `samepage` environment, then treat the positionning argument. If the title is empty, we simulate the "e" positionning.

```latex
\do@stilt
\e@sti
\n@sti
\c@sti
\l@sti
\r@sti
\df@sti
\mtc@CkStr
\slttitle
\if@mtc@FE
\samepage\relax
```

We adjust some formatting parameters and avoid a page break between the title and the sectlot, then we set the font:

```latex
\raggedright
\parskip
\sltfont
```

The sectlot title is set in a `tabular` environment (to inhibit a page break between the title and the top rule), with a rule at its bottom if necessary. This rule is an `\hline`. It is the top rule of the sectlot.

```latex
\columnwidth
\do@stilt
\mtc@v
\slttitle
\hline
```

Then, we adjust the position under the top rule and set the indentation and some formatting parameters:

```latex
\mtc@zrule
\mtc@BBR
\stcindent
```

```latex
\do@stilt
\e@sti
\n@sti
\c@sti
\l@sti
\r@sti
\df@sti
\mtc@CkStr
\slttitle
\if@mtc@FE
\samepage\relax
\raggedright
\parskip
\sltfont
\columnwidth
\do@stilt
\mtc@v
\slttitle
\hline
\raggedright
\parskip=\z@%
\reset@font\sltfont%
\parindent=\z@%
\nopagebreak[4]%
```

```latex
\do@stilt
\e@sti
\n@sti
\c@sti
\l@sti
\r@sti
\df@sti
\mtc@CkStr
\slttitle
\if@mtc@FE
\samepage\relax
\raggedright
\parskip
\sltfont
\columnwidth
\do@stilt
\mtc@v
\slttitle
\hline
\kern-0.8\baselineskip\nopagebreak[4]%
```

```latex
\do@stilt
\e@sti
\n@sti
\c@sti
\l@sti
\r@sti
\df@sti
\mtc@CkStr
\slttitle
\if@mtc@FE
\samepage\relax
\raggedright
\parskip
\sltfont
\columnwidth
\do@stilt
\mtc@v
\slttitle
```

```latex
\do@stilt
\e@sti
\n@sti
\c@sti
\l@sti
\r@sti
\df@sti
\mtc@CkStr
\slttitle
\if@mtc@FE
\samepage\relax
```

```latex
\do@stilt
\e@sti
\n@sti
\c@sti
\l@sti
\r@sti
\df@sti
\mtc@CkStr
\slttitle
\if@mtc@FE
\samepage\relax
\raggedright
\parskip
\sltfont
```

```latex
\do@stilt
\e@sti
\n@sti
\c@sti
\l@sti
\r@sti
\df@sti
\mtc@CkStr
\slttitle
\if@mtc@FE
\samepage\relax
\raggedright
\parskip
\sltfont
\columnwidth
\do@stilt
\mtc@v
\slttitle
```

```latex
\do@stilt
\e@sti
\n@sti
\c@sti
\l@sti
\r@sti
\df@sti
\mtc@CkStr
\slttitle
\if@mtc@FE
\samepage\relax
```

```latex
\do@stilt
\e@sti
\n@sti
\c@sti
\l@sti
\r@sti
\df@sti
\mtc@CkStr
\slttitle
\if@mtc@FE
\samepage\relax
\raggedright
\parskip
\sltfont
\columnwidth
\do@stilt
\mtc@v
\slttitle
```

```latex
\do@stilt
\e@sti
\n@sti
\c@sti
\l@sti
\r@sti
\df@sti
\mtc@CkStr
\slttitle
\if@mtc@FE
\samepage\relax
```

```latex
\do@stilt
\e@sti
\n@sti
\c@sti
\l@sti
\r@sti
\df@sti
\mtc@CkStr
\slttitle
\if@mtc@FE
\samepage\relax
\raggedright
\parskip
\sltfont
\columnwidth
\do@stilt
\mtc@v
\slttitle
```

```latex
\do@stilt
\e@sti
\n@sti
\c@sti
\l@sti
\r@sti
\df@sti
\mtc@CkStr
\slttitle
\if@mtc@FE
\samepage\relax
\raggedright
\parskip
\sltfont
\columnwidth
\do@stilt
\mtc@v
\slttitle
```

```latex
\do@stilt
\e@sti
\n@sti
\c@sti
\l@sti
\r@sti
\df@sti
\mtc@CkStr
\slttitle
\if@mtc@FE
\samepage\relax
\raggedright
\parskip
\sltfont
\columnwidth
\do@stilt
\mtc@v
\slttitle
```

```latex
\do@stilt
\e@sti
\n@sti
\c@sti
\l@sti
\r@sti
\df@sti
\mtc@CkStr
\slttitle
\if@mtc@FE
\samepage\relax
\raggedright
\parskip
\sltfont
\columnwidth
\do@stilt
\mtc@v
\slttitle
```

```latex
\do@stilt
\e@sti
\n@sti
\c@sti
\l@sti
\r@sti
\df@sti
\mtc@CkStr
\slttitle
\if@mtc@FE
\samepage\relax
\raggedright
\parskip
\sltfont
```

```latex
\do@stilt
\e@sti
\n@sti
\c@sti
\l@sti
\r@sti
\df@sti
\mtc@CkStr
\slttitle
\if@mtc@FE
\samepage\relax
\raggedright
\parskip
\sltfont
```

```latex
\do@stilt
\e@sti
\n@sti
\c@sti
\l@sti
\r@sti
\df@sti
\mtc@CkStr
\slttitle
\if@mtc@FE
\samepage\relax
```
We enter in a \texttt{stc@verse} environment to format the sectlot. The toc depth is forced (locally) to \texttt{sectlotdepth}. A little trick is necessary to adjust the position.

If the contents lines must have no numbers, we replace the macro \texttt{@dottedtocline} with its undotted version. A hook is added, and the formatting settings coming from \texttt{mtcsetformat} are activated via \texttt{slt@setform}. Then the sectlot file is inserted, followed by a strut, and the \texttt{stc@verse} environment is terminated. The “open” and “close” features are called just before and after the insertion of the mini-table file.

The final part is just to add the bottom rule, if necessary, a possible page break and \texttt{aftersectlot}.

We define auxiliary commands, used for the mini-tables and as delimiters in the TOC file (and LOF and LOT files). The depth of \texttt{xsect} is huge to inhibit the printing of its contents line (except if you cheat).
9.60 Patching the \section command (continued)

We patch the both branches of the \section command: \@sect for the unstarred version and \@ssect for the starred version. First, for the unstarred version (\@sect), we add a xsect contents line in the LOF and in the LOT. The test \ifnum #2=1 restricts the action to the section level macros (because \@sect is also used by \subsection and below, which have no mini-tables).

If it is a section (unstarred or starred via \@starsection), we add a xsect entry in the LOF and in the LOT:

And the remainder of the section header formatting:
Then we patch the unstarred branch (\@sect). We define also the delimiting commands \sectbegin and \sectend commands. We do not add \sectbegin if it is a subsection or deeper.

\sectbegin Then we patch the unstarred branch (\@sect). We define also the delimiting commands \sectbegin and \sectend commands. We do not add \sectbegin if it is a subsection or deeper.

\sectend \addtocontents

\@dosecttoc\@dosecttoc The \dosecttoc command extracts information from the .toc file and creates the .stc⟨N⟩ files (.stc becomes .S on MS-DOS).

\dosecttoc The \dosecttoc command extracts information from the .toc file and creates the .stc⟨N⟩ files (.stc becomes .S on MS-DOS).

9.61 The \dosecttoc command and siblings

The \dosecttoc command is very similar to \dominitoc.
The \dosectlof command extracts information from the .lof file and creates the .slf(\textbackslash N) files (.slf becomes .H on MS-DOS).

\dosectlof\textbackslash SLF@next
\def\dosectlof#1{\% 4706
\makeatletter 4707
\setcounter{stc}{0} 4708
\SLF@next#1.lof\relax\} 4709
\setcounter{stc}{0}}

The \dosectlot command extracts information from the .lot file and creates the .slt(\textbackslash N) files (.slt becomes .V on MS-DOS).

\dosectlot\textbackslash PLT@next
\def\dosectlot#1{\% 4710
\makeatletter 4711
\setcounter{stc}{0} 4712
\SLT@next#1.lot\relax\} 4713
\setcounter{stc}{0}}

We define the user-level macros, who detect the optional argument:

\dosecttoc\dosectlof\dosectlot
\@ifnextchar
\def\dosecttoc\@ifnextchar[\dosecttoc@]{\dosecttoc@[l]}
\def\dosectlof\@ifnextchar[\dosectlof@]{\dosectlof@[l]}
\def\dosectlot\@ifnextchar[\dosectlot@]{\dosectlot@[l]}

We treat the optional argument of \dosecttoc (it becomes the default position for titles of \secttocs) and flag this macro as used; a hint detects any spurious invocation.

\dosecttoc\@ifmct@hints\@ifdosecttoc@used\@mct@hints@given@true
\def\dosecttoc@[#1]{\% 4717
\df@stitc 4718
\if\mct@hints@ 4719
\if\dosecttoc@used@ 4720
\mctPackageInfo[I0045]{\mctinfo{\text{minitoc(hints)}}}\% 4721
\c@stitc 4722
\l@stitc 4723
\r@stitc 4724
\MessageBreak 4725
\global\@mct@hints@given@true 4726
\fi 4727
\global\@dosecttoc@used@true 4728
\fi 4729
\if #1e\let\df@stitc\e@stitc\% 4730
\else\if #1n\let\df@stitc\n@stitc\% 4731
\else\if #1c\let\df@stitc\c@stitc\% 4732
\else\if #1l\let\df@stitc\l@stitc\% 4733
\else\if #1r\let\df@stitc\r@stitc\% 4734
\fi\fi\fi\fi\fi\% 4735
\@dosecttoc}
We treat the optional argument of `\dosectlof` (it becomes the default position for titles of sectlofs) and flag this macro as used; a hint detects any spurious invocation.

\dosectlof% 
\if@mtc@hints% 
\@dosectlof@used@ 
\if@mtc@hints@given@true 
4736 \def\dosectlof@[#1]{% 
4737 \if@mtc@hints% 
4738 \if@dosectlof@used@ 
4739 \mtcPackageInfo[10045]{minitoc(hints)}% 
4740 \MessageBreak 
4741 \MessageBreak 
4742 \global\@dosectlof@used@true 
4743 \global\@mtc@hints@given@true 
4744 \fi 
4745 \fi 
4746 \fi 
4747 \global\@dosectlof@used@true 
4748 \if #1\let\df@stilf\e@sti% 
4749 \else\if #1n\let\df@stilf\n@sti% 
4750 \if #1c\let\df@stilf\c@sti% 
4751 \else\if #1l\let\df@stilf\l@sti% 
4752 \else\if #1r\let\df@stilf\r@sti% 
4753 \fi\fi\fi\fi\fi% 
4754 \@@dosectlof

We treat the optional argument of `\dosectlot` (it becomes the default position for titles of sectlofs) and flag this macro as used; a hint detects any spurious invocation.

\dosectlot% 
\if@mtc@hints% 
\@dosectlot@used@ 
\if@mtc@hints@given@true 
4755 \def\dosectlot@[#1]{% 
4756 \if@mtc@hints% 
4757 \if@dosectlot@used@ 
4758 \mtcPackageInfo[10045]{minitoc(hints)}% 
4759 \MessageBreak 
4760 \MessageBreak 
4761 \global\@dosectlot@used@true 
4762 \global\@mtc@hints@given@true 
4763 \fi 
4764 \fi 
4765 \fi 
4766 \global\@dosectlot@used@true 
4767 \if #1\let\df@stilt\e@sti% 
4768 \else\if #1n\let\df@stilt\n@sti% 
4769 \if #1c\let\df@stilt\c@sti% 
4770 \else\if #1l\let\df@stilt\l@sti% 
4771 \else\if #1r\let\df@stilt\r@sti% 
4772 \fi\fi\fi\fi\fi% 
4773 \@@dosectlot
These macros invoke the \@dosect... commands to create the mini-table file, then close the file descriptor.

These macros invoke the \@dosect... commands to create the mini-table file, then close the file descriptor.

\STC@next Processing the next entry in the list and remove it from the head of the list:
\STC@list
\STC@loop
\STC@toc
\STC@list
\STC@explist
\STC@contentsline The macro \STC@contentsline analyses the lines read from the TOC file. If \section is found, the stc counter is incremented and a new secttoc file is created.
\section{thetc}
\mTC@WriteContentsLine
\ifmtc@longext@ We test if long or short extensions are used, to build the name of the mini-table file, then open it:
\ifmtc@longext@
\mTC@WriteContentsLine
\else
\fi
\immediate\closeout\tf@mtc
\immediate\openout\tf@mtc=\stcname
The token register \mtc@toks is used to pass the entry to \MTC@WriteContentsline or \MTC@WriteCoffeeline. Now, we filter the relevant contents lines:

\mtc@toks\noexpand\leavevmode #2%%%%
\expandafter\ifx\csname #1\endcsname\coffee
\MTC@WriteCoffeeline{#1}{#3}%%
\fi
\expandafter\ifx\csname #1\endcsname\subsection
\MTC@WriteContentsline{#1}{stcSS}{#3}{#4}%%
\fi
\expandafter\ifx\csname #1\endcsname\subsubsection
\MTC@WriteContentsline{#1}{stcSSS}{#3}{#4}%%
\fi
\expandafter\ifx\csname #1\endcsname\paragraph
\MTC@WriteContentsline{#1}{stcP}{#3}{#4}%%
\fi
\expandafter\ifx\csname #1\endcsname\subparagraph
\MTC@WriteContentsline{#1}{stcSP}{#3}{#4}%%
\fi

A starred section terminates the current section and creates a new secttoc file:

\expandafter\ifx\csname #1\endcsname\starsection
\stepcounter{stc}%%
\gdef\thestc{\arabic{stc}}%%
\if@mtc@longext@%%
\mtcPackageInfo[I0033]{minitoc}{Writing\space\jobname.stc\thestc}%%
\def\stcname{\jobname.stc\thestc}%%
\else
\mtcPackageInfo[I0033]{minitoc}{Writing\space\jobname.S\thestc}%%
\def\stcname{\jobname.S\thestc}%%
\fi
\immediate\closeout\tf@mtc
\immediate\openout\tf@mtc=\stcname

We process the entries for starred sectioning commands:

\expandafter\ifx\csname #1\endcsname\starsubsection
\MTC@WriteContentsline{#1}{stcSS}{#3}{#4}%%
\fi
\expandafter\ifx\csname #1\endcsname\starsubsubsection
\MTC@WriteContentsline{#1}{stcSSS}{#3}{#4}%%
\fi
\expandafter\ifx\csname #1\endcsname\starparagraph
\MTC@WriteContentsline{#1}{stcP}{#3}{#4}%%
\fi
\expandafter\ifx\csname #1\endcsname\starsubparagraph
\MTC@WriteContentsline{#1}{stcSP}{#3}{#4}%%
\fi
\MTC@WriteContentsline{#1}{stcSP}{#3}{#4}\i
\STC@explist

\STC@loop
\STC@toc
\STC@read

\STC@read

\STC@test

\STC@test
\STC@contentsline
\mtc@string
\STC@list
\STC@toc
\STC@read
\sectend
\sectbegin
\else\ifx#1\@input
Commented code of the \texttt{minitoc} package

\begin{verbatim}
\edef\STC@toc{
\edef\STC@list{#2\relax}
\else\ifx#1\sectend
\immediate\closeout\tf@mtc
\immediate\openout\tf@mtc=\jobname.mtc
\else\ifx#1\sectbegin
\addtocounter{stc}{-1}\i\fi\fi\fi
\ifeof\@inputcheck\expandafter\STC@toc
\else\expandafter\STC@read\fi}
\SLF@next Processing the next entry in the list and remove it from the head of the list:
\SLF@list
\SLF@loop
\def\SLF@next#1\relax#2\{}\{%
\edef\SLF@list{#2}\
\SLF@loop{#1}
\SLF@lof Check if the list is empty:
\SLF@list
\SLF@explist
\def\SLF@lof{%\ifx\SLF@list\@empty\else\expandafter\SLF@explist\fi}
\SLF@contentsline The macro \texttt{\SLF@contentsline} analyses the lines read from the \texttt{LOF} file. If \texttt{\section} is found, the \texttt{stc} counter is incremented and a new \texttt{sectlof} file is created.
\section
\the\stc
\tf@mtc
\slfname
\MTC@WriteContentsLine
\def\SLF@contentsline#1#2#3#4{%\gdef\thestc{\arabic{stc}}\expandafter\ifx\csname #1\endcsname\xsect\stepcounter{stc}\if@mtc@longext@\slfname
We test if long or short extensions are used, to build the name of the mini-table file, then open it:
\if@mtc@longext@
\slfname
\def\mtcPackageInfo[I0033][minitoc]{%\Writing\space\jobname.slfl\thestc}%
\def\slfname{\jobname.slfl\thestc}
\else
\mtcPackageInfo[I0033][minitoc]{%\Writing\space\jobname.H\thestc}%
\def\slfname{\jobname.H\thestc}
\fi
\immediate\closeout\tf@mtc
\immediate\openout\tf@mtc=\slfname
\fi
\end{verbatim}
\mtc@toks
\MTC@WriteContentsline
\figure
\subfigure

The token register \mtc@toks is used to pass the entry to \MTC@WriteContentsline. Now, we filter the relevant contents lines:

\mtc@toks{\noexpand\leavevmode #2}%
\expandafter\ifx\csname #1\endcsname\figure
\MTC@WriteContentsline{#1}{slf}{#3}{#4}%
\fi
\expandafter\ifx\csname #1\endcsname\subfigure
\MTC@WriteContentsline{#1}{slfS}{#3}{#4}%
\fi
}

\SLF@explist
\SLF@next
\SLF@list

The loop to read the lines of the LOF file; expands the list of entries and call \SLF@next to process the first one.

\SLF@loop#1{\openin\@inputcheck#1\relax
\ifeof\@inputcheck
\mtcPackageWarning[W0014]{minitoc}{No file #1 \MessageBreak \expandafter\SLF@lof}
\else
\mtcPackageInfo[I0036]{minitoc}{PREPARING SECTLOFS FROM #1}\expandafter\SLF@read\fi}

\SLF@read

Read the next entry of the .lof file.

\SLF@test
\SLF@line

The ..... make sure that \SLF@test has enough arguments:

\SLF@test{\read\@inputcheck to\SLF@line\expandafter\SLF@test\SLF@line.....\SLF@%}

\SLF@test
\SLF@contentsline
\sectend
\sectbegin

The \SLF@test macro finds the “interesting” commands in the LOF file, mainly to delimit sections;
[9] — Commented code of the minitoc package

4919 \ifx\contentsline
4920 \let\mtc@string\string
4921 \SLF@contentsline{#2}{#3}{#4}{#5}%
4922 \let\mtc@string\relax
4923 \else\ifx#1\@input
4924 \edef\SLF@list{\SLF@list#2\relax}%
4925 \else\ifx#1\sectend
4926 \immediate\closeout\tf@mtc
4927 \immediate\openout\tf@mtc=\jobname.mtc
4928 \else\ifx#1\sectbegin
4929 \addtocounter{stc}{-1}%
4930 \fi\fi\fi\fi
4931 \ifeof\@inputcheck\expandafter\SLF@lof
4932 \else\expandafter\SLF@read\fi}%
4933 \SLT@next
4934 \SLT@list
4935 \SLT@loop
4936 \def\SLT@next#1\relax#2\{%
4937 \edef\SLT@list{#2}%
4938 \SLT@loop{#1}}%
4939 \SLT@lot
4940 \SLT@list
4941 \SLT@explist
4942 \def\SLT@contentsline#1#2#3#4{%
4943 \gdef\thestc{\arabic{stc}}%
4944 \expandafter\ifx\csname #1\endcsname\xsect
4945 \stepcounter{stc}%
4946 \if@mtc@longext@
4947 \mtcPackageInfo[10033]{minitoc}\%
4948 \{Writing\space\jobname.slt\thestc}\%
4949 \def\sltname{\jobname.slt\thestc}\%
4950 \else
4951 \mtcPackageInfo[10033]{minitoc}\%
4952 \{Writing\space\jobname.V\thestc}\%
4953 \def\sltname{\jobname.V\thestc}\%
4954 \fi
4955 \SLT@contentsline
4956 \section
4957 \thestc
4958 \tf@mtc
4959 \sltname
4960 \MTC@WriteContentsLine
4961 \expandafter\ifx\csname #1\endcsname\xsect
4962 \stepcounter{stc}%
4963 \if0\mtc@longext%
4964 \mtcPackageInfo[10033]{minitoc}\%
4965 \{Writing\space\jobname.slt\thestc}\%
4966 \def\sltname{\jobname.slt\thestc}\%
4967 \else
4968 \mtcPackageInfo[10033]{minitoc}\%
4969 \{Writing\space\jobname.V\thestc}\%
4970 \def\sltname{\jobname.V\thestc}\%
4971 \fi
4972 \SLT@list
4973 \SLT@loop
4974 \SLT@lot
4975 \SLT@explist
4976 \SLT@contentsline
4977 \section
4978 \thestc
4979 \tf@mtc
4980 \sltname
4981 \MTC@WriteContentsLine
4982 \expandafter\ifx\csname #1\endcsname\xsect
4983 \stepcounter{stc}%
4984 \if0\mtc@longext%
4985 \mtcPackageInfo[10033]{minitoc}\%
4986 \{Writing\space\jobname.slt\thestc}\%
4987 \def\sltname{\jobname.slt\thestc}\%
4988 \else
4989 \mtcPackageInfo[10033]{minitoc}\%
4990 \{Writing\space\jobname.V\thestc}\%
4991 \def\sltname{\jobname.V\thestc}\%
4992 \fi
The token register `\mtc@toks` is used to pass the entry to `\MTC@WriteContentsline`. Now, we filter the relevant contents lines:

```latex
\mtc@toks{\noexpand\leavevmode #2}\
\expandafter\ifx\csname #1\endcsname\table
\MTC@WriteContentsline{#1}{slt}{#3}{#4}\
\fi
\expandafter\ifx\csname #1\endcsname\subtable
\MTC@WriteContentsline{#1}{sltS}{#3}{#4}\
\fi
```

The loop to read the lines of the LOT file; expands the list of entries and call `\SLT@next` to process the first one.

```latex
\def\SLT@explist{\expandafter\SLT@next\SLT@list}\
```

If an entry is found, loop through line by line, looking for interesting entries. Otherwise, process the next entry in the list.

```latex
\def\SLT@loop#1{\openin\@inputcheck#1\relax
\ifeof\@inputcheck
\mtcPackageWarning[W0015]{minitoc}{No file #1 MessageBreak}
\expandafter\SLT@lot
\else
\mtcPackageInfo[I0039]{minitoc}{PREPARING SECTLOTS FROM #1 MessageBreak}
\expandafter\SLT@read\fi}
```

Read the next entry of the .lot file.

```latex
\def\SLT@read{\\read\@inputcheck to\SLT@line
\expandafter\SLT@test\SLT@line.....\SLT@}%
```

The ..... make sure that `\SLT@test` has enough arguments:
9.62 End of section-level commands

We terminate the else branch of the test \@ifundefined{section}, the true branch of the test \@ifundefined{chapter} and add an empty else branch to that test:

\mtcprepare\@ifnextchar\mtcprepare@\@ifundefined\IfFileExists\jobname\doparttoc\[#1\]{}\%\def\mtcprepare\@ifnextchar[\@ifnextchar\mtcprepare@\@ifundefined\IfFileExists\jobname\dopartlof\[#1\]{}\%\IfFileExists\jobname\dopartlot\[#1\]{}\%\@ifundefined{chapter}{%\@ifundefined{section}{}%\IfFileExists\jobname\dosecttoc\[#1\]{}\%\IfFileExists\jobname\dosectlof\[#1\]{}\%\IfFileExists\jobname\dosectlot\[#1\]{}\%\mtcPackageInfo

9.63 The \mtcprepare command

This command tests the availability of the \do... minitoc preparation commands and of the contents files, then calls as much as possible of these preparation commands. A hint is given.
9.64 Use with `nofiles`

In case the document uses the `nofiles` command (in its preamble), the auxiliary files for the mini-tables should not be overwritten by the preparation commands, so these ones must be just faked; as these commands may have an optional argument, they will be faked using the internal \LaTeX macro \@ifnextchar (to get the optional argument) and the new utility command \gobbleopt. Problem signaled by Andreas Deininger.

\begin{verbatim}
\def\gobbleopt[#1]{\relax}
\end{verbatim}

A test is placed in a `\AtBeginDocument` and gives a warning if `nofiles` is used:

\begin{verbatim}
\AtBeginDocument{\if@filesw\relax\else
  \mtcPackageWarningNoLine[\@gobble]{minitoc}{--- You have used the `\string\nofiles` space command in your preamble; all preparation commands in the body of the document will be ignored}
\fi}
\end{verbatim}

Since `nofiles` has been used, we must disable all the preparation commands:
9.65 Necessary \l@... commands

\ifundefined{section}{}{\let\l@listof\l@section}
\ifundefined{chapter}{}{\let\l@listof\l@chapter}
\ifundefined{part}{}{\let\l@starpart\l@part}
\ifundefined{chapter}{}{\let\l@starchapter\l@chapter}
\ifundefined{section}{}{\let\l@starsection\l@section}
\ifundefined{subsection}{}{\let\l@starsubsection\l@subsection}
\ifundefined{subsubsection}{}{\let\l@starsubsubsection\l@subsubsection}
\ifundefined{paragraph}{}{\let\l@starparagraph\l@paragraph}
\ifundefined{subparagraph}{}{\let\l@starsubparagraph\l@subparagraph}
The horizontal rules and their default values

We define here the various commands to activate or inhibit the horizontal rules in the various kinds of mini-tables. Each such command is an indirect definition of the corresponding horizontal rule. The rules are .4 pt high horizontal rules. We begin with rules for mini-tables of contents.

\def\nopcrule{\let\ptc@rule\relax}
\def\nomtrule{\let\mtc@rule\relax}
\def\nostcrule{\let\stc@rule\relax}
\def\ptcrule{\def\ptc@rule{\kern-3\p@ \hrule width \columnwidth \kern2.6\p@}}
\def\mtcrule{\def\mtc@rule{\kern-3\p@ \hrule width \columnwidth \kern2.6\p@}}
\def\stcrule{\def\stc@rule{\kern-3\p@ \hrule width \columnwidth \kern2.6\p@}}
\def\ptc@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}
\def\mtc@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}
\def\stc@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}

Then, the rules for mini-lists of figures:

\def\noplfrule{\let\plf@rule\relax}
\def\nomlfrule{\let\mlf@rule\relax}
\def\noslfrule{\let\slf@rule\relax}
\def\plfrule{\def\plf@rule{\kern-3\p@ \hrule width \columnwidth \kern2.6\p@}}
\def\mlfrule{\def\mlf@rule{\kern-3\p@ \hrule width \columnwidth \kern2.6\p@}}
\def\slfrule{\def\slf@rule{\kern-3\p@ \hrule width \columnwidth \kern2.6\p@}}
\def\plf@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}
\def\mlf@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}
\def\slf@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}

Then, the rules for mini-lists of tables:

\def\nopltrule{\let\plt@rule\relax}
\def\nomltrule{\let\mlt@rule\relax}
\def\nosltrule{\let\slt@rule\relax}
\def\pltrule{\def\plt@rule{\kern-3\p@ \hrule width \columnwidth \kern2.6\p@}}
\def\mltrule{\def\mlt@rule{\kern-3\p@ \hrule width \columnwidth \kern2.6\p@}}
\def\sltrule{\def\slt@rule{\kern-3\p@ \hrule width \columnwidth \kern2.6\p@}}
\def\plt@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}
\def\mlt@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}
\def\slt@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}
9.67 The \mtcset... commands

These commands have been introduced to build a nicer user interface, and each of them replaces numerous user commands, offering a rather unified and logical syntax.

9.67.1 Keywords for the \mtcset... commands

\@namedef \@nameuse

We define some common keywords for the \mtcset... commands. A keyword is created via the \@namedef – \@nameuse mechanism the following way:

\@namedef{mtc@family@name}{abbreviation}

where family is the name of a group of keywords relative to one or several \mtcset... macros, name is the keyword that the user gives as argument to the \mtcset... macro, and abbreviation is a string used to build the name of the macro effectively used. As some \mtcset... macros have several keyword parameters, this method can reduce the number of macros at the user level, at the cost of few keyword families.

\@namedef

We define a family (typetable) of keywords for the types of mini-tables:

\@namedef{mtc@typetable@parttoc}{ptc}\def\mtc@typetable@parttoc{ptc}
\@namedef{mtc@typetable@partlof}{plf}\def\mtc@typetable@partlof{plf}
\@namedef{mtc@typetable@partlot}{plt}\def\mtc@typetable@partlot{plt}
\@namedef{mtc@typetable@minitoc}{mtc}\def\mtc@typetable@minitoc{mtc}
\@namedef{mtc@typetable@minilof}{mlf}\def\mtc@typetable@minilof{mlf}
\@namedef{mtc@typetable@minilot}{mlt}\def\mtc@typetable@minilot{mlt}
\@namedef{mtc@typetable@secttoc}{stc}\def\mtc@typetable@secttoc{stc}
\@namedef{mtc@typetable@sectlof}{slf}\def\mtc@typetable@sectlof{slf}
\@namedef{mtc@typetable@sectlot}{slt}\def\mtc@typetable@sectlot{slt}

\@namedef

Then another family (typetitle) for the titles of the mini-tables:

\@namedef{mtc@typetitle@parttoc}{pti}\def\mtc@typetitle@parttoc{pti}
\@namedef{mtc@typetitle@partlof}{pti}\def\mtc@typetitle@partlof{pti}
\@namedef{mtc@typetitle@partlot}{pti}\def\mtc@typetitle@partlot{pti}
\@namedef{mtc@typetitle@minitoc}{mti}\def\mtc@typetitle@minitoc{mti}
\@namedef{mtc@typetitle@minilof}{mti}\def\mtc@typetitle@minilof{mti}
\@namedef{mtc@typetitle@minilot}{mti}\def\mtc@typetitle@minilot{mti}
\@namedef{mtc@typetitle@secttoc}{sti}\def\mtc@typetitle@secttoc{sti}
\@namedef{mtc@typetitle@sectlof}{sti}\def\mtc@typetitle@sectlof{sti}
\@namedef{mtc@typetitle@sectlot}{sti}\def\mtc@typetitle@sectlot{sti}

\@namedef

11The general concept of the \mtcset... commands was proposed by Benjamin BAYART.
We define a family \(\text{YN}\) of keywords to recognize the keywords “off” and “on”, with their many synonyms\(^{12}\) and meaning false or true\(^{13}\):

\[
\begin{align*}
\@namedef{mtc@YN@off}{no} & \def \mtc@YN@off{no} \\
\@namedef{mtc@YN@OFF}{no} & \def \mtc@YN@OFF{no} \\
\@namedef{mtc@YN@no}{no} & \def \mtc@YN@no{no} \\
\@namedef{mtc@YN@NO}{no} & \def \mtc@YN@NO{no} \\
\@namedef{mtc@YN@n}{no} & \def \mtc@YN@n{no} \\
\@namedef{mtc@YN@N}{no} & \def \mtc@YN@N{no} \\
\@namedef{mtc@YN@false}{no} & \def \mtc@YN@false{no} \\
\@namedef{mtc@YN@FALSE}{no} & \def \mtc@YN@FALSE{no} \\
\@namedef{mtc@YN@faux}{no} & \def \mtc@YN@faux{no} \\
\@namedef{mtc@YN@FAUX}{no} & \def \mtc@YN@FAUX{no} \\
\@namedef{mtc@YN@f}{no} & \def \mtc@YN@f{no} \\
\@namedef{mtc@YN@F}{no} & \def \mtc@YN@F{no} \\
\@namedef{mtc@YN@NON}{no} & \def \mtc@YN@NON{no} \\
\@namedef{mtc@YN@non}{no} & \def \mtc@YN@non{no} \\
\@namedef{mtc@YN@0}{no} & \expandafter\def\csname mtc@YN@0\endcsname{no} \\
\@namedef{mtc@YN@-}{no} & \expandafter\def\csname mtc@YN@-\endcsname{no} \\
\@namedef{mtc@YN@on}{\} & \def \mtc@YN@on{} \\
\@namedef{mtc@YN@ON}{\} & \def \mtc@YN@ON{} \\
\@namedef{mtc@YN@yes}{\} & \def \mtc@YN@yes{} \\
\@namedef{mtc@YN@YES}{\} & \def \mtc@YN@YES{} \\
\@namedef{mtc@YN@y}{\} & \def \mtc@YN@y{} \\
\@namedef{mtc@YN@Y}{\} & \def \mtc@YN@Y{} \\
\@namedef{mtc@YN@true}{\} & \def \mtc@YN@true{} \\
\@namedef{mtc@YN@TRUE}{\} & \def \mtc@YN@TRUE{} \\
\@namedef{mtc@YN@t}{\} & \def \mtc@YN@t{} \\
\@namedef{mtc@YN@T}{\} & \def \mtc@YN@T{} \\
\@namedef{mtc@YN@vrai}{\} & \def \mtc@YN@vrai{} \\
\@namedef{mtc@YN@VRAI}{\} & \def \mtc@YN@VRAI{} \\
\@namedef{mtc@YN@v}{\} & \def \mtc@YN@v{} \\
\@namedef{mtc@YN@V}{\} & \def \mtc@YN@V{} \\
\@namedef{mtc@YN@OUI}{\} & \def \mtc@YN@OUI{} \\
\@namedef{mtc@YN@oui}{\} & \def \mtc@YN@oui{} \\
\@namedef{mtc@YN@O}{\} & \def \mtc@YN@O{} \\
\@namedef{mtc@YN@o}{\} & \def \mtc@YN@o{} \\
\@namedef{mtc@YN@1}{\} & \expandafter\def\csname mtc@YN@1\endcsname{} \\
\@namedef{mtc@YN@+}{\} & \expandafter\def\csname mtc@YN@+\endcsname{} \\
\@namedef{mtc@YN@11}{\} & \expandafter\def\csname mtc@YN@11\endcsname{} \\
\@namedef{mtc@YN@+}{\} & \expandafter\def\csname mtc@YN@+\endcsname{} \\
\end{align*}
\]

\(^{12}\)This (deliberately extreme) case shows the easyness for creating synonyms of frequently used keywords. Note also that when a keyword contains a non-letter character, we must use a hack with \texttt{\expandafter \csname ... \endcsname}.

\(^{13}\)O and o are the letter O, \(\theta\) is the zero digit.
9.67.2 The \mtcsetfont command

\@namedef We define the sectionning level keywords (note that part is not a member of this family (sectlevel), because no contents line for a part can appear in a mini-table, part being the highest sectionning level); "*" represents “any level”, and is used to set the global default font for a given kind of mini-table.

\mtcsetfont The \mtcsetfont command has the following syntax:

\mtcsetfont{mini-table}{level-name}{font commands}

The mini-table type is a keyword like minitoc, the level-name is a sectionning level like subsection (no backslash). The font commands are a font specification, using NFSS [291] basic commands usually.

\if\mtc@setfont@ First, we declare a flag, set true:

\if\mtc@setfont@ 5149 newif\if\mtc@setfont@\@mtc@setfont@true

\mtcsetfont Then, we begin the command, which has three arguments:

\mtc@mta@abbrev \if\mtc@setfont@ The two first arguments of this command are keywords. They must be translated into the effective strings. We process the first argument, a keyword from the typetable family. The result is stored in \mtc@mta@abbrev. Example: if #1 is minitoc, we get mtc.

\if\mtc@setfont@ \@namedefuse \def\mtc@mta@abbrev{X} \@mtc@setfont@false
\expandafter\ifx\csname mtc@typetable@#1\endcsname\relax
The second argument, a keyword from the family sectlevel, is processed the same way and the result is stored into a macro \mtc@level@abbrev. Example: if #2 is the subparagraph keyword, we get SP.

Then, we construct the effective macro to be applied:

Example: if #1 is minitoc and #2 is subsection, we get mtcSSfont, which is the name of the command for the font of a subsection entry in a minitoc (the backslash is missing, but we will use a \csname ... \endcsname pair to apply the constructed command).
But all combinations are not legal (the level of the entry must be lower than the level of the mini-table, and the kind\(^{14}\) of the entry must be consistent with that of the mini-table), so we must test. Special care must be taken for testing via internal defined commands (quarks) with \verb+\@mtcqk+ at the end of their names.

\begin{verbatim}
\if\@mtc@setfont\true
\def\parttoc@mtcqk{parttoc@mtcqk}
\def\minitoc@mtcqk{minitoc@mtcqk}
\def\secttoc@mtcqk{secttoc@mtcqk}
\def\partlof@mtcqk{partlof@mtcqk}
\def\minilof@mtcqk{minilof@mtcqk}
\def\sectlof@mtcqk{sectlof@mtcqk}
\def\partlot@mtcqk{partlot@mtcqk}
\def\minilot@mtcqk{minilot@mtcqk}
\def\sectlot@mtcqk{sectlot@mtcqk}
\def\part@mtcqk{part@mtcqk}
\def\chapter@mtcqk{chapter@mtcqk}
\def\appendix@mtcqk{appendix@mtcqk}
\def\section@mtcqk{section@mtcqk}
\def\subsection@mtcqk{subsection@mtcqk}
\def\subsubsection@mtcqk{subsubsection@mtcqk}
\def\paragraph@mtcqk{paragraph@mtcqk}
\def\subparagraph@mtcqk{subparagraph@mtcqk}
\def\figure@mtcqk{figure@mtcqk}
\def\table@mtcqk{table@mtcqk}
\def\subfigure@mtcqk{subfigure@mtcqk}
\def\subtable@mtcqk{subtable@mtcqk}
\@mtc@setfont\true
\expandafter\if\csname #1@mtcqk\endcsname\parttoc@mtcqk\relax
\expandafter\if\csname #2@mtcqk\endcsname\figure@mtcqk\relax\@mtc@setfont\false\fi
\expandafter\if\csname #2@mtcqk\endcsname\subfigure@mtcqk\relax\@mtc@setfont\false\fi
\expandafter\if\csname #2@mtcqk\endcsname\table@mtcqk\relax\@mtc@setfont\false\fi
\fi
\expandafter\if\csname #1@mtcqk\endcsname\partlof@mtcqk\relax
\expandafter\if\csname #2@mtcqk\endcsname\table@mtcqk\relax\@mtc@setfont\false\fi
\expandafter\if\csname #2@mtcqk\endcsname\subtable@mtcqk\relax\@mtc@setfont\false\fi
\fi
\expandafter\if\csname #1@mtcqk\endcsname\minitoc@mtcqk\relax
\expandafter\if\csname #2@mtcqk\endcsname\part@mtcqk\relax\@mtc@setfont\false\fi
\expandafter\if\csname #2@mtcqk\endcsname\chapter@mtcqk\relax\@mtc@setfont\false\fi
\expandafter\if\csname #2@mtcqk\endcsname\appendix@mtcqk\relax\@mtc@setfont\false\fi
\expandafter\if\csname #2@mtcqk\endcsname\figure@mtcqk\relax\@mtc@setfont\false\fi
\expandafter\if\csname #2@mtcqk\endcsname\subfigure@mtcqk\relax\@mtc@setfont\false\fi
\expandafter\if\csname #2@mtcqk\endcsname\table@mtcqk\relax\@mtc@setfont\false\fi
\expandafter\if\csname #2@mtcqk\endcsname\subtable@mtcqk\relax\@mtc@setfont\false\fi
\fi
\expandafter\if\csname #1@mtcqk\endcsname\minilof@mtcqk\relax
\expandafter\if\csname #2@mtcqk\endcsname\part@mtcqk\relax\@mtc@setfont\false\fi
\expandafter\if\csname #2@mtcqk\endcsname\figure@mtcqk\relax\@mtc@setfont\false\fi
\fi
\end{verbatim}

\(^{14}\)”Kind” being sectioning, (sub-)figure, or (sub-)table.
If the combination is legal, we apply it, i.e., we redefine the meaning of the constructed macro with the sequence of commands given as third argument of \mtcsetfont and we log that event (we store the third argument in a token register to print it \emph{verbatim}); if the combination is not legal, an error message is displayed.

\input{minitoc}
9.67.3 The \mtcsettitlefont command

The \mtcsettitlefont command is very similar to the \mtcsetfont command. Its syntax is almost identical:

\mtcsettitlefont{mini-table}{font commands}

The \textit{mini-table} type is a keyword like \texttt{minitoc}. The \textit{font commands} are a font specification, using NFSS \cite{291} basic commands usually. The difference is the absence of the second keyword argument, because the \textit{font commands} will be applied to the title of each mini-table of the given kind.

First, we declare a flag, set true:

\newif\@mtc@settitlefont\@mtc@settitlefont@true

And we begin the definition of the \mtcsettitlefont command, which has two arguments:

\newcommand{\mtcsettitlefont}[2]{

We process the first argument, a keyword of the \texttt{typetitle} family, then the result is stored into \texttt{\mtc@mtatf@abbrev}:

\def{\mtc@mtatf@abbrev}(X)
\@mtc@settitlefont@true
\expandafter{\if\csname mtc@typetitle@#1\endcsname\relax
\@mtc@settitlefont@false
\def{\mtc@mtatf@abbrev}(X)
\mtcPackageError{E0022}{minitoc}%
(\string\mtcsettitlefont \texttt{space has a wrong first argument
\MessageBreak
\@mtc@settitlefont@false
\MessageBreak
\@mtc@settitlefont@false
\MessageBreak
It should be a mini-table type}
## 9.67.4 The \texttt{\mtcsettitle} command

\texttt{\mtcsettitle} This command is very similar to the \texttt{\mtcsettitlefont} command. Its syntax is almost identical:

\begin{verbatim}
\mtcsettitle[\textit{mini-table}]{text}
\end{verbatim}

The \textit{mini-table} type is a keyword like \texttt{minitoc}. The \textit{text} is the text for a mini-table title.

\begin{verbatim}
\if@mtc@settitle@ \newif\if@mtc@settitle@\@mtc@settitle@true
\end{verbatim}
\mtcsettitle

Then we define the \mtcsettitle command, which has two arguments:

\newcommand{\mtcsettitle}[2]{
\mtc@mtati@abbrev
\if@mtc@settitle@
\@nameuse
We process the first argument, a keyword of the typetable family. The result is stored in
\mtc@mtati@abbrev:
\def{\mtc@mtati@abbrev}{X}
\@mtc@settitle@true
\expandafter{\ifx{\csname mtc@typetable@#1\endcsname}{relax}
\@mtc@settitle@false
\def{\mtc@mtati@abbrev}{X}
\mtcPackageError{E0021}{minitoc}%
{\string\mtcsettitle has a wrong first argument (\texttt{#1}).
\MessageBreak
It should be a mini-table type
\MessageBreak
\{Correct the source code.
\MessageBreak
Type <\texttt{return}> and rerun \LaTeX}%
\else
\edef{\mtc@mtati@abbrev}{\@nameuse{mtc@typetable@\texttt{#1}}}
\fi
}

\if@mtc@settitle@
\mtc@tmpti@name
\mtc@mtati@abbrev
\mtc@toks
And we construct the name of the effective macro and apply it:
\def{\mtc@tmpti@name}{\mtc@mtati@abbrev title}
\mtc@toks{#2}
\mtcPackageInfo{I0017}{minitoc}%
{\string\mtcsettitle redefines the macro "\mtc@tmpti@name" as "the\mtc@toks"%}
\expandafter{\edef{\csname\mtc@tmpti@name\endcsname}{\the\mtc@toks}}
\else
\mtcPackageError{E0033}{minitoc}%
{The macro \string\mtcsettitle space uses an illegal type of table (#1).%}
\MessageBreak
{Correct the source code.
\MessageBreak
Type <\texttt{return}> and rerun \LaTeX}{\relax}
\fi
}
9.67.5 The \mtcsetformat command

\@namedef We define first the keywords (family formatparam) for the three formatting parameters that this command can alter:

5373 \@namedef{mtc@formatparam@dotinterval}{dotsep}%
5374 \def\mtc@arg@dotinterval{dotsep}
5375 \@namedef{mtc@formatparam@tocrightmargin}{tocrmarg}%
5376 \def\mtc@arg@tocrightmargin{tocrmarg}
5377 \@namedef{mtc@formatparam@pagenumwidth}{pnumwidth}%
5378 \def\mtc@arg@pagenumwidth{pnumwidth}
5379 \@namedef{mtc@arg@numwidth}{numwidth} \textit{not yet available}
5380 \@namedef{mtc@arg@numwidth}{numwidth} \textit{not yet available}

\AtBeginDocument The \mtcsetformat command needs an initialization to be done at the beginning of the document, to set the defaults values of the formatting parameters:

5381 \AtBeginDocument{%
\@pnumwidth\ptcpnumwidth\mtcpnumwidth\stcpnumwidth\plfpnumwidth\mlfpnumwidth\slfpnumwidth\pltpnumwidth\mltpnumwidth\sltpnumwidth
\@tocrmarg\ptctocrmarg\mtctocrmarg\stctocrmarg\plftocrmarg\mlftocrmarg\slftocrmarg\plttocrmarg\mlttocrmarg\slttocrmarg
\@tocrmarg We take, if possible, the default value of \@pnumwidth for each type of mini-tables:

5382 \@ifundefined{ptcpnumwidth}{\let\ptcpnumwidth\@pnumwidth}{}%
5383 \@ifundefined{stcpnumwidth}{\let\stcpnumwidth\@pnumwidth}{}%
5384 \@ifundefined{plfpnumwidth}{\let\plfpnumwidth\@pnumwidth}{}%
5385 \@ifundefined{mlfpnumwidth}{\let\mlfpnumwidth\@pnumwidth}{}%
5386 \@ifundefined{slfpnumwidth}{\let\slfpnumwidth\@pnumwidth}{}%
5387 \@ifundefined{pltpnumwidth}{\let\pltpnumwidth\@pnumwidth}{}%
5388 \@ifundefined{mltpnumwidth}{\let\mltpnumwidth\@pnumwidth}{}%
5389 \@ifundefined{sltpnumwidth}{\let\sltpnumwidth\@pnumwidth}{}%

\@tocrmarg We take, if possible, the default value of \@tocrmarg for each type of mini-tables:

5391 \@ifundefined{ptctocrmarg}{\let\ptctocrmarg\@tocrmarg}{}%
We take, if possible, the default value of `@dotsep` for each type of mini-tables:

```latex
\ifundefined{ptcdotsep}{\let\ptcdotsep\@dotsep}{}%
\ifundefined{mtcdotsep}{\let\mtcdotsep\@dotsep}{}%
\ifundefined{stcdotsep}{\let\stcdotsep\@dotsep}{}%
\ifundefined{plfdotsep}{\let\plfdotsep\@dotsep}{}%
\ifundefined{mlfdotsep}{\let\mlfdotsep\@dotsep}{}%
\ifundefined{slfdotsep}{\let\slfdotsep\@dotsep}{}%
\ifundefined{pltdotsep}{\let\pltdotsep\@dotsep}{}%
\ifundefined{mltdotsep}{\let\mltdotsep\@dotsep}{}%
\ifundefined{sltdotsep}{\let\sltdotsep\@dotsep}{}%
```

And we terminate the `\AtBeginDocument` block:

```latex
)}%```

The executive part is done via the following macros, which are invoked in the `mtc@verse`-like environments for each kind of mini-table. These commands activate the values recorded by `mtcsetformat`.

```latex
\def\ptc@setform{\let\@pnumwidth\ptcpnumwidth\relax
\let\@tocrmarg\ptctocrmarg\relax
\let\@dotsep\ptcdotsep\relax
}
```

```latex
\def\mtc@setform{\let\@pnumwidth\mtcpnumwidth\relax
\let\@tocrmarg\mtctocrmarg\relax
\let\@dotsep\mtcdotsep\relax
}
```

```latex
\def\stc@setform{\let\@pnumwidth\stcpnumwidth\relax
\let\@tocrmarg\stctocrmarg\relax
\let\@dotsep\stcdotsep\relax
}
```

The `\ptc@setform` macro is invoked in `ptc@verse` to set format parameters:

```latex
\def\ptc@setform[%
\let\@pnumwidth\ptcpnumwidth\relax
\let\@tocrmarg\ptctocrmarg\relax
\let\@dotsep\ptcdotsep\relax
}
```

The `\mtc@setform` macro is invoked in `mtc@verse` to set format parameters:

```latex
\def\mtc@setform[%
\let\@pnumwidth\mtcpnumwidth\relax
\let\@tocrmarg\mtctocrmarg\relax
\let\@dotsep\mtcdotsep\relax
}
```

The `\stc@setform` macro is invoked in `stc@verse` to set format parameters:

```latex
\def\stc@setform[%
\let\@pnumwidth\stcpnumwidth\relax
\let\@tocrmarg\stctocrmarg\relax
\let\@dotsep\stcdotsep\relax
}
```
\texttt{\texttt{plf@setform}} \texttt{ptc@verse}  

The \texttt{plf@setform} macro is invoked in \texttt{ptc@verse} to set format parameters:

\begin{verbatim}
\def\plf@setform{%
\let\@pnumwidth\plfpnumwidth\relax
\let\@tocrmarg\plftocrmarg\relax
\let\@dotsep\plfdotsep\relax
}
\end{verbatim}

\texttt{\texttt{mlf@setform}} \texttt{mtc@verse}  

The \texttt{mlf@setform} macro is invoked in \texttt{mtc@verse} to set format parameters:

\begin{verbatim}
\def\mlf@setform{%
\let\@pnumwidth\mlfpnumwidth\relax
\let\@tocrmarg\mlftocrmarg\relax
\let\@dotsep\mlfdotsep\relax
}
\end{verbatim}

\texttt{\texttt{slf@setform}} \texttt{stc@verse}  

The \texttt{slf@setform} macro is invoked in \texttt{stc@verse} to set format parameters:

\begin{verbatim}
\def\slf@setform{%
\let\@pnumwidth\slfpnumwidth\relax
\let\@tocrmarg\slftocrmarg\relax
\let\@dotsep\slfdotsep\relax
}
\end{verbatim}

\texttt{\texttt{plt@setform}} \texttt{ptc@verse}  

The \texttt{plt@setform} macro is invoked in \texttt{ptc@verse} to set format parameters:

\begin{verbatim}
\def\plt@setform{%
\let\@pnumwidth\pltpnumwidth\relax
\let\@tocrmarg\plttocrmarg\relax
\let\@dotsep\pltdotsep\relax
}
\end{verbatim}

\texttt{\texttt{mlt@setform}} \texttt{mtc@verse}  

The \texttt{mlt@setform} macro is invoked in \texttt{mtc@verse} to set format parameters:

\begin{verbatim}
\def\mlt@setform{%
\let\@pnumwidth\plfpnumwidth\relax
\let\@tocrmarg\plftocrmarg\relax
\let\@dotsep\plfdotsep\relax
}
\end{verbatim}
The `\slt@setform` macro is invoked in `stc@verse` to set format parameters:

```latex
\def\slt@setform{\let\@pnumwidth\plfpnumwidth\relax
\let\@tocrmarg\plftocrmarg\relax
\let\@dotsep\plfdotsep\relax
}
```

We now define a flag and the `\mtcsetformat` command, with has the following syntax:

```latex
\mtcsetformat{\textit{mini-table}}{\textit{parameter-name}}{\textit{value}}
```

where `mini-table` is a keyword of the typetable family, `parameter-name` is a keyword of the formatparam family and `value`, the value of this parameter for the given kind of mini-table.

```latex
\newif\if\mtc@setformat@\@mtc@setformat@true
\newcommand{\mtcsetformat}[3]{\mtc@mtf@abbrev
\mtc@fparam@abbrev
\if\eqntext{\mtc@typetable@#1}{true}
\@mtc@setformat@false
\else \edef\mtc@mtf@abbrev{\@nameuse{mtc@typetable@#1}}\fi
\mtcPackageError[E0015]{minitoc}%
{\string\mtcsetformat \ space has a wrong first argument
MessageBreak
(#1).
MessageBreak
It should be a mini-table type
MessageBreak
(parttoc...sectlot)}%
{Correct the source code.
MessageBreak
Type \texttt{\textless return\textgreater} and rerun LaTeX}
\else \edef\mtc@mtf@abbrev{\@nameuse{mtc@typetable@#1}}\fi
\mtc@fparam@abbrev
```

Then we process the second argument and store the result into a macro `\mtc@fparam@abbrev`:


9.67.6 The \mtcsetpageminbers command

This command activates or inhibits page numbers in the mini-tables of a given kind. Its syntax is the following:

\mtcsetpageminbers{mini-table}{on|off}

where mini-table is a keyword for a kind of mini-table (parttoc, ... sectlot), or on and off a keyword to activate (on) or inhibit (off) the page numbers. on and off have many synonyms.
We define some flags:
\newif\if@mtc@setpagenumbers\@mtc@setpagenumbersfalse
\newif\if@mtc@spn@ok

\mtcsetpagenumbers We define the user-level macro. If the first argument is a star, we call the internal macro \mtcsetpagenumbers for each type of mini-table available; else, we call this internal macro only once, for the specified type of mini-table.
\newcommand{\mtcsetpagenumbers} [2] {%
\expandafter\ifx\csname #1\endcsname\*\relax
\@ifundefined{part}{}%
{\mtcsetpagenumbers@{parttoc}{#2}}
\mtcsetpagenumbers@{partlof}{#2}
\mtcsetpagenumbers@{partlot}{#2}
\@ifundefined{chapter}{}%
{\mtcsetpagenumbers@{minitoc}{#2}}
\mtcsetpagenumbers@{minilof}{#2}
\mtcsetpagenumbers@{minilot}{#2}
\@ifundefined{section}{}%
{\mtcsetpagenumbers@{secttoc}{#2}}
\mtcsetpagenumbers@{sectlof}{#2}
\mtcsetpagenumbers@{sectlot}{#2}
\else
\mtcsetpagenumbers@{#1}{#2}%
\fi%
%
Then the \mtcsetpagenumbers@ internal macro, with two arguments:
\newcommand{\mtcsetpagenumbers@} [2] {%

\mtc@mttpn@abbrev \mtc@pnsw@abbrev
We process the first argument, a keyword of the typetable family, and store the result in \mtc@mttpn@abbrev:
\def\mtc@mttpn@abbrev{X}
\@mtc@setpagenumbers@true
\def\mtc@pnsw@abbrev{X}
\def\mtc@mttpn@abbrev{X}
\mtcPackageError[E0017]{minitoc}%
{\string\mtcsetpagenumbers \space has a wrong first argument}\%
Then the second argument, a keyword of the YN family, and store the result into a macro \mtc@pnsw@abbrev. The name of the effective macro is built and the macro executed.

```latex
\if@mtc@spn@ok@
\mtc@pnsw@abbrev
\if@mtc@setpagenumbers@
\mtc@tmppn@name
\else
\edef\mtc@pnsw@abbrev{X}
\def\mtc@mttpn@abbrev{X}
\@mtc@setpagenumbers@false
\def\mtc@mttpn@abbrev{X}
\mtcPackageError{\string\mtcsetpagenumbers \space has a wrong second argument (#2)}{It should be a boolean value (0/1, yes/no, on/off, ...)}
\MessageBreak
Correct the source code.
\MessageBreak
Type <return> and rerun LaTeX}
\else
\edef\mtc@pnsw@abbrev{X}
\def\mtc@pnsw@abbrevX{X}
\def\mtc@noX{mtc@noX}
\def\mtc@tmppn@name{\mtc@pnsw@abbrev\mtc@mttpn@abbrev pagenumbers}
\expandafter\ifx\csname mtc@\mtc@pnsw@abbrev X\endcsname\mtc@noX
\mtcPackageInfo{Page numbers are inhibited}
\else
\mtcPackageInfo{Page numbers are activated}
\fi
\csname\mtc@tmppn@name\endcsname{}
\fi
```

[9] — Commented code of the minitoc package
9.67.7 The \mtcsetrules command

This macro is very similar to \mtcsetpagenumbers and its syntax is the same:

\mtcsetrules\{\textit{mini-table}\}\{\textit{on}\mid \textit{off}\}

where \textit{mini-table} is a keyword for a kind of mini-table (parttoc, ... sectlot), or \textit{on} and \textit{off} a keyword to activate (\textit{on}) or inhibit (\textit{off}) the horizontal rules. \textit{on} and \textit{off} have many synonyms.

Hence the code is similar.

\begin{verbatim}
\@ifundefined{\mtcsetrules}{}\% We define some flags:
\newif\if\mtc@setrules@ \@mtc@setrules@false
\newif\if\mtc@sru@ok@
\mtcsetrules\@ifundefined{\mtcsetrules}{}\% We define the user-level macro. If the first argument is a star, we call the internal macro \mtcsetrules@ for each type of mini-table available; else, we call this internal macro only once, for the specified type of mini-table.
\newcommand{\mtcsetrules@}[2]{}\%
\expandafter\ifx\csname #1\endcsname\*\relax
\@ifundefined{part}{}\%
\mtcsetrules@{parttoc}{#2}
\mtcsetrules@{partlof}{#2}
\mtcsetrules@{partlot}{#2}
\@ifundefined{chapter}{}\%
\mtcsetrules@{minitoc}{#2}
\mtcsetrules@{minilof}{#2}
\mtcsetrules@{minilot}{#2}
\@ifundefined{section}{}\%
\mtcsetrules@{secttoc}{#2}
\mtcsetrules@{sectlof}{#2}
\mtcsetrules@{sectlot}{#2}
\else
\mtcsetrules@{#1}{#2}\%
\fi
\end{verbatim}

\begin{verbatim}
\mtcsetrules@ \% Then the \mtcsetrules@ internal macro, which has two arguments:
\newcommand{\mtcsetrules@}[2]{}\%
\end{verbatim}
We process the first argument, a keyword of the `typetable` family and store the result in a macro `\mtc@mttru@abbrev`:

```latex
\def\mtc@mttru@abbrev{X}
\@mtc@setrules@true
\def\mtc@rusw@abbrev{}
\expandafter\ifx\csname mtc@typetable@#1\endcsname\relax
\@mtc@setrules@false
\def\mtc@rusw@abbrev{X}
\def\mtc@mttru@abbrev{X}
\mtcPackageError[E0019]{minitoc}{\string\mtcsetrules \space has a wrong first argument (#1)}{It should be a mini-table type (parttoc...sectlot)
Correct the source code.
Type <return> and rerun LaTeX}
\else
\edef\mtc@mttru@abbrev\@nameuse{mtc@typetable@#1}
\fi
```

Then the second argument, a keyword of the `YN` family, and store the result in a macro `\mtc@rusw@abbrev`. The name of the effective macro is built and the macro executed.

```latex
\if\mtc@rsruok@\mtc@rusw@abbrev\mtc@mttru@abbrev\fi
\def\mtc@rsruok@false
\def\mtc@rusw@abbrev{}
\expandafter\ifx\csname mtc@YN@#2\endcsname\relax
\@mtc@rsruok@false
\def\mtc@rusw@abbrev{X}
\def\mtc@mttru@abbrev{X}
\@mtc@setrules@false
\mtcPackageError[E0020]{minitoc}{\string\mtcsetrules \space has a wrong second argument (#2)}{It should be a boolean value (0/1, yes/no, on/off, ...)
Correct the source code.
Type <return> and rerun LaTeX}
\else
\edef\mtc@rusw@abbrev\@nameuse{mtc@YN@#2}
\mtc@noX\mtc@tmppn@name
\expandafter\ifx\csname mtc@YN@#2\mtc@noX\endcsname\relax
\mtcPackageInfo[I0008]{minitoc}{Horizontal rules are inhibited}
\else
\edef\mtc@rusw@abbrev\@nameuse{mtc@YN@#2}
\def\mtc@rusw@abbrev{X}
\def\mtc@noX{mtc@noX}
\def\mtc@tmppn@name{\mtc@rusw@abbrev rule}
\expandafter\ifx\csname mtc@YN@#2\mtc@noX rule\endcsname\relax
\mtcPackageInfo[I0008]{minitoc}{Horizontal rules are inhibited}
\fi
```

(Type <return> and rerun LaTeX)
Commented code of the minitoc package

for the #1s}
\else
\mtcPackageInfo[I0007]{minitoc}%
{Horizontal rules are activated
\MessageBreak
for the #1s}
\fi
\csname\mtc@tmppn@name\endcsname{}
\fi
}

9.67.8 The \mtcsetfeature command

For this command, we must define three families of keywords, but the third is just used to add the word “style” for the “pagestyle” when “pagestyle” is used.

A family (ltypetable) for the long names of the types of mini-tables:

\@namedef{mtc@ltypetable@parttoc}{parttoc}\def\mtc@ltypetable@parttoc{parttoc}
\@namedef{mtc@ltypetable@partlof}{partlof}\def\mtc@ltypetable@partlof{partlof}
\@namedef{mtc@ltypetable@partlot}{partlot}\def\mtc@ltypetable@partlot{partlot}
\@namedef{mtc@ltypetable@minitoc}{minitoc}\def\mtc@ltypetable@minitoc{minitoc}
\@namedef{mtc@ltypetable@minilof}{minilof}\def\mtc@ltypetable@minilof{minilof}
\@namedef{mtc@ltypetable@minilot}{minilot}\def\mtc@ltypetable@minilot{minilot}
\@namedef{mtc@ltypetable@secttoc}{secttoc}\def\mtc@ltypetable@secttoc{secttoc}
\@namedef{mtc@ltypetable@sectlof}{sectlof}\def\mtc@ltypetable@sectlof{sectlof}
\@namedef{mtc@ltypetable@sectlot}{sectlot}\def\mtc@ltypetable@sectlot{sectlot}

A family (featureparam) for the type of feature:

\@namedef{mtc@featureparam@before}{before}%
\def\mtc@featureparam@before{before}
\@namedef{mtc@featureparam@after}{after}%
\def\mtc@featureparam@after{after}
\@namedef{mtc@featureparam@open}{open}%
\def\mtc@featureparam@open{open}
\@namedef{mtc@featureparam@close}{close}%
\def\mtc@featureparam@close{close}
\@namedef{mtc@featureparam@pagestyle}{thispage}%
\def\mtc@featureparam@pagestyle{thispage}

And a family (ft3) to add “style” if it is a “pagestyle” feature:

\@namedef{mtc@ft3@before}{\expandafter\def\csname mtc@ft3@before\endcsname{}}
\@namedef{mtc@ft3@after}{\expandafter\def\csname mtc@ft3@after\endcsname{}}
\@namedef{mtc@ft3@open}{\expandafter\def\csname mtc@ft3@open\endcsname{}}
\@namedef{mtc@ft3@close}{\expandafter\def\csname mtc@ft3@close\endcsname{}}
\@namedef{mtc@ft3@pagestyle}{style}%
\expandafter\def\csname mtc@ft3@pagestyle\endcsname{style}
The \mtcsetfeature command has the following syntax:

```latex
\mtcsetfeature{mini-table}{feature-name}{commands}
```

where mini-table is a keyword of the \ltypetable family, feature-name is a keyword of the \featureparam family (but also of the \ft3 family), and commands are the commands which constitute the selected feature.

We define a flag and the \mtcsetfeature command, with three arguments:

```latex
\newif\if@mtc@setfeature@\@mtc@setfeature@true
\newcommand{\mtcsetfeature}[3]{%}
\mtc@mtfeat@abbrev
\mtc@featparam@abbrev
\@mtc@setfeature@true
\expandafter\ifx\csname mtc@ltypetable@#1\endcsname\relax
\@mtc@setfeature@false
\def\mtc@mtfeat@abbrev{X}
\def\mtc@featparam@third{X}
\mtcPackageError[E0011]{minitoc}\space has a wrong first argument
\string\mtcsetfeature \space has a wrong first argument
\MessageBreak
\MessageBreak
\MessageBreak
\MessageBreak
\MessageBreak
(\#1).
\MessageBreak
It should be a mini-table type
\MessageBreak
\MessageBreak
\MessageBreak
\MessageBreak
Correct the source code.
\MessageBreak
\MessageBreak
Type <return> and rerun LaTeX}
\else
\edef\mtc@mtfeat@abbrev{\@nameuse{mtc@ltypetable@#1}}
\fi
```

The second argument is a keyword of the \featureparam family, the result is stored in \mtc@featparam@abbrev, and the complement is computed from the first argument, interpreted as a keyword of the \ft3 family and whose result is stored in \mtc@featparam@third.

```latex
\if@mtc@setfeature@
\mtc@featparam@abbrev
\mtc@featparam@third
\@nameuse\MessageBreak
\expandafter\ifx\csname mtc@featureparam@#2\endcsname\relax
\@mtc@setfeature@false
\def\mtc@featparam@abbrev{X}
\def\mtc@featparam@third{X}
\mtcPackageError[E0012]{minitoc}\space has a wrong second argument
```

\MessageBreak
9.6.7.9 The \texttt{mtcsetdepth} command

This command is very similar to the \texttt{mtcsettitle} command. Its syntax is almost identical:

\begin{verbatim}
\mtcsetdepth\{mini-table\}\{depth\}
\end{verbatim}

The \texttt{mini-table} type is a keyword like \texttt{minitoc}. The \texttt{depth} is the depth for a mini-table. If it is a mini-table for a list of figures or tables, the corresponding depth counter must be available, i.e., must have been created (often by an adequate package, like the subfig package\cite{subfig}).
First, we declare a flag, set true:
\newif\ift@mtc@setdepth\t@mtc@setdepthtrue

Then we define the \mtcsetdepth command, with two arguments:
\newcommand{\mtcsetdepth}[2]{%
\mtc@mtade@abbrev
\ift@mtc@setdepth\@nameuse
We process the first argument, a keyword of the \ltypetable family. The result is stored in
\def\mtc@mtade@abbrev{X}
\@t@mtc@setdepthtrue
\expandafter\ifx\csname mtc@ltypetable@#1\endcsname\relax
\ift@mtc@setdepthfalse
\def\mtc@mtade@abbrev{X}
\mtcPackageError[E0009]{minitoc}{% 
\string\mtcsetdepth \space has a wrong first argument
\MessageBreak
\MessageBreak
\MessageBreak
It should be a mini-table type
\MessageBreak
\MessageBreak
{Correct the source code.
\MessageBreak
\MessageBreak
Type <return> and rerun LaTeX}
\else
\edef\mtc@mtade@abbrev{\@nameuse{mtc@ltypetable@#1}}
\fi
And we construct the name of the effective counter and gave it the value:
\if\ift@mtc@setdepth\t@mtc@setdepthfalse
\def\mtc@tmpde@name{\mtc@mtade@abbrev depth}
\@ifundefined{c@\mtc@tmpde@name}{% 
\string\mtcsetdepth \space attempts to use
\MessageBreak
\MessageBreak
an undefined counter (#1depth).}{{Correct the source code.
\MessageBreak
\MessageBreak
Type <return> and rerun LaTeX}}{% 
\mtcPackageInfo[I0013]{minitoc}{% 
{\string\mtcsetdepth \space redefines the counter
\MessageBreak
\MessageBreak
"\mtc@tmpde@name" as "\the\mtc@toks"}%
\expandafter\csname c@\mtc@tmpde@name\endcsname=#2}%
9.67.10 The \mtcsetoffset command

This command is very similar to the \mtcsettitle command. Its syntax is almost identical:

\mtcsetoffset{mini-table}{value}

The mini-table type is a keyword like minitoc. The value is the offset value for a mini-table.

\if@mtc@setoffset@ First, we declare a flag, set true:
\newif\if@mtc@setoffset@ \@mtc@setoffset@true

\mtcsetoffset Then we define the \mtcsetoffset command, with two arguments:
\newcommand{\mtcsetoffset}{2}[

\mtc@mtaof@abbrev We process the first argument, a keyword of the typetable family. The result is stored in \mtc@mtaof@abbrev:
\if@mtc@setoffset@ \@nameuse{mtc@typetable@#1}\fi

\def{\mtc@mtaof@abbrev}{X}
\edef{\mtc@mtaof@abbrev}{\@nameuse{mtc@typetable@#1}}

\else
\edef{\mtc@mtaof@abbrev}{\@nameuse{mtc@typetable@#1}}
\fi
And we construct the name of the effective offset and gave it the value:
\begin{verbatim}
\if@mtc@setoffset@ \mtc@tmpof@name \mtc@mtaof@abbrev \setcounter \mtc@toks \setcounter \mtc@OFFSET
\edef\mtc@tmpof@name{\mtc@mtaof@abbrev offset.}
\@ifundefined{\mtc@mtaof@abbrev offset}{}
\mtcPackageError[E0041]{minitoc}{\string\mtcsetoffset \space attempts to use \the\mtc@toks}
\MessageBreak (Correct the source code. Type <return> and rerun LaTeX)
\mtcPackageInfo[I0052]{minitoc}{\string\mtcsetoffset \space redefines \"\mtc@mtaof@abbrev offset\" as \"\the\mtc@toks\"}
\expandafter\def\csname \mtc@mtaof@abbrev offset\endcsname{#2}
\else
\mtcPackageError[E0043]{minitoc}{\string\mtcsetoffset: \space Illegal type of table (#1)}
\MessageBreak (Correct the source code. Type <return> and rerun LaTeX)
\relax\fi\end{verbatim}

9.68 Polymorphic entries

A toc entry should be able to have variants when it appears in the normal text (like the mandatory argument of a sectionning command), in a page header or in the main TOC (like the optionnal argument of a sectionning command), in a minitable (parttoc, minitoc or secttoc). Similar behaviour should be available for entries in the LOF or the LOT. So we define three commands to be used inside the optionnal argument of a sectionning command or of \caption for a figure or a table. These commands must be robust (because used in optionnal arguments) and have 4 arguments: (1) the variant to appear in a parttoc (or partlof or partlof), (2) the variant to appear in a minitoc (or minilof or minilof), (3) the variant to appear in a secttoc (or sectlof or sectlof), (4) the variant to appear in the main TOC (or LOF or LOT). The variant to appear locally as title of the sectionning unit or as local caption of the figure or table is the mandatory argument of the sectionning command or of the caption command (see section 1.4.13 on page 43). We use the \ifin... flags.

\begin{verbatim}
\DeclareRobustCommand{\mtcpolymtoc}[4]{\ifinparttoc\relax[#1]\else\ifinminitoc\relax[#2]\else\ifinsecttoc\relax[#3]\else\relax[#4]\fi\fi\fi\fi}
\end{verbatim}
9.69 The \texttt{mtchideinmaintoc} environment and siblings

The flag \texttt{\if@mtc@Himtoc@} is used to detect an incorrect imbrication of this environment:

\begin{verbatim}
\newif\if@mtc@Himtoc@
@mtc@Himtoc@false
\newcommand{\mtc@savetocdepth}{\xdef\mtc@sv@tocdepth{\arabic{tocdepth}}}%
\newcommand{\mtc@restoretocdepth}{\setcounter{tocdepth}{\mtc@sv@tocdepth}}%
\newenvironment{mtchideinmaintoc}{\if@mtc@Himtoc@\mtcPackageError[\texttt{E0005}]{minitoc}{Imbrication of mtchideinmaintoc environments}{The hiding in main ToC could be incorrect}\fi}{\if@mtc@Himtoc@false\fi}
\end{verbatim}
The \texttt{mtchideinmainlof} and \texttt{mtchideinmainlot} environments are similar, but we must verify the presence of the associated depth counter, so we have two versions of each of these environments. This must be done after the loading of the packages.

First, for the list of figures:

\begin{verbatim}
\AtBeginDocument{\if@mtc@Himlof@ \mtc@savelofdepth \empty \mtc@sv@lofdepth \mtc@sv@tocdepth \mtc@svf@tocdepth \arabic \addtocontents \setcounter}
\newif\if@mtc@Himlof@ \@mtc@Himlof@false
\AtBeginDocument{\@ifundefined{c@lofdepth}{\providecommand{\mtc@savelofdepth}{\empty}}}{\newenvironment{mtchideinmainlof}[1][-1]{\if@mtc@Himlof@\mtcPackageError{E0003}{minitoc}{Imbrication of mtchideinmainlof environments}{The hiding in main LoF could be incorrect}}{\global\@mtc@Himlof@true}
\def\mtc@sv@tocdepth{\arabic{tocdepth}}\def\mtc@sv@lofdepth{\arabic{lofdepth}}\addtocontents{lof}{\protect\mtc@savetocdepth}\addtocontents{lof}{\protect\setcounter{tocdepth}{#1}}}{\if@mtc@Himlof@false\mtcPackageError{E0029}{minitoc}{Unbalanced mtchideinmainlof environment}{The hiding in main LoF could be incorrect}}\global\@mtc@Himlof@false\addtocontents{lof}{\protect\mtc@restoretocdepth}}
\end{verbatim}

Then for the list of tables:

\begin{verbatim}
\AtBeginDocument{\if@mtc@Himlot@ \mtchideinmainlot \mtc@saveloddepth \mtc@sv@loddepth \mtc@sv@tocdepth \mtc@svf@tocdepth \arabic \addtocontents \setcounter}
\newif\if@mtc@Himlot@ \@mtc@Himlot@false
\AtBeginDocument{\@ifundefined{c@loddepth}{\newcommand{\mtc@saveloddepth}{\xdef\mtc@sv@loddepth{\arabic{loddepth}}}\newcommand{\mtc@restorloddepth}{\setcounter{loddepth}{\mtc@sv@loddepth}}\newenvironment{mtchideinmainlot}[1][-1]{\if@mtc@Himlot@\mtcPackageError{E0003}{minitoc}{Imbrication of mtchideinmainlot environments}{The hiding in main LoT could be incorrect}}{\global\@mtc@Himlot@true}
\def\mtc@sv@tocdepth{\arabic{tocdepth}}\def\mtc@sv@loddepth{\arabic{lofdepth}}\addtocontents{lod}{\protect\mtc@savetocdepth}\addtocontents{lod}{\protect\setcounter{tocdepth}{#1}}}{\if@mtc@Himlot@false\mtcPackageError{E0029}{minitoc}{Unbalanced mtchideinmainlot environment}{The hiding in main LoT could be incorrect}}\global\@mtc@Himlot@false\addtocontents{lod}{\protect\mtc@restoretocdepth}}
\end{verbatim}
9.70 Fixing the “Glossary” entry in the TOC

This macro is complex. Its syntax is:

\mtcfixglossary[part|chapter|section]

\@ifundefined{\mtc@glofix@level} Depending on the document class, the “Glossary” entry in the TOC is treated as a starred chapter or a starred section. Hence we must first determine the default value of the optional argument. The default value is then stored in the macro \mtc@glofix@level. This is done by the following code, which eventually gives a warning message:

\@ifundefined{chapter} \@ifundefined{section}
Commented code of the minitoc package

```latex
\MessageBreak
Cannot use \string\mtcfixglossary \space without \MessageBreak
optional argument [part]%%
\def\if@mtcfixglossary%@false
\expandafter\ifx\csname #1\endcsname\part\relax\@mtcfixglossary@true\fi
\expandafter\ifx\csname #1\endcsname\chapter\relax\@mtcfixglossary@true\fi
\expandafter\ifx\csname #1\endcsname\section\relax\@mtcfixglossary@true\fi
\if@mtcfixglossary@
\addcontentsline{lof}{x\mtc@glofix@level}{}%
\addcontentsline{lot}{x\mtc@glofix@level}{}%
\csname mtcadd\mtc@glofix@level\endcsname\relax
\else
\MessageBreak
\mtcPackageError[E0026]{minitoc}%%
{The optional argument of \string\mtcfixglossary
 \MessageBreak
is wrong}%%
{It must be omitted (\mtc@glofix@level), or be part, chapter or section}%%
\fi
```

Then we define a flag (\if@mtcfixglossary@) and the command \mtcfixglossary, which adds the necessary lines in the TOC, the LOF and the LOT.
9.71 Fixing the “Index” entry in the TOC

This macro is complex. Its syntax is:

\mtcfixindex[part|chapter|section]

\@ifundefined \mtc@ixfix@level Depending on the document class, the “Index” entry in the TOC is treated as a starred chapter or a starred section. Hence we must first determine the default value of the optional argument. The default value is then stored in the macro \mtc@ixfix@level. This is done by the following code, which eventually gives a warning message:

\newif\if@mtcfixindex@ \@mtcfixindex@false \newcommand{\mtcfixindex}[1]{\@mtcfixindex@false \expandafter\ifx\csname #1\endcsname\part\relax\@mtcfixindex@true\fi \expandafter\ifx\csname #1\endcsname\chapter\relax\@mtcfixindex@true\fi \expandafter\ifx\csname #1\endcsname\section\relax\@mtcfixindex@true\fi \if@mtcfixindex@ \addcontentsline{lof}{x\mtc@ixfix@level}{}% \fi \def\mtc@ixfix@level{part} \def\mtc@ixfix@level{section}}} \if@mtcfixindex@ \mtcfixindex \addcontentsline

\newif\if@mtcfixindex@ \@mtcfixindex@false \newcommand{\mtcfixindex}[1]{\@mtcfixindex@false \expandafter\ifx\csname #1\endcsname\part\relax\@mtcfixindex@true\fi \expandafter\ifx\csname #1\endcsname\chapter\relax\@mtcfixindex@true\fi \expandafter\ifx\csname #1\endcsname\section\relax\@mtcfixindex@true\fi \if@mtcfixindex@ \addcontentsline{lof}{x\mtc@ixfix@level}{}% \fi \def\mtc@ixfix@level{part} \def\mtc@ixfix@level{section}}} \if@mtcfixindex@ \mtcfixindex \addcontentsline
9.72 Fixing the “Nomenclature” entry in the TOC

This macro is complex. Its syntax is:

\mtcfixnomenclature[part|chapter|section]

Depending on the document class, the “Nomenclature” entry in the TOC is treated as a starred chapter or a starred section. Hence we must first determine the default value of the optional argument. The default value is then stored in the macro \mtc@nomenclfix@level.

This is done by the following code, which eventually gives a warning message:

\@ifundefined{chapter}{%\@ifundefined{section}{}{\mtcPackageWarningNoLine[W0095]{minitoc}{\string\chapter\space and \string\section\space are undefined.\MessageBreakCannot use \string\mtcfixnomenclature\space without\MessageBreakoptional argument [part]}}%\@ifundefined{part}{}{%\mtcPackageWarningNoLine[W0096]{minitoc}{\string\mtcfixnomenclature\space can only be used with\MessageBreakthe [part] optional argument,\MessageBreakwhich becomes the default}%%\def\mtc@nomenclfix@level{part}%%\def\mtc@nomenclfix@level{section}}%\def\mtc@nomenclfix@level{chapter}}

\footnote{If you are using the nomencl package [456] or nomentbl package [161] (nomencl calls nomentbl).}
Then we define a flag and the command \mtcfixnomenclature, which adds the necessary lines in the TOC, the LOF and the LOT.

\if@mtcfixnomclature@ \mtcfixnomclaturefalse
\newcommand{\mtcfixnomenclature}{\mtc@nomenclfix@level}{%\addcontentsline{lof}{x\mtc@nomenclfix@level}{}%\addcontentsline{lot}{x\mtc@nomenclfix@level}{}%\csname mtcadd\mtc@nomenclfix@level\endcsname\relax\else\mtcPackageError{E0040}{minitoc}{The optional argument of \string\mtcfixnomenclature is wrong}{It must be omitted (\mtc@nomenclfix@level), or be part, chapter or section}%\fi}%

9.73 The \mtcselectlanguage command

This command loads a minitoc language definition file language.mld to set the language-dependent titles for the mini-tables. But first, we verify that this file exists. The flag \ifmtc@insellang@ is true while we are in this macro.

\mtcselectlanguage{\ifmtc@insellang@ \mtc@insellangfalse \IfFileExists{\input}{#1.mld}{%\mtcPackageInfo{I0010}{minitoc}{The #1 language is selected.}{It must be omitted (\mtc@nomenclfix@level), or be part, chapter or section}%\mtcPackageError{E0006}{minitoc}{#1 is not a known language, #1.mld not found.}{Command ignored}{See the minitoc documentation.}{Correct the source using a valid language name.}{Press RETURN}}}
This command loads a minitoc language object file \texttt{language.mlo} to set the language-dependent titles for the mini-tables when exotic characters are needed. This command is used only in some .mld files when the title strings can not be generated by the normal processing of minitoc.dtx. The .mlo files are generated by \texttt{filecontents} environments in the minitoc.ins file. But first, we verify that this .mlo file exists.

\textit{This command should not be invoked directly by the user. This is verified via the flag \texttt{if@mtc@insellang@}.}

\begin{verbatim}
def\mtcloadmlo#1{% 
  \ifmtc@insellang@
  \InputIfFileExists{#1.mlo}{}%  
    \mtcPackageInfoSKU011\{minitoc\}{}%  
    #1 minitoc language object selected.  
    \MessageBreak%  
  \else  
    \mtcPackageErrorSKU007\{minitoc\}{}%  
    #1 is not a known minitoc language object file (.mlo),  
    #1.mlo not found.  
    \MessageBreak  
    Command ignored\}  
    \See the minitoc documentation.  
    \MessageBreak  
    Correct the source using a valid language name.  
    \MessageBreak  
    Press RETURN}%)% 
\end{verbatim}
9.75 The “coffee breaks”

For the minutes package [300] (by Knut Lückert), we need some commands to insert special entries, undotted, in the TOC to mark “coffee breaks” in a conference. Hence we define \addcoffeeline, \coffeeline and \l@coffee, and internal commands analog to the standard internal commands to format the TOC.

\def\addcoffeeline#1#2#3{\addtocontents{#1}{\protect\coffeeline{#2}{#3}{\null}}}
\def\coffeeline#1{\csname l@#1\endcsname}
\newcommand\l@coffee{\@Undottedtocline{1}{1.5em}{2.3em}}

9.76 Initialization of counters

At the beginning of the document, we initialize the absolute counters for parts, chapters and sections, if they are defined.

\AtBeginDocument{\if@Ifundefined{c@ptc}{\setcounter{ptc}{0}}{\setcounter{ptc}{0}}}
\AtBeginDocument{\if@Ifundefined{c@mtc}{\setcounter{mtc}{0}}{\setcounter{mtc}{0}}}
\AtBeginDocument{\if@Ifundefined{c@stc}{\setcounter{stc}{0}}{\setcounter{stc}{0}}}

9.77 Declarations for simple options

These options are just setting a flag.

9.77.1 Options tight and loose, k-tight and k-loose

\DeclareOption{tight}{\tightmtctrue}
\ DeclareOption{loose}{\tightmtcfalse} % default
\ DeclareOption{k-tight}{\ktightmtctrue}
\ DeclareOption{k-loose}{\ktightmtcfalse} % default
9.77.2 Options checkfiles and nocheckfiles

These options activate or inhibit the checking for empty mini-table files.

\DeclareOption{checkfiles}{\@mtc@checkfilestrue} % default
\DeclareOption{nocheckfiles}{\@mtc@checkfilesfalse}

9.77.3 Options dotted and undotted

These options activate or inhibit the leaders (lines of dots) in the mini-tables.

\DeclareOption{undotted}{\undottedmtctrue} \DeclareOption{dotted}{\undottedmtcfalse} % default

9.77.4 Option notoccite

This option will later load the notoccite package [14].

\DeclareOption{notoccite}{\@mtc@notoccite@true}

9.77.5 Option shortext

This option forces the use of short extensions.

\DeclareOption{shortext}{\@mtc@longext@false} \mtcPackageWarningNoLine[W0020]{minitoc}%{You have forced the use of short extensions}

9.78 The insection option

This option is available only if \chapter is not defined and \section defined. It is to be revised when chapter/section level commands will ever be allowed together, sometime in the far away future, with a lot of luck (and work)\textsuperscript{16}.

\if@mtc@ss@insection@ \ifundefined \DeclareOption{\@mtc@ss@insection@false}

\textsuperscript{16}Please, do not dream too much!
9.79 The listfiles and nolistfiles options

\if@mtc@listfiles@ \DeclareOption{listfiles}{\@mtc@listfiles@true} \fi
\DeclareOption{nolistfiles}{\@mtc@listfiles@false}

9.80 Language options

\def\@gobblethree#1#2#3{\empty}
\def\mtc@listmisslanguages{}\def\mtc@addmisslanguage#1{\let\mtc@LML\mtc@listmisslanguages\edef\mtc@listmisslanguages{\mtc@LML \MessageBreak #1}}
Before defining a language option, we must verify that the corresponding .mld file exists, and, if necessary, that the corresponding .mlo file exists. Hence, we must first define a flag \if@mtc@misslang and two macros to test the presence of these files; if the files are available, we define the language option.

\newcommand{\mtc@setlangopt}{\IfFileExists{#1.mld}{\DeclareOption{#1}{\mtcselectlanguage{#1}}}{\@mtc@misslangtrue \mtc@addmisslanguage{#1.mld} \mtcPackageWarningNoLine{minitoc}{The required "#1.mld" file is missing. \MessageBreak The "#1" language option will not be available. \MessageBreak Please install it from a recent distribution \MessageBreak or from the CTAN archives\@gobble}}}{\IfFileExists{#1.mlo}{\mtc@setlangopt{#1}}{\@mtc@misslangtrue \mtc@addmisslanguage{#1.mlo} \mtcPackageInfo{minitoc}{The required "#1.mlo" file is missing. \MessageBreak The "#1" language option will not be available. \MessageBreak Please install it from a recent distribution \MessageBreak or from the CTAN archives\@gobble}}}{\IfFileExists{#1.mld}{}{\@mtc@misslangtrue \mtc@addmisslanguage{#1.mld} \mtcPackageInfo{minitoc}{The required "#1.mld" file is missing. \MessageBreak The "#1" language option will not be available. \MessageBreak Please install it from a recent distribution \MessageBreak or from the CTAN archives\@gobble}}}%

\newcommand{\mtc@setlangoptm}{\IfFileExists{#1.mld}{\DeclareOption{#1}{\mtcselectlanguage{#1}}}{}{\@mtc@misslangtrue \mtc@addmisslanguage{#1.mld} \mtcPackageInfo{minitoc}{The required "#1.mld" file is missing. \MessageBreak The "#1" language option will not be available. \MessageBreak Please install it from a recent distribution \MessageBreak or from the CTAN archives\@gobble}}}%

Some .mld files are mandatory (english.mld because english is the default language), so their absence is a serious error:
\providecommand \ptctitle  
\plftitle  
\plttitle  
\mtctitle  
\mlftitle  
\mlttitle  
\stctitle  
\slftitle  
\slttitle  

\providecommand\ptctitle{Table of Contents}  
\providecommand\plftitle{List of Figures}  
\providecommand\plttitle{List of Tables}  
\providecommand\mtctitle{Contents}  
\providecommand\mlftitle{Figures}  
\providecommand\mlttitle{Tables}  
\providecommand\stctitle{Contents}  
\providecommand\slftitle{Figures}  
\providecommand\slttitle{Tables}  

\AtEndDocument  
\if@mtc@misslang  
\mtcPackageWarningNoLine  
\MessageBreak  
\mtc@listmisslanguages  
@gobblethree  
\mtcPackageWarningNoLine[W0093] [minitoc]  
\MessageBreak  
{Some ".mld" or ".mlo" files are missing}  
\MessageBreak  
in your installation.  
\MessageBreak  
Search for the I0050 and I0051 info messages  
\MessageBreak  
in the \string\jobname.log file.  
\MessageBreak  
The full list of the missing language files  
\MessageBreak  
is given in the W0094 warning message.  
\MessageBreak  
Please install the missing files from
Each language option reads the corresponding *language*.mld file via the specialized macro \mtcselectlanguage, after verification by \mtc@setlangopt or \mtc@setlangopto (when a .mlo file is required), by \mtc@setlangoptm when the language is mandatory. If the file does not exist, a standard error message is displayed. The language options are (should be) in alphabetical order (to make maintenance easier). Several options could load the same file, but, by convention, there should be a *language*.mld file for each language option, given that this file may load another one (as *american*.mld loads *english*.mld).
Commented code of the minitoc package

\mct@setlangopt{germanb}%
\mct@setlangopt{germanb2}%
\mct@setlangopt{greek}%
\mct@setlangopt{greek-mono}%
\mct@setlangopt{greek-polydemo}%
\mct@setlangopt{greek-polykatha}%
\mct@setlangopt{guarani}%
\mct@setlangopto{hangul1}%
\mct@setlangopto{hangul2}%
\mct@setlangopto{hangul3}%
\mct@setlangopto{hangul4}%
\mct@setlangopto{hangul-u8}%
\mct@setlangopto{hanja1}%
\mct@setlangopto{hanja2}%
\mct@setlangopto{hanja-u8}%
\mct@setlangopt{hebrew}%
\mct@setlangopt{hebrew2}%
\mct@setlangopt{hindi}%
\mct@setlangopt{hindi-modern}%
\mct@setlangopt{hungarian}%
\mct@setlangopt{icelandic}%
\mct@setlangopt{indon}%
\mct@setlangopt{indonesian}%
\mct@setlangopt{interlingua}%
\mct@setlangopt{irish}%
\mct@setlangopt{italian}%
\mct@setlangopt{italian2}%
\mct@setlangopto{japanese}%
\mct@setlangopto{japanese2}%
\mct@setlangopto{japanese3}%
\mct@setlangopto{japanese4}%
\mct@setlangopto{japanese5}%
\mct@setlangopto{japanese6}%
\mct@setlangopto{kannada}%
\mct@setlangopto{khalkha}%
\mct@setlangopto{latin}%
\mct@setlangopto{latin2}%
\mct@setlangopto{latinc}%
\mct@setlangopto{latinc2}%
\mct@setlangopto{latvian}%
\mct@setlangopto{latvian2}%
\mct@setlangopto{letton}%
\mct@setlangopto{letton2}%
\mct@setlangopto{lituanian}%
\mct@setlangopto{lituanian2}%
\mct@setlangopto{lowersorbian}%
\mct@setlangopto{magyar}%
\mct@setlangopto{magyar2}%
\mct@setlangopto{magyar3}%
\mct@setlangopto{malay}%
\mct@setlangopto{malayalam-b}%
\mct@setlangopto{malayalam-keli}%
\mct@setlangopto{malayalam-keli2}%
The hints option is made of three parts: the first, `\mtc@hints@begindoc`, is executed via `\AtBeginDocument` and looks if some packages or classes are loaded, then gives warnings about their compatibility with `minitoc`.

The second part is made of tiny pieces of code inserted in the `minitoc` code, to verify that some macros are called in the right order.

The third and last part, `\mtc@hints@enddoc`, is executed via `\AtEndDocument` and examines the flags set by the first and the second parts. Then, if necessary, it writes some infos in the `document.log` file and/or warnings on the screen and in the `document.log` file. The hints option does not signal errors, only infos and warnings, so it does not stop the \LaTeX run.

First part: `\mtc@hints@begindoc`

We declare some flags and the first part of the hints option (for an `\AtBeginDocument` block):

```latex
\DeclareOption{hints}{\@mtc@hints@true}
\DeclareOption{nohints}{\@mtc@hints@false}
\newif\if@mtc@abstract@loaded\@mtc@abstract@loadedfalse
\newif\if@mtc@toc@used\global\@mtc@toc@usedfalse
\newif\if@mtc@lof@used\global\@mtc@lof@usedfalse
\newif\if@mtc@lot@used\global\@mtc@lot@usedfalse
\DeclareOption{hints}{\@mtc@abstract@loadedfalse\@mtc@toc@usedfalse\@mtc@lof@usedfalse\@mtc@lot@usedfalse
  \mtcPackageInfo[10049]{minitoc(hints)}
  {==> You requested the hints option.
    Some hints are eventually given below\@gobble}}
```

9.81.1 First part: `\mtc@hints@begindoc`
9.81.1.1 Hint about the alphanum package

\ifpackageloaded{alphanum} We test the presence of the alphanum package (part of the jura class [103]), and emit a warning, because this package is incompatible with minitoc:

\if@mtc@hints@given@ We test the presence of the alphanum package (part of the jura class [103]), and emit a warning, because this package is incompatible with minitoc:

\mtcPackageWarningNoLine[W0025]{minitoc(hints)}%

--- The alphanum package is loaded.
\MessageBreak
It is incompatible
\MessageBreak
with the minitoc package}}%

9.81.1.2 Hint about the appendix package

\ifpackageloaded{appendix} We test the presence of the appendix package [471]:

\if@mtc@hints@given@ We test the presence of the appendix package [471]:

\mtcPackageInfo[I0042]{minitoc(hints)}%

--- The appendix package is loaded.
\MessageBreak
See the minitoc package documentation
\MessageBreak
for specific precautions\@gobble}}%

9.81.1.3 Hint about the tocbibind package

\ifpackageloaded{tocbibind} We test the presence of the tocbibind package [472]:

\if@mtc@hints@given@ We test the presence of the tocbibind package [472]:

\mtcPackageInfo[I0046]{minitoc(hints)}%

--- The tocbibind package is loaded.
\MessageBreak
See the minitoc package documentation
\MessageBreak
for specific precautions\@gobble}}%

9.81.1.4 Hint about the KOMA-Script classes

\ifclassloaded We test the presence of each minitoc-compatible KOMA-Script class [343, 344, 399]:

\if@mtc@hints@given@ We test the presence of each minitoc-compatible KOMA-Script class [343, 344, 399]:

Commented code of the minitoc package

\ifclassloaded{scrbook}{}
\mtcPackageInfo[10043]{minitoc(hints)}%
\MessageBreak
\MessageBreak
See the minitoc package documentation
\MessageBreak
for specific precautions\gobble}
\ifclassloaded{scrreprt}{}
\mtcPackageInfo[10043]{minitoc(hints)}%
\MessageBreak
\MessageBreak
See the minitoc package documentation
\MessageBreak
for specific precautions\gobble}
\ifclassloaded{scrartcl}{}
\mtcPackageInfo[10043]{minitoc(hints)}%
\MessageBreak
\MessageBreak
See the minitoc package documentation
\MessageBreak
for specific precautions\gobble}
\ifpackageloaded\if@mtc@hints@given@
We test the presence of the tocloft package [469]:
\ifpackageloaded{tocloft}{}
\mtcPackageInfo[10047]{minitoc(hints)}%
\MessageBreak
\MessageBreak
See the minitoc package documentation
\MessageBreak
for specific precautions\gobble}
\ifpackageloaded{titlesec}{}
\mtcPackageWarningNoLine[W0099]{minitoc(hints)}%
\MessageBreak
\MessageBreak
See the minitoc package documentation
\MessageBreak
for specific precautions\gobble}

9.81.1.5 Hint about the tocloft package

9.81.1.6 Hint about the titlesec package
[9] — Commented code of the minitoc package

\MessageBreak
It is incompatible
\MessageBreak
with the minitoc package

9.81.1.7 Hint about the titletoc package

\ifpackageloaded{titletoc}{
\Ifmtc@hints@given@
\We test the presence of the titletoc package [46], and emit a warning, because this package is incompatible with minitoc:

\@ifpackageloaded{titletoc}{
\Ifmtc@hints@given@
\mtcPackageWarningNoLine[W0040]{minitoc(hints)}{--- The titletoc package is loaded.}
\MessageBreak
\It is incompatible
\MessageBreak
\with the minitoc package}{}}%

9.81.1.8 Hint about the placeins package

\ifpackageloaded{placeins}{
\Ifmtc@ss@insection@\Ifpackagewith{placeins}{section}{
\Ifmtc@hints@given@
\mtcPackageWarningNoLine[W0031]{minitoc(hints)}{--- The placeins package is loaded}
\MessageBreak
\without the section option,
\MessageBreak
\but minitoc used the insection option
\MessageBreak
\which implies it. Try to inverse the
\MessageBreak
\loading order and use consistent options.
\MessageBreak
\You may have got a message
\MessageBreak
\LaTeX Error: Option clash for package placeins}{}}%
\MessageBreak
\Ifpackagewith{placeins}{above}{
\Ifmtc@hints@given@
\mtcPackageWarningNoLine[W0084]{minitoc(hints)}{--- The placeins package is loaded}
\MessageBreak
\}%
Commented code of the minitoc package

\MessageBreak
With the above option,
\MessageBreak
but minitoc used the insection option
\MessageBreak
which is incompatible with it.
\MessageBreak
Try to remove the above option
\MessageBreak
and use consistent options)%
}\{}
\ifpackagewith{placeins}{below}%
\ifmtc@hints@given@true
\mtcPackageWarningNoLine[W0085]{minitoc(hints)}%
\ MessageBreak
--- The placeins package is loaded
\MessageBreak
with the below option,
\MessageBreak
but minitoc used the insection option
\MessageBreak
which is incompatible with it.
\MessageBreak
Try to remove the below option
\MessageBreak
and use consistent options)%
}\{}
\fi
\ifpackagelater{placeins}{2005/04/18}{%\ifmtc@hints@given@true
\mtcPackageWarningNoLine[W0032]{minitoc(hints)}%
\ MessageBreak
--- The placeins package loaded is too old. You should use a version
dated of 2005/04/18 at least}%
}\{%
\%

9.81.1.9 Hint about the memoir class

\ifclassloaded We test if the memoir class [479, 481, 482] is loaded:
\ifmtc@hints@given@true
\mtcPackageInfo[I0044]{minitoc(hints)}%
\ MessageBreak
--- The memoir class is loaded.
\MessageBreak
See the minitoc package documentation
for specific precautions\@gobble}{%
9.81.1.10 Hint about the amsart and amsproc classes

We test if the amsart or amsproc class is loaded and emit a warning, because these classes are incompatible with minitoc:

\ifclassloaded{amsart}\%\ifmtc@hints@given@\mtcPackageWarningNoLine[W0026]{minitoc(hints)}\%--- The amsart class is loaded.\MessageBreakIt is incompatible with the minitoc package\%\fi\%\fi\ifclassloaded{amsproc}\%\ifmtc@hints@given@\mtcPackageWarningNoLine[W0027]{minitoc(hints)}\%--- The amsproc class is loaded.\MessageBreakIt is incompatible with the minitoc package\%\fi\%

9.81.1.11 Hint about the amsbook class

We test if the amsbook class is loaded:

\ifclassloaded{amsbook}\%\ifmtc@hints@given@\mtcPackageInfo[I0041]{minitoc(hints)}\%--- The amsbook class is loaded.\MessageBreakSee the minitoc package documentation for specific precautions\@gobble\%\fi\%

9.81.1.12 Hint about the abstract package

We test the presence of the abstract package [470], then its options:

\ifpackageloaded{abstract}\%\ifpackagewith{abstract}\%\ifmtc@hints@given@\mtcPackageInfo[I0048]{minitoc(hints)}\%The "abstract" package has been\MessageBreak\fi\%\fi\%

---
9.81.1.13 Hint about the \texttt{jura} class

\@ifclassloaded\texttt{jura} We test if the \texttt{jura} class is loaded and emit a warning, because this class is incompatible with \texttt{minitoc}:

\begin{verbatim}
\@ifclassloaded{jura}\%
{\mtc@hints@given@true
  \mtcPackageWarningNoLine\[W0029\]{minitoc(hints)}%
  {--- The jura class is loaded.
   \MessageBreak
   It is incompatible with the minitoc package}}%
\end{verbatim}

9.81.1.14 Hint about the \texttt{flowfram} package

\@ifpackageloaded\texttt{flowfram} We test the presence of the \texttt{flowfram} package \cite{433, 434}, and emit a warning, because this package is incompatible with \texttt{minitoc}:

\begin{verbatim}
\@ifpackageloaded{flowfram}\%
{\mtc@hints@given@true
  \mtcPackageWarningNoLine\[W0097\]{minitoc(hints)}%
  {--- The flowfram package is loaded.
   \MessageBreak
   It is incompatible
   \MessageBreak
   with the minitoc package}}%
\end{verbatim}

9.81.1.15 Hint about the alteration of the sectionning commands

To check if the sectionning commands \texttt{\part}, \texttt{\chapter} or \texttt{\section} have been altered by some package or in the preamble, we compare them (when executing an \texttt{\AtBeginDocument} block) with their saved versions (saved by the \texttt{minitoc} package when it is loaded) \texttt{\mtc@hints@part}, \texttt{\mtc@hints@chapter} and \texttt{\mtc@hints@section}. For each sectionning command, we must perform the comparison for the command itself, its unstarred branch and its starred branch. But the \texttt{hyperref} package \cite{390} may interfere, hence the formal precautions in the messages.
9.81.15.1 Alteration of \part

We check the alteration of \part, \@part and \@spart:

\if@mtc@hints@w@
\@ifundefined{part}\{}\{\ifx\part\mtc@hints@part\relax \else\@mtc@hints@w@true\fi\}
\@ifundefined{part}\{}\{\ifx\@part\mtc@hints@@part\relax \else\@mtc@hints@w@true\fi\}
\@ifundefined{part}\{}\{\ifx\@spart\mtc@hints@@spart\relax \else\@mtc@hints@w@true\fi\}
\if@mtc@hints@w@
\@mtc@hints@given@true\%
\mtcPackageWarningNoLine[W0030]{minitoc(hints)}{--- The \string\part\space command is altered after minitoc}\MessageBreak
\if@mtc@hyper@used@
\mtcPackageWarningNoLine[W0023]{minitoc(hints)}{--- It may be the consequence of loading the ‘‘hyperref’’ package}\fi
\fi

9.81.15.2 Alteration of \chapter

We check the alteration of \chapter, \@chapter and \@schapter:

\if@mtc@hints@w@
\@ifundefined{chapter}\{}\{\ifx\chapter\mtc@hints@chapter\relax \else\@mtc@hints@w@true\fi\}
\@ifundefined{chapter}\{}\{\ifx\@chapter\mtc@hints@@chapter\relax \else\@mtc@hints@w@true\fi\}
\@ifundefined{chapter}\{}\{\ifx\@schapter\mtc@hints@@schapter\relax \else\@mtc@hints@w@true\fi\}
\if@mtc@hints@w@
\@mtc@hints@given@true\%
\mtcPackageWarningNoLine[W0028]{minitoc(hints)}{--- The \string\chapter\space command is altered after minitoc}\MessageBreak
\if@mtc@hyper@used@
\mtcPackageWarningNoLine[W0023]{minitoc(hints)}{--- It may be the consequence of loading the ‘‘hyperref’’ package}\fi
\fi
9.81.1.15.3 Alteration of \section

We check the alteration of \section, \@sect and \@ssect:

\if@mtc@hints@w@\if@mtc@hyper@used@\mtcPackageWarningNoLine[W0039]{minitoc(hints)}%--- The \string\section\space command is altered after minitoc\fi\else\fi\relax\else\fi

\if@mtc@hints@w@false\if@mtc@hyper@used@\mtcPackageWarningNoLine[W0023]{minitoc(hints)}%--- It may be the consequence of loading the ‘‘hyperref’’ package\fi\else\fi
\relax\else\fi{\relax}\if@mtc@hints@w@false\if@mtc@hyper@used@\mtcPackageWarningNoLine[W0039]{minitoc(hints)}%--- The \string\section\space command is altered after minitoc\fi\else\fi\relax\else\fi

Hence we initialize some flags:

\if@mtc@hints@\global\@mtc@toc@used@false\fi
\if@mtc@lof@used@\global\@mtc@lof@used@false\fi
\if@mtc@lot@used@\global\@mtc@lot@used@false\fi
\if@mtc@hyper@used@\fi

Then we patch the involved commands to set the corresponding flag when they are used. First, the commands for the main summaries:

\let\mtc@sv@listoffigures\listoffigures
\let\mtc@sv@listoftables\listoftables
\if@mtc@hyper@used@

And finally, we prepare the consistency tests about the calling sequences of triplets of associated commands like \doparttoc, \parttoc and \[fake\]tableofcontents, and similar: to be able to use \parttoc, a table of contents file must have been created via \[fake\]tableofcontents and splitted into parttoc files via \doparttoc.
[9] — Commented code of the minitoc package

\def\listoffigures{
\global\@mtc@lof@used@true\mtc@sv@listoffigures
\let\mtc@sv@listoftables\listoftables
\def\listoftables{
\global\@mtc@lot@used@true\mtc@sv@listoftables
\mtc@sv@fktableofcontents
\faketableofcontents
\if\@mtc@toc@used@
\mtc@sv@fklistoffigures
\fakelistoffigures
\if\@mtc@lof@used@
\mtc@sv@fklistoftables
\fakelistoftables
\if\@mtc@lot@used@
\let\mtc@sv@fktableofcontents\faketableofcontents
\def\faketableofcontents{
\global\@mtc@toc@used@true\mtc@sv@fktableofcontents
\let\mtc@sv@fklistoffigures\fakelistoffigures
\def\fakelistoffigures{
\global\@mtc@lof@used@true\mtc@sv@fklistoffigures
\let\mtc@sv@fklistoftables\fakelistoftables
\def\fakelistoftables{
\global\@mtc@lot@used@true\mtc@sv@fklistoftables
\fi
\fi
\fi
\fi
\fi
\fi
\fi
\mtc@hints@begindoc
Then, their “fake” siblings:
\let\mtc@sv@fktableofcontents\faketableofcontents
\def\faketableofcontents{
\global\@mtc@toc@used@true\mtc@sv@fktableofcontents
\let\mtc@sv@fklistoffigures\fakelistoffigures
\def\fakelistoffigures{
\global\@mtc@lof@used@true\mtc@sv@fklistoffigures
\let\mtc@sv@fklistoftables\fakelistoftables
\def\fakelistoftables{
\global\@mtc@lot@used@true\mtc@sv@fklistoftables
\fi
\fi
\fi
\fi
\fi
\mtc@hints@begindoc
And the \mtc@hints@begindoc definition is finished (it begins in section 9.81.1 on page 414):
\mtc@hints@enddoc
\AtEndDocument
The final part of the hints option is executed via \AtEndDocument. Its code is in the \mtc@hints@enddoc macro. It is a sequence of tests on the packages or classes loaded and the flags set during the first and the second parts of this option. First, we declare the \mtc@hints@enddoc macro:
\def\mtc@hints@enddoc{}

9.81.2 Final part: \mtc@hints@enddoc

\AtEndDocument
The final part of the hints option is executed via \AtEndDocument. Its code is in the \mtc@hints@enddoc macro. It is a sequence of tests on the packages or classes loaded and the flags set during the first and the second parts of this option. First, we declare the \mtc@hints@enddoc macro:
\def\mtc@hints@enddoc{}

9.81.2.1 Hint about \sect-lof|lot and the insection option

\if\@mtc@section@floats@\if\@dosectlof@used@
\if\@dosectlot@used@
\if\@sectlof@used@
\if\@sectlot@used@
\if\@mtc@section@def@
We look if some section-level lists of figures or tables have been requested.
\fi
\fi
\fi
\fi
\fi
\fi
\if\@mtc@section@def@
\fi
9.81.2.2 Final part of the consistency tests

We test if \parttoc has been used without \doparttoc, etc., for each pair of preparation/insertion commands.

\if@mtc@part@def@  For the part level commands:
  \if@parttoc@used@
    \if@doparttoc@used@ \else
      \if@mtc@part@def@ \else
        \mtcPackageWarningNoLine{minitoc(hints)}{You have used \string\parttoc, \MessageBreak but not \string\doparttoc} \@mtc@hints@given@true \fi
      \fi
    \fi
  \fi
\fi

\if@partlof@used@
  \if@dopartlof@used@ \else
    \if@mtc@part@def@ \else
      \mtcPackageWarningNoLine{minitoc(hints)}{You have used \string\partlof, \MessageBreak but not \string\dopartlof} \@mtc@hints@given@true \fi
    \fi
  \fi
\fi

\if@partlot@used@
  \if@dopartlot@used@ \else
    \if@mtc@part@def@ \else
      \mtcPackageWarningNoLine{minitoc(hints)}{You have used \string\partlot, \MessageBreak but not \string\dopartlot} \@mtc@hints@given@true \fi
    \fi
  \fi
\fi

\if@parttoc@used@
  \if@doparttoc@used@ \else
    \if@mtc@part@def@ \else
      \mtcPackageWarningNoLine{minitoc(hints)}{You have used \string\parttoc, \MessageBreak but not \string\doparttoc} \@mtc@hints@given@true \fi
    \fi
  \fi
\fi

\if@partlof@used@
  \if@dopartlof@used@ \else
    \if@mtc@part@def@ \else
      \mtcPackageWarningNoLine{minitoc(hints)}{You have used \string\partlof, \MessageBreak but not \string\dopartlof} \@mtc@hints@given@true \fi
    \fi
  \fi
\fi

\if@partlot@used@
  \if@dopartlot@used@ \else
    \if@mtc@part@def@ \else
      \mtcPackageWarningNoLine{minitoc(hints)}{You have used \string\partlot, \MessageBreak but not \string\dopartlot} \@mtc@hints@given@true \fi
    \fi
  \fi
\fi

If yes, we verify that the \placeins package \cite{placeins} has been loaded with the correct options or that the \insection option of the minitoc package has been invoked. If not, a warning is given.
\if@mtc@chapter@def@ For the chapter level commands:
\if@minitoc@used@
\if@dominitoc@used@ 6789 \if@mtc@chapter@def@
\if@minilof@used@ 6790 \if@minitoc@used@
\if@dominilof@used@ 6791 \if@minilit@used@
\if@dominilot@used@ 6792 \mtcPackageWarningNoLine[W0059]{minitoc(hints)}%
6793 \MessageBreak
6794 {You have used \string\minitoc,
6795 but not \string\dominitoc}
6796 \if@mtc@chapter@def@
6797 \fi
6798 \fi
6799 \if@minilof@used@
6800 \if@dominilof@used@\else
6801 \mtcPackageWarningNoLine[W0057]{minitoc(hints)}%
6802 \MessageBreak
6803 {You have used \string\minilof,
6804 but not \string\dominilof}
6805 \if@mtc@chapter@def@
6806 \fi
6807 \fi
6808 \if@minilot@used@
6809 \if@dominilot@used@\else
6810 \mtcPackageWarningNoLine[W0058]{minitoc(hints)}%
6811 \MessageBreak
6812 {You have used \string\minilot,
6813 but not \string\dominilot}
6814 \if@mtc@chapter@def@
6815 \fi
6816 \fi
6817 \fi

\if@mtc@section@def@ For the section level commands:
\if@secttoc@used@
\if@dosecttoc@used@ 6818 \if@mtc@section@def@
\if@sectlof@used@ 6819 \if@secttoc@used@
\if@dosectlof@used@ 6820 \if@secttoc@used@\else
6821 \mtcPackageWarningNoLine[W0065]{minitoc(hints)}%
6822 \MessageBreak
6823 {You have used \string\secttoc,
6824 but not \string\dosecttoc}
6825 \if@mtc@section@def@
6826 \fi
6827 \fi
6828 \if@sectlof@used@
9.81.2.3  Check if the main tables have been prepared (first part)

Now, we test if a \doparttoc macro has been called but without any matching \parttoc, hence it is a vain call. We do the same for each analog command.

\if@mtc@part@def@ Part level commands:
\if@doparttoc@used@ 6846 \if@mtc@part@def@ 6847 \if@doparttoc@used@ 6848 \if@mtc@part@def@ 6849 \if@doparttoc@used@ 6850 \if@mtc@part@def@ 6851 \if@doparttoc@used@ 6852 \if@mtc@part@def@ 6853 \if@doparttoc@used@ 6854 \if@mtc@part@def@ 6855 \if@doparttoc@used@ 6856 \if@mtc@part@def@ 6857 \if@doparttoc@used@ 6858 \if@mtc@part@def@ 6859 \if@doparttoc@used@ 6860 \if@mtc@part@def@ 6861 \if@doparttoc@used@ 6862 \if@mtc@part@def@ 6863 \if@doparttoc@used@ 6864 \if@mtc@part@def@ 6865 \if@doparttoc@used@ 6866 \if@mtc@part@def@ 6867 \if@doparttoc@used@ 6868 \if@mtc@part@def@ 6869 \if@doparttoc@used@ 6870 \if@mtc@part@def@
Chapter level commands:

\if@mtc@chapter@def@ \if@dominitoc@used@ \if@minitoc@used@ \if@mtc@hints@given@ \if@dominilof@used@ \if@minilof@used@ \if@dominilot@used@ \if@minilot@used@ \else \mtcPackageWarningNoLine[W0078]{minitoc(hints)}\MessageBreak {You have used \string\dominitoc, \string\dominilof, \string\dominilot} \else \mtcPackageWarningNoLine[W0079]{minitoc(hints)}\MessageBreak {You have used \string\dominitoc, \string\dominilof, \string\dominilot} \fi \else \mtcPackageWarningNoLine[W0080]{minitoc(hints)}\MessageBreak {You have used \string\dominitoc, \string\dominilof, \string\dominilot} \fi \else \mtcPackageWarningNoLine[W0081]{minitoc(hints)}\MessageBreak {You have used \string\dosecttoc, \string\secttoc, \string\sectlof, \string\sectlot} \fi \else \mtcPackageWarningNoLine[W0082]{minitoc(hints)}\MessageBreak {You have used \string\dosecttoc, \string\secttoc, \string\sectlof, \string\sectlot} \fi \else \mtcPackageWarningNoLine[W0083]{minitoc(hints)}\MessageBreak {You have used \string\dosecttoc, \string\secttoc, \string\sectlof, \string\sectlot} \fi \fi \fi \fi \fi

Section level commands:

\if@mtc@section@def@ \if@dosecttoc@used@ \if@secttoc@used@ \if@mtc@chapter@def@ \if@dosectlof@used@ \if@sectlof@used@ \if@dosectlot@used@ \if@sectlot@used@ \else \mtcPackageWarningNoLine[W0084]{minitoc(hints)}\MessageBreak {You have used \string\dosecttoc, \string\secttoc, \string\sectlof, \string\sectlot} \else \mtcPackageWarningNoLine[W0085]{minitoc(hints)}\MessageBreak {You have used \string\dosecttoc, \string\secttoc, \string\sectlof, \string\sectlot} \fi \else \mtcPackageWarningNoLine[W0086]{minitoc(hints)}\MessageBreak {You have used \string\dosecttoc, \string\secttoc, \string\sectlof, \string\sectlot} \fi \else \mtcPackageWarningNoLine[W0087]{minitoc(hints)}\MessageBreak {You have used \string\dosecttoc, \string\secttoc, \string\sectlof, \string\sectlot} \fi \else \mtcPackageWarningNoLine[W0088]{minitoc(hints)}\MessageBreak {You have used \string\dosecttoc, \string\secttoc, \string\sectlof, \string\sectlot} \fi \else \mtcPackageWarningNoLine[W0089]{minitoc(hints)}\MessageBreak {You have used \string\dosecttoc, \string\secttoc, \string\sectlof, \string\sectlot} \fi \else \mtcPackageWarningNoLine[W0090]{minitoc(hints)}\MessageBreak {You have used \string\dosecttoc, \string\secttoc, \string\sectlof, \string\sectlot} \fi \else \mtcPackageWarningNoLine[W0091]{minitoc(hints)}\MessageBreak {You have used \string\dosecttoc, \string\secttoc, \string\sectlof, \string\sectlot} \fi \else \mtcPackageWarningNoLine[W0092]{minitoc(hints)}\MessageBreak {You have used \string\dosecttoc, \string\secttoc, \string\sectlof, \string\sectlot} \fi \else \mtcPackageWarningNoLine[W0093]{minitoc(hints)}\MessageBreak {You have used \string\dosecttoc, \string\secttoc, \string\sectlof, \string\sectlot} \fi \else \mtcPackageWarningNoLine[W0094]{minitoc(hints)}\MessageBreak {You have used \string\dosecttoc, \string\secttoc, \string\sectlof, \string\sectlot} \fi \else \mtcPackageWarningNoLine[W0095]{minitoc(hints)}\MessageBreak {You have used \string\dosecttoc, \string\secttoc, \string\sectlof, \string\sectlot} \fi \else \mtcPackageWarningNoLine[W0096]{minitoc(hints)}\MessageBreak {You have used \string\dosecttoc, \string\secttoc, \string\sectlof, \string\sectlot} \fi \else \mtcPackageWarningNoLine[W0097]{minitoc(hints)}\MessageBreak {You have used \string\dosecttoc, \string\secttoc, \string\sectlof, \string\sectlot} \fi \else \mtcPackageWarningNoLine[W0098]{minitoc(hints)}\MessageBreak {You have used \string\dosecttoc, \string\secttoc, \string\sectlof, \string\sectlot} \fi
9.81.2.4 Check if the main tables have been prepared (second part)

Another consistency test verifies that if the macro \parttoc has been called, then the macro \tableofcontents or \faketableofcontents has also been called (to create the necessary contents file); similar tests are made for the other mini-table commands.

\if@mtc@part@def\if@mtc@hints@given\if@parttoc@used\if@mtc@toc@used\if@partlof@used\if@mtc@lof@used\if@partlot@used\if@mtc@lot@used
\fi\fi\fi\fi\fi\fi\fi\fi\fi

Part level commands:
[9] — Commented code of the minitoc package

429

\if@partlot@used@
\if@mtc@lot@used@\else
6958
\mtcPackageWarningNoLine[W0070]{minitoc(hints)}%
6959
{You have used \string\partlot\space but not
6960
\MessageBreak
6961
\string\listoftables
6962
\MessageBreak
6963
nor \string\fakelistoftables}
6964
\@mtc@hints@given@true
6965
\fi
6966 \fi
6967 \fi
6956

6957

\if@mtc@chapter@def@
\if@mtc@hints@given@
\if@minitoc@used@
\ifmtc@toc@used@
\if@minilof@used@
\ifmtc@lof@used@
\if@minilot@used@
\ifmtc@lot@used@

Chapter level commands:

W0068
W0066

6968 \if@mtc@chapter@def@

\if@minitoc@used@
6970
\if@mtc@toc@used@\else
6971
\mtcPackageWarningNoLine[W0068]{minitoc(hints)}%
6972
{You have used \string\minitoc\space but not
6973
\MessageBreak
6974
\string\tableofcontents
6975
\MessageBreak
6976
nor \string\faketableofcontents}
6977
\@mtc@hints@given@true
6978
\fi
6979 \fi
6980 \if@minilof@used@
6981
\if@mtc@lof@used@\else
6982
\mtcPackageWarningNoLine[W0066]{minitoc(hints)}%
6983
{You have used \string\minilof\space but not
6984
\MessageBreak
6985
\string\listoffigures
6986
\MessageBreak
6987
nor \string\fakelistoffigures}
6988
\@mtc@hints@given@true
6989
\fi
6990 \fi
6991 \if@minilot@used@
6992
\if@mtc@lot@used@\else
6993
\mtcPackageWarningNoLine[W0067]{minitoc(hints)}%
6994
{You have used \string\minilot\space but not
6995
\MessageBreak
6996
\string\listoftables
6997
\MessageBreak
6998
nor \string\fakelistoftables}
6999
\@mtc@hints@given@true
7000
\fi
7001 \fi
7002 \fi
6969

W0067


Section level commands:

\if@mtc@section@def@ \if@mtc@hints@given@ \if@secttoc@used@ \if@mtc@toc@used@ \if@sectlof@used@ \if@mtc@lof@used@ \if@sectlot@used@ \if@mtc@lot@used@

Section level commands:

\if@mtc@section@def@ \if@secttoc@used@ \if@mtc@toc@used@ \if@sectlof@used@ \if@mtc@lof@used@ \if@sectlot@used@ \if@mtc@lot@used@

\mtcPackageWarningNoLine[W0074]{minitoc(hints)}% (You have used \string\secttoc\space but not \MessageBreak \string\tableofcontents \MessageBreak nor \string\faketableofcontents)
\@mtc@hints@given@true \fi
\fi
\fi
\fi
\mtcPackageWarningNoLine[W0072]{minitoc(hints)}% (You have used \string\sectlof\space but not \MessageBreak \string\listoffigures \MessageBreak nor \string\fakelistoffigures)
\@mtc@hints@given@true \fi
\fi
\fi
\fi

\mtc@hints@checklongext
\if@mtc@longext@
\if@mtc@part@def@ \value
\if@mtc@chapter@def@ \if@mtc@section@def@ 

9.81.2.5 Check the number of mini-tables, in case of short extensions

If short extensions are used, you can use only 99 mini-tables of each kind. If more are created, the auxiliary files can be overwritten: the hundredth minitoc file \jobname.U100 has its name truncated to \jobname.U10, which is already the tenth minitoc file. Thus, we need a hint to signal this situation. The code is rather simple, but the remedy is bitter and costly: either use a better operating system \footnote{On the long term, a good investment.}, either redesign the document.

\def\mtc@hints@checklongext{% 
\if@mtc@longext@ 
\fi 
\fi 
\fi 
\fi 

9.81.2.6 Final part of the hint about the \texttt{sectsty} package

We test if \texttt{sectsty} has been loaded before (correct) or after (incorrect) \texttt{minitoc}. See section 9.9.1 on page 275.

9.81.2.7 Final part of the hint about the \texttt{varsects} package

We test if \texttt{varsects} has been loaded before (correct) or after (incorrect) \texttt{minitoc}. See
section 9.9.2 on page 275.

\if@mtc@varsectsLoaded@\else
\if@mtc@varsectsLoaded@a@
\mtcPackageWarningNoLine[W0038]{minitoc(hints)}%
{The varsects package should be
\MessageBreak
loaded BEFORE the minitoc package}
\@mtc@hints@given@true
\fi
\fi

9.81.2.8 Final part of the hint about the fncychap package

\if@mtc@fncychapLoaded@
\if@mtc@fncychapLoaded@a@
\if@mtc@hints@given@
We test if fncychap has been loaded before (correct) or after (incorrect) minitoc. See
section 9.9.3 on page 275.
\fi
\fi
\fi

\if@mtc@HgcLoaded@
\if@mtc@HgcLoaded@a@
\if@mtc@hints@given@
We test if hangcaption has been loaded before (correct) or after (incorrect) minitoc. See
section 9.9.4 on page 275.
\fi
\fi
\fi
9.81.2.10 Final part of the hint about the quotchap package

We test if quotchap has been loaded before (correct) or after (incorrect) minitoc. See section 9.9.5 on page 276.

\if@mtc@quotchapLoaded\else \if@mtc@quotchapLoaded@a \if@mtc@hints@given
\mtcPackageWarningNoLine[W0087]{minitoc(hints)}% The quotchap package should be loaded BEFORE the minitoc package\@mtc@hints@given@true \fi \fi

9.81.2.11 Final part of the hint about the romannum package

We test if romannum has been loaded before (correct) or after (incorrect) minitoc. See section 9.9.6 on page 276.

\if@mtc@romannumLoaded\else \if@mtc@romannumLoaded@a \if@mtc@hints@given
\mtcPackageWarningNoLine[W0088]{minitoc(hints)}% The romannum package should be loaded BEFORE the minitoc package\@mtc@hints@given@true \fi \fi

9.81.2.12 Final part of the hint about the sfheaders package

We test if sfheaders has been loaded before (correct) or after (incorrect) minitoc. See section 9.9.7 on page 276.

\if@mtc@sfheadersLoaded\else \if@mtc@sfheadersLoaded@a \if@mtc@hints@given
\mtcPackageWarningNoLine[W0089]{minitoc(hints)}% The sfheaders package should be loaded BEFORE the minitoc package\@mtc@hints@given@true \fi \fi
9.81.2.13 Final part of the hint about the alnumsec package

We test if alnumsec has been loaded before (correct) or after (incorrect) minitoc. See section 9.9.8 on page 276.

\if@mtc@alnumsecLoaded@\else
\if@mtc@alnumsecLoaded@a@
\mtcPackageWarningNoLine[W0090]{minitoc(hints)}%
\MessageBreak
 loaded BEFORE the minitoc package
\@mtc@hints@given@true
\fi
\fi

9.81.2.14 Final part of the hint about the captcont package

We test if captcont has been loaded before (correct) or after (incorrect) minitoc. See section 9.9.9 on page 277.

\if@mtc@captcontLoaded@\else
\if@mtc@captcontLoaded@a@
\mtcPackageWarningNoLine[W0091]{minitoc(hints)}%
\MessageBreak
 loaded BEFORE the minitoc package
\@mtc@hints@given@true
\fi
\fi

9.81.2.15 Final part of the hint about the caption package

We test if caption has been loaded before (correct) or after (incorrect) minitoc. See section 9.9.10 on page 277.

\if@mtc@captionLoaded@\else
\if@mtc@captionLoaded@a@
\mtcPackageWarningNoLine[W0033]{minitoc(hints)}%
\MessageBreak
 loaded BEFORE the minitoc package
\@mtc@hints@given@true
\fi
\fi
9.81.2.16 Final part of the hint about the caption2 package

We test if caption2 has been loaded before (correct) or after (incorrect) minitoc. See section 9.9.11 on page 277.

\if@mtc@captionIILoaded@ \else
  \if@mtc@captionIILoaded@a@
    \mtcPackageWarningNoLine{W0034}{minitoc(hints)}%
    \MessageBreak
    loaded BEFORE the minitoc package
  \fi
  \@mtc@hints@given@true
  \fi
\fi

9.81.2.17 Final part of the hint about the ccaption package

We test if ccaption has been loaded before (correct) or after (incorrect) minitoc. See section 9.9.12 on page 277.

\if@mtc@ccaptionLoaded@ \else
  \if@mtc@ccaptionLoaded@a@
    \mtcPackageWarningNoLine{W0035}{minitoc(hints)}%
    \MessageBreak
    loaded BEFORE the minitoc package
  \fi
  \@mtc@hints@given@true
  \fi
\fi

9.81.2.18 Final part of the hint about the mcaption package

We test if mcaption has been loaded before (correct) or after (incorrect) minitoc. See section 9.9.13 on page 278.

\if@mtc@mcaptionLoaded@ \else
  \if@mtc@mcaptionLoaded@a@
    \mtcPackageWarningNoLine{W0036}{minitoc(hints)}%
    \MessageBreak
    loaded BEFORE the minitoc package
  \fi
  \@mtc@hints@given@true
  \fi
\fi
9.81.2.19 Final part of the hint about the float package

We test if float has been loaded. See section 9.9.14 on page 278.

\if@mtc@floatLoaded@
\if@mtc@hints@given@
\mtcPackageInfo[@0053]{minitoc(hints)}%
{You have loaded the float package;}
\MessageBreak
\MessageBreak
please be aware that the minitoc package
\MessageBreak
facilities can not be used for new types
\MessageBreak
of floats defined by the float package\gobble}
\@mtc@hints@given@true
\fi
\fi

9.81.2.20 Final part of the hint about the floatrow package

We test if floatrow has been loaded. See section 9.9.15 on page 278.

\if@mtc@floatrowLoaded@
\if@mtc@hints@given@
\mtcPackageInfo[@0053]{minitoc(hints)}%
{You have loaded the floatrow package;}
\MessageBreak
\MessageBreak
please be aware that the minitoc package
\MessageBreak
facilities can not be used for new types
\MessageBreak
of floats defined by the floatrow package\gobble}
\@mtc@hints@given@true
\fi
\fi

9.81.2.21 Final part of the hint about the trivfloat package

We test if trivfloat has been loaded. See section 9.9.16 on page 278.

\if@mtc@trivfloatLoaded@
\if@mtc@hints@given@
\mtcPackageInfo[@0053]{minitoc(hints)}%
{You have loaded the trivfloat package;}
\MessageBreak
\MessageBreak
please be aware that the minitoc package
\MessageBreak
facilities can not be used for new types
\MessageBreak
of floats defined by the trivfloat package\gobble}
\@mtc@hints@given@true
\fi
\fi
9.81.2.22 Final part of the hint about the rotfloat package

We test if rotfloat has been loaded. See section 9.9.17 on page 278.

\if@mtc@rotfloatLoaded@ \if@mtc@hints@given@
\mtcPackageInfo[I0053]{minitoc(hints)}% You have loaded the rotfloat package;
\MessageBreak please be aware that the minitoc package facilities can not be used for new types of floats defined by the rotfloat package\@gobble}
\@mtc@hints@given@true \fi
\fi

9.81.2.23 Check if empty mini-tables have been detected

We test for each kind of mini-tables.

\if@mtc@empty@parttoc@ For parttocs:
\if@mtc@empty@parttoc@
\mtcPackageWarningNoLine[W0046]{minitoc(hints)}% You have attempted to insert empty parttocs}
\@mtc@hints@given@true \fi
\fi

\if@mtc@empty@partlof@ For partlofs:
\if@mtc@empty@partlof@
\mtcPackageWarningNoLine[W0044]{minitoc(hints)}% You have attempted to insert empty partlofs}
\@mtc@hints@given@true \fi
\fi

\if@mtc@empty@partlot@ For partlots:
\if@mtc@empty@partlot@
\mtcPackageWarningNoLine[W0045]{minitoc(hints)}%
\fi
\fi
\if@mtc@empty@minitoc@ For minitocs:

\if@mtc@empty@minitoc@
    \mtcPackageWarningNoLine[W0043]{minitoc(hints)}%
    {You have attempted to insert empty partlots}
    \@mtc@hints@given@true
\fi

\if@mtc@empty@minilof@ For minilofs:

\if@mtc@empty@minilof@
    \mtcPackageWarningNoLine[W0041]{minitoc(hints)}%
    {You have attempted to insert empty minilofs}
    \@mtc@hints@given@true
\fi

\if@mtc@empty@minilot@ For minilots:

\if@mtc@empty@minilot@
    \mtcPackageWarningNoLine[W0042]{minitoc(hints)}%
    {You have attempted to insert empty minilots}
    \@mtc@hints@given@true
\fi

\if@mtc@empty@secttoc@ For secttocs:

\if@mtc@empty@secttoc@
    \mtcPackageWarningNoLine[W0049]{minitoc(hints)}%
    {You have attempted to insert empty secttocs}
    \@mtc@hints@given@true
\fi


9.8.1.24 Check if obsolete commands have been used

This hint is just a reminder if you have used obsolete commands, which are also signalled in the document.log file.

\if@firstpartis@used@ Obsolete macro \firstpartis:

\if@firstchapteris@used@ Obsolete macro \firstchapteris:
9.81.2.25 Check if some hints have been written

We come at the end of the third part of the hints option: if problems have been detected, a warning is displayed; the warning is not displayed but only written in the document.log file if no problems have been detected. And we terminate the \mtc@hints@enddoc macro by a closing brace.

\if@mtc@hints@given@
\mtc@hints@enddoc
\else
\fi

9.82 Processing of options

First, if possible, we apply the default language option, english:

\InputIfFileExists{english.mld}%
(\ExecuteOptions{english})%
Else, we signal a severe error and provide the missing default titles:

\mtcPackageError
\providecommand
\ptctitle
\plftitle
\plttitle
\mtctitle
\mlftitle
\mlttitle
\stctitle
\slftitle
\slttitle

Else, we signal a severe error and provide the missing default titles:

\mtcPackageError{E0036}{minitoc}
\providecommand
\ptctitle
\plftitle
\plttitle
\mtctitle
\mlftitle
\mlttitle
\stctitle
\slftitle
\slttitle

Then, we execute all requested options: for most options, it is just setting a flag, or loading a
file for the language options.

\ProcessOptions*

We now examine the flags for some options and execute the necessary actions.

### 9.8.2.1 Processing the \texttt{insection} option

For the \texttt{insection} option, we load the \texttt{placeins} package [15] with its options \texttt{verbose} and
\texttt{section}, after the \texttt{flafter} package (described in [288] and [330, page 286]); the correct loading
is verified:

\if@mtc@ss@insection@
\RequirePackage
\@ifpackageloaded
\if@mtc@placeinsLoaded@
\fi
9.82.2 Processing the notoccite option

\if@mtc@notoccite\RequirePackage{notoccite}\fi

For the notoccite option, we just load the notoccite package [14]:

\if@mtc@notoccite\RequirePackage{notoccite}\fi

9.82.3 Processing the listfiles option

\mtc@maf
\if@mtc@longext\mtc@maf@long\else\mtc@maf@short\fi
\IfFileExists{\jobname.mtc}{\mtc@addtomaf{\jobname.mtc}}{}
\IfFileExists{\jobname.mtc0}{\mtc@addtomaf{\jobname.mtc0}}{}
\@ifundefined{c@ptc}{}{\loop\ifnum\c@ptc>\z@\relax
\mtc@addtomaf{\jobname.ptc\arabic{ptc}}
\mtc@addtomaf{\jobname.plf\arabic{ptc}}
\mtc@addtomaf{\jobname.plt\arabic{ptc}}
\advance\c@ptc\m@ne\repeat}
\@ifundefined{c@mtc}{}{\loop\ifnum\c@mtc>\z@\relax
\mtc@addtomaf{\jobname.mtc\arabic{mtc}}
\mtc@addtomaf{\jobname.mlf\arabic{mtc}}
\mtc@addtomaf{\jobname.mlt\arabic{mtc}}
\advance\c@mtc\m@ne\repeat}
\@ifundefined{c@stc}{}{\loop\ifnum\c@stc>\z@\relax
\mtc@addtomaf{\jobname.stc\arabic{stc}}
\mtc@addtomaf{\jobname.slf\arabic{stc}}
\mtc@addtomaf{\jobname.slt\arabic{stc}}
\advance\c@stc\m@ne\repeat}

Some users could made a cleanup using this file as a list of files to delete, so it must not be in the list.
9.82.4 Processing the hints option

For the hints option, we set its first part in an \AtBeginDocument block and its third (last) part in an \AtEndDocument block:

\if@mtc@hints@
\AtBeginDocument{\mtc@hints@begindoc}%
\AtEndDocument{\mtc@hints@enddoc}%
\fi

9.82.5 Saving the sectioning commands

And, at least, we save the definitions of sectioning commands (and of their unstarrred and starred branches), for comparisons (this is a part of the hints option executed in the preamble):

\if@mtc@listfiles@
\AtEndDocument{\mtc@maf}
\fi

\if@mtc@hints@
\AtBeginDocument{\mtc@hints@begindoc}%
\AtEndDocument{\mtc@hints@enddoc}%
\fi

For the \part command:

\ifdefined{\mtc@hints@part}
\part
\fi
[9] — Commented code of the minitoc package

\@ifundefined{chapter}{}{\let\mtc@hints@chapter\chapter}
\let\mtc@hints@@chapter\@chapter
\let\mtc@hints@@schapter\@schapter

For the \chapter command:

\@ifundefined{section}{}{\let\mtc@hints@section\section}
\let\mtc@hints@@sect\@sect
\let\mtc@hints@@ssect\@ssect

For the \section command:

9.83 Trapping the undefined preparation and insertion commands

\mtc@classck\mtcPackageError

It may happen that you use a preparation command (like \dominitoc) or an insertion command (like \dominitoc) in a document using a class where that command is not available (like \article). To get a better diagnostic for such errors, we intercept such commands by providing a default definition which just emits an error message. These default definitions are made in an \AtBeginDocument block.

\def\mtc@classck#1{%\mtcPackageError[E0037]{minitoc}%{The \csname #1@endcsname\space command is incompatible with the document class}%{Correct the source code.}\MessageBreak{Type <return> and rerun LaTeX}%}%

\AtBeginDocument An \AtBeginDocument bloc:

\providecommand\doparttoc\dopartlof\dopartlot

Part-level preparation commands:

\providecommand\doparttoc[1][1]{\mtc@classck{doparttoc}}%
\providecommand\dopartlof[1][1]{\mtc@classck{dopartlof}}%
\providecommand\dopartlot[1][1]{\mtc@classck{dopartlot}}%
Chapter-level preparation commands:
\providecommand{\dominitoc}[1][1]{\mtc@classck{dominitoc}}%
\providecommand{\dominilof}[1][1]{\mtc@classck{dominilof}}%
\providecommand{\dominilot}[1][1]{\mtc@classck{dominilot}}%

Section-level preparation commands:
\providecommand{\dosecttoc}[1][1]{\mtc@classck{dosecttoc}}%
\providecommand{\dosectlof}[1][1]{\mtc@classck{dosectlof}}%
\providecommand{\dosectlot}[1][1]{\mtc@classck{dosectlot}}%

Part-level insertion commands:
\providecommand{\parttoc}[1][1]{\mtc@classck{parttoc}}%
\providecommand{\partlof}[1][1]{\mtc@classck{partlof}}%
\providecommand{\partlot}[1][1]{\mtc@classck{partlot}}%

Chapter-level insertion commands:
\providecommand{\minitoc}[1][1]{\mtc@classck{minitoc}}%
\providecommand{\minilof}[1][1]{\mtc@classck{minilof}}%
\providecommand{\minilot}[1][1]{\mtc@classck{minilot}}%

Section-level insertion commands:
\providecommand{\secttoc}[1][1]{\mtc@classck{secttoc}}%
\providecommand{\sectlof}[1][1]{\mtc@classck{sectlof}}%
\providecommand{\sectlot}[1][1]{\mtc@classck{sectlot}}%
}

And the package is terminated.

\jobname
\input This short file is necessary to create the french documentation. Its rôle is to set \jobname to minitoc-fr in place of minitoc. As minitoc.ins generates the minitoc.lan and minitoc-fr.lan files which set a language number \LANG, and minitoc.dtx reads then the \jobname.lan file, the documentation can be in several languages (english and french here) in minitoc.dtx, the language being selected by \ifcase\LANG\relax ... \or\relax ... \fi constructs. The \relax primitives are necessary to avoid bad surprises.
[9] — Commented code of the minitoc package

7451 \{\minitoc-fr\}
7452 \ProvidesFile{minitoc-fr.dtx}\%  
7453 \[2018/07/12 minitoc v62 french documentation start file\]
7454 \input{minitoc.dtx}
7455 \{/minitoc-fr\}
Chapter 10

Commented code of the mtcoff package

Contents

10.1 Why mtcoff? .................................................. 448
10.2 Identification of the package .............................. 448
10.3 Faking counters and dimensions ........................... 449
10.4 Faking simple commands .................................. 452
10.5 Faking commands with one optional argument ............... 453
10.6 Faking flags .................................................. 454
10.7 Disabling the internal commands ......................... 454
10.8 Disabling the font commands ............................... 455
10.9 Disabling the \mtcset... commands ......................... 456
10.10 Disabling the \mtcopolym... commands .................. 456
10.11 Disabling the new \l@... commands ....................... 456
10.12 Ignore the obsolete commands ............................ 457
10.13 Disabling the \mtcselectlanguage and \mtcloadmlo commands 457
10.14 Disabling the commands for the horizontal rules ............. 457
10.15 Disabling the commands for the page numbers .............. 458
10.16 Disabling the mini-table features commands ............... 458
10.17 Disabling miscellaneous flags and commands .............. 459
10.18 Caution for some commands ............................... 460
10.19 Disabling commands for “coffee” ......................... 461
10.20 Disabling the \mtchideinmain... environments ............. 461
10.21 Inhibition of the \mtc@[save|restore]XXXdepth internal commands 462
10.22 Disabling the \mtcfixglossary command .................... 462
10.23 Disabling the \mtcfixindex command ....................... 462
10.24 Disabling the \mtcfixnomenclature command ................ 462
10.25 Disabling the \addstarred... commands ................... 463
10.1 Why mtcoff?

The `minitoc` package\cite{156, 157} requires that the user inserts many commands in the source code of her/his document, and not only into the preamble of the document. Hence the concept of a replacement package, `mtcoff` (means “minitoc off”), which substitutes to all commands and environments of the `minitoc` package some alternative commands and environments with the same names and syntaxes, but doing nothing (except emitting some harmless warnings, for special cases). This way, to turn off easily the `minitoc` package, you just have to write, in the preamble of your document, something like:

\begin{verbatim}
\usepackage[...options...]{minitoc}
%\usepackage{mtcoff}
\end{verbatim}

then the `minitoc` package is activated with the specified options. If you modify this two lines this way:

\begin{verbatim}
%\usepackage[...options...]{minitoc}
\usepackage{mtcoff}
\end{verbatim}

then the `minitoc` package is desactivated and all its commands and environments are ignored. This is much easier, faster and safer than commenting out all the commands and environments of `minitoc`. Moreover, this operation is reversible.

10.2 Identification of the package

```
\NeedsTeXFormat
\ProvidesPackage
\RequirePackage\mtcmess
\ProvidesPackage{mtcoff}[2018/07/12 v62 The mtcoff package]
\RequirePackage{mtcmess}[2006/03/14]
```

First, we identify the package and check the version of \LaTeX\footnote{This checking is not really useful for the mtcoff package itself, but it is good to check that your version of \LaTeX is not too old to support minitoc.}; we need the `mtcmess` package to write messages with unique identifiers.
10.3 Faking counters and dimensions

\[ \text{As minitoc declares some counters and dimensions registers, we fake them using } \text{\texttt{\count@}} \text{ or \texttt{\dimen@}. For \texttt{\mtcskipamount}, we must use its default definition, \texttt{\bigskipamount}.} \]

\[ \text{\begin{verbatim} \let\c@minitocdepth\count@ \let\mtcindent\dimen@ \let\mtcskipamount\bigskipamount \let\c@parttocdepth\count@ \let\ptcindent\dimen@ \let\c@secttocdepth\count@ \let\stcindent\dimen@ \end{verbatim}} \]

\[ \text{\begin{verbatim} \let\c@mtc\count@ \let\c@ptc\count@ \let\c@stc\count@ \end{verbatim}} \]

\[ \text{As \texttt{\mtcgapbeforeheads} and \texttt{\mtcgapafterheads} receive their default values:} \]

\[ \text{\begin{verbatim} \def\mtcgapbeforeheads{50\p@} \def\mtcgapafterheads{40\p@} \end{verbatim}} \]

\[ \text{\begin{verbatim} \@ifundefined{part}{}{\texttt{\mtcPackageWarning[F0008]{mtcoff}{The macro \string\kernafterparttoc should not be used out of context}}} \kern-1.\baselineskip\kern.5ex} \end{verbatim}} \]

\[ \text{\begin{verbatim} \@ifundefined{part}{}{\texttt{\mtcPackageWarning[F0008]{mtcoff}{The macro \string\kernafterpartlof should not be used out of context}}} \kern-1.\baselineskip\kern.5ex} \end{verbatim}} \]

\[ \text{\begin{verbatim} \@ifundefined{part}{}{\texttt{\mtcPackageWarning[F0008]{mtcoff}{The macro \string\kernafterpartlot should not be used out of context}}} \kern-1.\baselineskip\kern.5ex} \end{verbatim}} \]
\MessageBreak
should not be used out of context
\MessageBreak\kern-1.\baselineskip\kern.5ex\%
\ifundefined{chapter}{%\ifundefined{section}{%\ifundefined{section}{%\def\kernaftersecttoc{%\mtcoffwarn@true\mtcPackageWarning[F0008]{mtcoff}{The macro \string\kernaftersecttoc should not be used out of context}\MessageBreak\kern-1.\baselineskip\kern.5ex\%\def\kernaftersectlof{%\mtcoffwarn@true\mtcPackageWarning[F0008]{mtcoff}{The macro \string\kernaftersectlof should not be used out of context}\MessageBreak\kern-1.\baselineskip\kern.5ex\%\def\kernaftersectlot{%\mtcoffwarn@true\mtcPackageWarning[F0008]{mtcoff}{The macro \string\kernaftersectlot should not be used out of context}\MessageBreak\kern-1.\baselineskip\kern.5ex\}%\}%}%}%{\def\kernafterminitoc{%\mtcoffwarn@true\mtcPackageWarning[F0008]{mtcoff}{The macro \string\kernafterminitoc should not be used out of context}\MessageBreak\kern-.5\baselineskip\kern.0ex\%\def\kernafterminilof{%\mtcoffwarn@true\mtcPackageWarning[F0008]{mtcoff}{The macro \string\kernafterminilof should not be used out of context}\MessageBreak\kern-1.\baselineskip\kern.0ex\%\def\kernafterminilot{%\mtcoffwarn@true\mtcPackageWarning[F0008]{mtcoff}{The macro \string\kernafterminilot should not be used out of context}\MessageBreak\kern-1.\baselineskip\kern.0ex\%\def\kernafterminilot%}{%\kern-1.\baselineskip\kern.5ex\}%\}%}
We must define the macros for the horizontal offsets of the mini-tables. The default values are used. We must issue a warning if one of these macros is used.
10.4 Faking simple commands

Some user commands are easy to fake:
\faketoc
\fakelistoffigures
\fakelistoftables
\mtcskip

\let\faketoc\relax
\let\fakelistoffigures\relax
\let\fakelistoftables\relax
\let\mtcskip\relax
The following commands are not directly called by the user, in normal circumstances, but must be faked:

\partend \let\partend\relax \
\partbegin \let\partbegin\relax \
\chapterend \let\chapterend\relax \
\chapterbegin \let\chapterbegin\relax \
\sectend \let\sectend\relax \
\sectbegin \let\sectbegin\relax 

10.5 Faking commands with one optional argument

\gobbleopt@ The user commands with an optional argument are faked using the internal \LaTeX macro \@ifnextchar (to get the optional argument) and the new utility command \gobbleopt@.

\@ifnextchar Commands for part level mini-tables:
\doparttoc \def\doparttoc{\@ifnextchar[\gobbleopt@}{\gobbleopt@[l]}
\dopartlof \def\dopartlof{\@ifnextchar[\gobbleopt@}{\gobbleopt@[l]}
\dopartlot \def\dopartlot{\@ifnextchar[\gobbleopt@}{\gobbleopt@[l]}
\parttoc \def\parttoc{\@ifnextchar[\gobbleopt@}{\gobbleopt@[l]}
\partlof \def\partlof{\@ifnextchar[\gobbleopt@}{\gobbleopt@[l]}
\partlot \def\partlot{\@ifnextchar[\gobbleopt@}{\gobbleopt@[l]}

\@ifnextchar Commands for chapter level mini-tables:
\dominitoc \def\dominitoc{\@ifnextchar[\gobbleopt@}{\gobbleopt@[l]}
\dominilof \def\dominilof{\@ifnextchar[\gobbleopt@}{\gobbleopt@[l]}
\dominilot \def\dominilot{\@ifnextchar[\gobbleopt@}{\gobbleopt@[l]}
\minitoc \def\minitoc{\@ifnextchar[\gobbleopt@}{\gobbleopt@[l]}
\minilof \def\minilof{\@ifnextchar[\gobbleopt@}{\gobbleopt@[l]}
\minilot \def\minilot{\@ifnextchar[\gobbleopt@}{\gobbleopt@[l]}

Basic adjustment commands are also easy:
\adjustptc \newcommand{\adjustptc}{\relax}
\adjustmtc \newcommand{\adjustmtc}{\relax}
\adjuststc \newcommand{\adjuststc}{\relax}
\decrementptc \let\decrementptc\relax \let\incrementptc\relax
\decrementmtc \let\decrementmtc\relax \let\incrementmtc\relax
\decrementstc \let\decrementstc\relax \let\incrementstc\relax
\incrementptc \let\incrementptc\relax \let\decrementptc\relax
\incrementmtc \let\incrementmtc\relax \let\decrementmtc\relax
\incrementstc \let\incrementstc\relax \let\decrementstc\relax
Commented code of the \texttt{mtoff} package

\texttt{\@ifnextchar} Commands for section level mini-tables:
\texttt{\dosecttoc} 7657 \def\dosecttoc{\@ifnextchar[]{{\gobbleopt@}{\gobbleopt@[1]}}}
\texttt{\dosectlof} 7658 \def\dosectlof{\@ifnextchar[]{{\gobbleopt@}{\gobbleopt@[1]}}}
\texttt{\secttoc} 7659 \def\secttoc{\@ifnextchar[]{{\gobbleopt@}{\gobbleopt@[1]}}}
\texttt{\sectlof} 7660 \def\sectlof{\@ifnextchar[]{{\gobbleopt@}{\gobbleopt@[1]}}}
\texttt{\sectlot} 7661 \def\sectlot{\@ifnextchar[]{{\gobbleopt@}{\gobbleopt@[1]}}}
\texttt{\sectlot} 7662 \def\sectlot{\@ifnextchar[]{{\gobbleopt@}{\gobbleopt@[1]}}}

\texttt{\@ifnextchar} Command \texttt{\mtcprepare}:
\texttt{\mtcprepare} 7663 \def\mtcprepare{\@ifnextchar[]{{\gobbleopt@}{\gobbleopt@[1]}}}

10.6 Faking flags

\texttt{\ifinparttoc} We defines flags which were true when inside a mini-table of the matching type, false outside.
\texttt{\ifinpartlof} 7664 \newif\ifinparttoc\inparttocfalse%
\texttt{\ifinpartlot} 7665 \newif\ifinpartlof\inpartloffalse%
\texttt{\ifinminitoc} 7666 \newif\ifinpartlot\inpartlotfalse%
\texttt{\ifinminilof} 7667 \newif\ifinsecttoc\insecttocfalse%
\texttt{\ifinminilot} 7668 \newif\ifinsectlof\insectloffalse%
\texttt{\ifinsecttoc} 7669 \newif\ifinsectlot\insectlotfalse%
\texttt{\ifinsectlof} 7670 \newif\ifinminitoc\inminitocfalse%
\texttt{\ifinminilot} 7671 \newif\ifinminilof\inminiloffalse%
\texttt{\ifinminilot} 7672 \newif\ifinminilot\inminilotfalse%

10.7 Disabling the internal commands

\texttt{\ifnum} We need also to disable some \texttt{minitoc} commands, with \texttt{\relax} (macros with no argument) or
\texttt{\let} (macros with two arguments):
\texttt{\gobbletwo} 7673 \let\tf@mtc\count@
\texttt{\mtc@string} 7674 \let\mtc@string\relax
\texttt{\appendixmtc} 7675 \let\appendixmtc\relax
\texttt{\xchapter} 7676 \let\xchapter\@gobbletwo
\texttt{\pchapter} 7677 \let\pchapter\relax
\texttt{\xsect} 7678 \let\xsect\relax
\texttt{\pxsect} 7679 \let\pxsect\relax
\texttt{\xpart} 7680 \let\xpart\@gobbletwo
\texttt{\pxpart} 7681 \let\pxpart\relax
\texttt{\xsect} 7682 \let\xsect\relax
\texttt{\pxsect} 7683 \let\pxsect\relax
Disabling the font commands

We disable the \texttt{minitoc} font commands (like \texttt{mtcSSfont}) with \texttt{\empty}, because some users might have used:

\texttt{\renewcommand{\mtcSSfont}{...}}

which will not work if we use \texttt{\relax} here.

Fonts for part level mini-tables:

\begin{verbatim}
\setcounter{equation}{0}
\let\ptcfont\empty
\let\ptcCfont\empty
\let\ptcSfont\empty
\let\ptcSSfont\empty
\let\ptcSSSfont\empty
\let\ptcPfont\empty
\let\ptcSPfont\empty
\let\plffont\empty
\let\plfSfont\empty
\let\pltfont\empty
\let\pltSfont\empty
\let\ptifont\empty
\end{verbatim}

Fonts for chapter level mini-tables:

\begin{verbatim}
\setcounter{section}{0}
\let\mtcfont\empty
\let\mtcSfont\empty
\let\mtcSSfont\empty
\let\mtcSSSfont\empty
\let\mtcPfont\empty
\let\mtcSPfont\empty
\end{verbatim}

Fonts for section level mini-tables:

\begin{verbatim}
\setcounter{subsection}{0}
\let\stcfont\empty
\let\stcSSfont\empty
\let\stcSSSfont\empty
\let\stcPfont\empty
\let\stcSPfont\empty
\end{verbatim}
10.9 Disabling the \mtcset... commands

These commands use two or three mandatory arguments:
\let\mtcsetdepth\@gobbletwo
\let\mtcsetoffset\@gobbletwo
\let\mtcsetfont\empty
\let\mtcsettitlefont\@gobbletwo
\let\mtcsettitle\@gobbletwo
\let\mtcsetformat\empty
\let\mtcsetfeature\empty
\let\mtcsetpagenumbers\@gobbletwo
\let\mtcsetrules\@gobbletwo

10.10 Disabling the \mtcpoly... commands

We simulate these commands by keeping only the fourth argument; they must still be robust.
\DeclareRobustCommand\mtcpolymtoc
\mtcpolymlof
\mtcpolymlot

10.11 Disabling the new \l@... commands

The minitoc package defines the \l@starXXX commands to format TOC entries for starred sectionning commands. We reset to the unstarred version, when necessary:
\l@starpart
\l@starchapter
\l@starsubsection
\l@starsubsubsection
\l@starparagraph
\l@starsubparagraph
10.12 Ignore the obsolete commands

\@gobble  
\firstpartis  
\firstchapteris  
\firstsectionis

We just ignore the obsolete commands (with one mandatory argument):
\let\firstpartis\@gobble \let\firstchapteris\@gobble \let\firstsectionis\@gobble

10.13 Disabling the \mtcselectlanguage and \mtcloadmlo commands

\@gobble  
\mtcselectlanguage  
\mtcloadmlo

These commands have one mandatory argument:
\let\mtcselectlanguage\@gobble \let\mtcloadmlo\@gobble

10.14 Disabling the commands for the horizontal rules

\ptcrule  
\noptcrule  
\mtcrule  
\nomtcrule  
\stcrule  
\nostcrule  
\plfrule  
\noplfrule  
\mlfrule  
\nomlfrule  
\stfrule  
\nostfrule  
\pltrule  
\nopltrule  
\mltrule  
\nomltrule  
\sttrule  
\nosttrule  

These commands have no argument:
\let\ptcrule\relax \let\noptcrule\relax \let\mtcrule\relax \let\nomtcrule\relax \let\stcrule\relax \let\nostcrule\relax \let\plfrule\relax \let\noplfrule\relax \let\mlfrule\relax \let\nomlfrule\relax \let\stfrule\relax \let\nostfrule\relax \let\pltrule\relax \let\nopltrule\relax \let\mltrule\relax \let\nomltrule\relax \let\sttrule\relax \let\nosttrule\relax
10.15 Disabling the commands for the page numbers

These commands have no argument:

\ptcpagenumbers \noptcpagenumbers \mtpagenumbers \nomtpagenumbers \stcpagenumbers \nostcpagenumbers \plfpagenumbers \noplfpagenumbers \mlfpagenumbers \nomlfpagenumbers \slfpagenumbers \noslfpagenumbers \pltpagenumbers \nopltpagenumbers \mltpagenumbers \nomltpagenumbers \sltangenbers \nosltangenbers

10.16 Disabling the mini-table features commands

We disable the commands for features (like \beforeparttoc) with \empty, because some users may have used:

\renewcommand{\beforeparttoc}{...}

which will not work if we use \relax here. These commands have no argument,

\beforeparttoc \beforepartlof \beforepartlot \afterparttoc \afterpartlof \afterpartlot \openparttoc \openpartlof \openpartlot \closeparttoc \closepartlof \closepartlot \thispageparttocstyle \thispagepartlofstyle \thispagepartlotstyle
10.17 Disabling miscellaneous flags and commands

There are some flags and commands that it is wise to declare:

```latex
\if@mtc@longext@
  \iftightmtc
    \ifktightmtc
      \l@listof
      \ifmtcsecondpart
        \chapter
    \fi
  \fi
\fi
```
10.18 Caution for some commands

Some minitoc commands should eventually be replaced if you decide to *definitely* stop using the minitoc package with your document. So we declare a flag and an \AtEndDocument block to signal that you have used these commands:

\newif\ifmtcoffwarn 
\AtEndDocument{\ifmtcoffwarn 
\mtcPackageWarningNoLine{mtcoff}{You should scan (backwards) your .log file to find some commands needing to be replaced if you decide to DEFINITELY stop using minitoc for this document. It is more wise to keep the string\usepackage\space lines for minitoc and mtcoff and to comment out only one of them}} 
\fi}

Then these commands are disabled and they set the flag and give a warning (useful to get the line number):
10.19 Disabling commands for “coffee”

\addcoffeeine \coffeeine \@gobble \@Undottedtoclinep 7876 \def\addcoffeeine#1#2#3{\relax}
7877 \let\coffeeine@gobble
7878 \let\l@coffee\relax
7879 \def\@Undottedtoclinep#1#2#3#4#5{\relax}
7880 \def\@Undottedtoclinep#1#2#3#4#5{\relax}

10.20 Disabling the mtchideinmain... environments

These environments accept one optional argument:
7881 \newenvironment{mtchideinmaintoc}[1][-1]{\empty}{\empty}
7882 \newenvironment{mtchideinmainlof}[1][-1]{\empty}{\empty}
7883 \newenvironment{mtchideinmainlot}[1][-1]{\empty}{\empty}
10.21 Inhibition of the \texttt{mtc@[save|restore]XXXdepth} internal commands

We must inhibit these commands, inserted in the .toc, .lof and .lot files by the hiding commands. So we will not have to delete these files when switching from the minitoc package to the mtcff package.

\begin{verbatim}
\let\mtc@savetocdepth\empty
\let\mtc@savelofdepth\empty
\let\mtc@savelotdepth\empty
\let\mtc@restoretocdepth\empty
\let\mtc@restorelofdepth\empty
\let\mtc@restorelotdepth\empty
\end{verbatim}

10.22 Disabling the \texttt{mtcfixglossary} command

\texttt{\textbackslash mtcfixglossary} This command accepts one optional argument:

\begin{verbatim}
\newcommand{\mtcfixglossary}[1][]{\relax}
\end{verbatim}

10.23 Disabling the \texttt{mtcfixindex} command

\texttt{\textbackslash mtcfixindex} This command accepts one optional argument:

\begin{verbatim}
\newcommand{\mtcfixindex}[1][]{\relax}
\end{verbatim}

10.24 Disabling the \texttt{mtcfixnomenclature} command

\texttt{\textbackslash mtcfixnomenclature} This command accepts one optional argument:

\begin{verbatim}
\newcommand{\mtcfixnomenclature}[1][]{\relax}
\end{verbatim}
10.25 Disabling the \addstarred... commands

These commands should be replaced by standard commands, but \mtcoff simulates and gives a warning, which will be reminded at the end of document:

\begin{verbatim}
7893 \def\addstarredpart#1{\mtcoffwarn@true
7894 \mtcPackageWarning[F0002]{mtcoff}%
7895 {\protect\addstarredpart{...} should be replaced by
7896 \MessageBreak
7897 \protect\addcontentsline{toc}{part}{...}
7898 \MessageBreak}
7899 \addcontentsline{toc}{part}{#1}}
7900 \def\addstarredchapter#1{\mtcoffwarn@true
7901 \mtcPackageWarning[F0001]{mtcoff}%
7902 {\protect\addstarredchapter{...} should be replaced by
7903 \MessageBreak
7904 \protect\addcontentsline{toc}{chapter}{...}
7905 \MessageBreak}
7906 \addcontentsline{toc}{chapter}{#1}}
7907 \def\addstarredsection#1{\mtcoffwarn@true
7908 \mtcPackageWarning[F0003]{mtcoff}%
7909 {\protect\addstarredsection{...} should be replaced by
7910 \MessageBreak
7911 \protect\addcontentsline{toc}{section}{...}
7912 \MessageBreak}
7913 \addcontentsline{toc}{section}{#1}}
\end{verbatim}

And the \mtcoff package is terminated.

7914 ⟨/mtcoff⟩
Chapter 11

Commented code of the \texttt{mtcmess} package

\begin{verbatim}
\mtcPackageInfo \PackageInfo \MessageBreak \mtcPackageWarning \PackageWarning
\mtcPackageWarningNoLine \PackageWarningNoLine \mtcPackageError \PackageError
\end{verbatim}

To make easier the search of a message in the documentation\footnote{For instance, using the search facility of some PDF reader utility.}, we will assign an unique identifier to each message of the \texttt{minitoc} and \texttt{mtcoff} packages. As the standard commands for such messages do not include this feature, we make extended versions, with the same syntax, plus a first optional argument:

Hence the first line of the message will contain the package name and the unique identifier of the message.

These macros are defined in a separate package because they are used by at least two packages (\texttt{minitoc} and \texttt{mtcoff}) and because they could be useful for other packages.
Chapter 12

Patch for the memoir class

This code must be loaded to fix an incompatibility of the minitoc package with some recent versions of the memoir class. This correction is no more necessary after the 2005/09/25 version of memoir.
Chapter 13

Language definition (.mld) and object (.mlo) files

Contents

13.1 Overview .................................................. 471
13.2 “Acadian” language: acadian.mld .................. 472
13.3 “Acadien” language: acadien.mld ............... 472
13.4 “Afrikaan” language: afrikaan.mld .............. 473
13.5 “Afrikaans” language: afrikaans.mld ......... 473
13.6 “Albanian” language: albanian.mld ............ 473
13.7 “American” language: american.mld ........... 474
13.8 “Arab” language: arab.mld ......................... 474
13.9 “Arab2” language: arab2.mld ..................... 475
13.10 “Arabi” language: arabi.mld ....................... 475
13.11 “Arabic” language: arabic.mld ................. 476
13.12 “Armenian” language: armenian.mld ........ 476
13.13 “Australian” language: australian.mld ....... 476
13.15 “Bahasa” language: bahasa.mld .............. 477
13.16 “Bahasai” language: bahasai.mld ............ 477
13.17 “Bahasam” language: bahasam.mld .......... 478
13.18 “Bangla” language: bangla.mld ............... 478
13.19 “Basque” language: basque.mld ............... 479
13.20 “Bengali” language: bengali.mld ............. 479
13.21 “Bicig” language: bicig.mld .................... 479
13.22 “Bicig2” language: bicig2.mld ............... 480
13.23 “Bicig3” language: bicig3.mld ............... 480
13.24 “Bithe” language: bithe.mld ................... 481
13.25 “Brazil” language: brazil.mld ................. 481
13.26 “Brazilian” language: brazilian.mld ....... 482
13.27 “Breton” language: breton.mld ............... 482
13.28 “British” language: british.mld .............. 483
13.29 “Bulgarian” language: bulgarian.mld ....... 483
13.30 “Bulgarianb” language: bulgarianb.mld .... 483
<table>
<thead>
<tr>
<th>Section</th>
<th>Language</th>
<th>File Type</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>13.31</td>
<td>Buryat</td>
<td>.mld</td>
<td>484</td>
</tr>
<tr>
<td>13.32</td>
<td>Buryat2</td>
<td>.mld</td>
<td>485</td>
</tr>
<tr>
<td>13.33</td>
<td>Canadian</td>
<td>.mld</td>
<td>485</td>
</tr>
<tr>
<td>13.34</td>
<td>Canadien</td>
<td>.mld</td>
<td>485</td>
</tr>
<tr>
<td>13.35</td>
<td>Castillan</td>
<td>.mld</td>
<td>486</td>
</tr>
<tr>
<td>13.36</td>
<td>Castillian</td>
<td>.mld</td>
<td>486</td>
</tr>
<tr>
<td>13.37</td>
<td>Catalan</td>
<td>.mld</td>
<td>486</td>
</tr>
<tr>
<td>13.38</td>
<td>Chinese1</td>
<td>.mld</td>
<td>487</td>
</tr>
<tr>
<td>13.39</td>
<td>Chinese2</td>
<td>.mld</td>
<td>487</td>
</tr>
<tr>
<td>13.40</td>
<td>Croatian</td>
<td>.mld</td>
<td>487</td>
</tr>
<tr>
<td>13.41</td>
<td>Croatian</td>
<td>.mld</td>
<td>487</td>
</tr>
<tr>
<td>13.42</td>
<td>Czech</td>
<td>.mld</td>
<td>488</td>
</tr>
<tr>
<td>13.43</td>
<td>Danish</td>
<td>.mld</td>
<td>488</td>
</tr>
<tr>
<td>13.44</td>
<td>Devanagari</td>
<td>.mld</td>
<td>489</td>
</tr>
<tr>
<td>13.45</td>
<td>Dutch</td>
<td>.mld</td>
<td>489</td>
</tr>
<tr>
<td>13.46</td>
<td>English1</td>
<td>.mld</td>
<td>490</td>
</tr>
<tr>
<td>13.47</td>
<td>English2</td>
<td>.mld</td>
<td>490</td>
</tr>
<tr>
<td>13.48</td>
<td>Esperanto</td>
<td>.mld</td>
<td>492</td>
</tr>
<tr>
<td>13.49</td>
<td>Esperanto</td>
<td>.mld</td>
<td>492</td>
</tr>
<tr>
<td>13.50</td>
<td>Estonian</td>
<td>.mld</td>
<td>492</td>
</tr>
<tr>
<td>13.51</td>
<td>Ethiopian</td>
<td>.mld</td>
<td>493</td>
</tr>
<tr>
<td>13.52</td>
<td>Ethiopian</td>
<td>.mld</td>
<td>493</td>
</tr>
<tr>
<td>13.53</td>
<td>Ethiopian2</td>
<td>.mld</td>
<td>494</td>
</tr>
<tr>
<td>13.54</td>
<td>Farsi1</td>
<td>.ml[d,o]</td>
<td>494</td>
</tr>
<tr>
<td>13.55</td>
<td>Farsi2</td>
<td>.ml[d,o]</td>
<td>495</td>
</tr>
<tr>
<td>13.56</td>
<td>Farsi3</td>
<td>.ml</td>
<td>495</td>
</tr>
<tr>
<td>13.57</td>
<td>Finnish</td>
<td>.ml</td>
<td>495</td>
</tr>
<tr>
<td>13.58</td>
<td>Finnish2</td>
<td>.ml</td>
<td>496</td>
</tr>
<tr>
<td>13.59</td>
<td>Francais</td>
<td>.ml</td>
<td>496</td>
</tr>
<tr>
<td>13.60</td>
<td>French</td>
<td>.ml</td>
<td>497</td>
</tr>
<tr>
<td>13.61</td>
<td>French1</td>
<td>.ml</td>
<td>497</td>
</tr>
<tr>
<td>13.62</td>
<td>French2</td>
<td>.ml</td>
<td>498</td>
</tr>
<tr>
<td>13.63</td>
<td>Frenchb</td>
<td>.ml</td>
<td>498</td>
</tr>
<tr>
<td>13.64</td>
<td>Frenchle</td>
<td>.ml</td>
<td>499</td>
</tr>
<tr>
<td>13.65</td>
<td>Frenchpro</td>
<td>.ml</td>
<td>499</td>
</tr>
<tr>
<td>13.66</td>
<td>Galician</td>
<td>.ml</td>
<td>499</td>
</tr>
<tr>
<td>13.67</td>
<td>German</td>
<td>.ml</td>
<td>500</td>
</tr>
<tr>
<td>13.68</td>
<td>Germanb</td>
<td>.ml</td>
<td>500</td>
</tr>
<tr>
<td>13.69</td>
<td>Germanb2</td>
<td>.ml</td>
<td>501</td>
</tr>
<tr>
<td>13.70</td>
<td>Greek</td>
<td>.ml</td>
<td>501</td>
</tr>
<tr>
<td>13.71</td>
<td>Greek-mono</td>
<td>.ml</td>
<td>502</td>
</tr>
<tr>
<td>13.72</td>
<td>Greek-polydemo</td>
<td>.ml</td>
<td>502</td>
</tr>
<tr>
<td>13.73</td>
<td>Greek-polykatha</td>
<td>.ml</td>
<td>503</td>
</tr>
<tr>
<td>13.74</td>
<td>Guarani</td>
<td>.ml</td>
<td>504</td>
</tr>
<tr>
<td>13.75</td>
<td>Hangul1</td>
<td>.ml[d,o]</td>
<td>505</td>
</tr>
<tr>
<td>13.76</td>
<td>Hangul2</td>
<td>.ml[d,o]</td>
<td>505</td>
</tr>
<tr>
<td>13.77</td>
<td>Hangul3</td>
<td>.ml[d,o]</td>
<td>506</td>
</tr>
<tr>
<td>13.78</td>
<td>Hangul4</td>
<td>.ml[d,o]</td>
<td>506</td>
</tr>
<tr>
<td>13.79</td>
<td>Hangul-u8</td>
<td>.ml[d,o]</td>
<td>507</td>
</tr>
<tr>
<td>13.80</td>
<td>Hanja1</td>
<td>.ml[d,o]</td>
<td>507</td>
</tr>
<tr>
<td>13.81</td>
<td>Hanja2</td>
<td>.ml[d,o]</td>
<td>508</td>
</tr>
<tr>
<td>13.82</td>
<td>Hanja-u8</td>
<td>.ml[d,o]</td>
<td>508</td>
</tr>
<tr>
<td>13.83</td>
<td>Hebrew</td>
<td>.ml</td>
<td>508</td>
</tr>
</tbody>
</table>
13.84 “Hebrew” language: hebrew2.mld ........................................ 509
13.85 “Hindi” language: hindi.mld ........................................... 510
13.86 “Hindi-modern” language: hindi-modern.mld ......................... 510
13.87 “Hungarian” language: hungarian.mld ................................ 510
13.88 “Icelandic” language: icelandic.mld .................................... 511
13.89 “Indon” language: indon.mld ........................................... 511
13.90 “Indonesian” language: indonesian.mld ................................. 511
13.91 “Interlingua” language: interlingua.mld ................................ 512
13.92 “Irish” language: irish.mld ............................................. 512
13.93 “Italian” language: italian.mld .......................................... 513
13.94 “Italian2” language: italian2.mld ....................................... 513
13.95 “Japanese” language: japanese.ml[d|o] .................................. 514
13.96 “Japanese2” language: japanese2.ml[d|o] ............................... 514
13.101 “Kannada” language: kannada.mld .................................... 517
13.102 “Khalkha” language: khalkha.mld ..................................... 517
13.103 “Latin” language: latin.mld ........................................... 517
13.104 “Latin2” language: latin2.mld ....................................... 518
13.105 “Latinc” language: latinc.mld ......................................... 518
13.106 “Latinc2” language: latinc2.mld ...................................... 519
13.107 “Latvian” language: latvian.mld ...................................... 519
13.108 “Latvian2” language: latvian2.mld ................................... 520
13.109 “Letton” language: letton.mld ....................................... 520
13.110 “Letton2” language: letton2.mld ..................................... 521
13.111 “Lithuanian” language: lithuanian.mld ............................... 521
13.112 “Lithuanian2” language: lithuanian2.mld ......................... 521
13.113 “Lowersorbian” language: lowersorbian.mld ....................... 522
13.114 “Lsorbian” language: lsorbian.mld .................................. 522
13.115 “Magyar” language: magyar.mld ..................................... 523
13.116 “Magyar2” language: magyar2.mld .................................. 523
13.117 “Magyar3” language: magyar3.mld .................................. 524
13.118 “Malay” language: malay.mld ........................................ 524
13.119 “Malayalam-b” language: malayalam-b.mld ......................... 524
13.120 “Malayalam-keli” language: malayalam-keli.mld .................. 525
13.121 “Malayalam-keli2” language: malayalam-keli2.mld ............... 525
13.122 “Malayalam-mr” language: malayalam-mr.mld ..................... 526
13.124 “Malayalam-rachana” language: malayalam-rachana.mld ....... 527
13.125 “Malayalam-rachana2” language: malayalam-rachana2.mld .... 527
13.126 “Malayalam-rachana3” language: malayalam-rachana3.mld ..... 528
13.127 “Manju” language: manju.mld ......................................... 528
13.128 “Mexican” language: mexican.mld .................................... 528
13.129 “Meyulu” language: meyulu.mld ...................................... 529
13.130 “Mongol” language: mongol.mld ...................................... 529
13.131 “Mongolb” language: mongolb.mld ................................... 530
13.132 “Mongolian” language: mongolian.mld ............................... 531
13.133 “Naustrian” language: naustrian.mld ................................ 531
13.134 “Newzealand” language: newzealand.mld ......................... 531
13.135 “Ngerman” language: ngerman.mld .................................. 531
13.136 “Ngermanb” language: ngermanb.mld ............................... 532
<table>
<thead>
<tr>
<th>Language</th>
<th>File Name</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Germanb2</td>
<td>ngermanb2.mld</td>
<td>532</td>
</tr>
<tr>
<td>Norsk</td>
<td>norsk.mld</td>
<td>533</td>
</tr>
<tr>
<td>Nynorsk</td>
<td>nynorsk.mld</td>
<td>534</td>
</tr>
<tr>
<td>Nynorsk2</td>
<td>nynorsk2.mld</td>
<td>534</td>
</tr>
<tr>
<td>Occitan</td>
<td>occitan.mld</td>
<td>535</td>
</tr>
<tr>
<td>Occitan2</td>
<td>occitan2.mld</td>
<td>535</td>
</tr>
<tr>
<td>Polish</td>
<td>polish.mld</td>
<td>536</td>
</tr>
<tr>
<td>Polish2</td>
<td>polish2.mld</td>
<td>536</td>
</tr>
<tr>
<td>Polski</td>
<td>polski.mld</td>
<td>537</td>
</tr>
<tr>
<td>Portugese</td>
<td>portuguese.mld</td>
<td>537</td>
</tr>
<tr>
<td>Romanian</td>
<td>romanian.mld</td>
<td>538</td>
</tr>
<tr>
<td>Romanian2</td>
<td>romanian2.mld</td>
<td>538</td>
</tr>
<tr>
<td>Romanian3</td>
<td>romanian3.mld</td>
<td>539</td>
</tr>
<tr>
<td>Russian</td>
<td>russian.mld</td>
<td>539</td>
</tr>
<tr>
<td>Russian2m</td>
<td>russian2m.mld</td>
<td>540</td>
</tr>
<tr>
<td>Russian2o</td>
<td>russian2o.mld</td>
<td>541</td>
</tr>
<tr>
<td>Russianb</td>
<td>russianb.mld</td>
<td>541</td>
</tr>
<tr>
<td>Russiannc</td>
<td>russiannc.mld</td>
<td>542</td>
</tr>
<tr>
<td>Russian-cca</td>
<td>russian-cca.mld[d</td>
<td>o]</td>
</tr>
<tr>
<td>Russian-ccal</td>
<td>russian-ccal.mld[d</td>
<td>o]</td>
</tr>
<tr>
<td>Russian-lh</td>
<td>russian-lh.mld[d</td>
<td>o]</td>
</tr>
<tr>
<td>Russian-lhcyral</td>
<td>russian-lhcyral.mld</td>
<td>544</td>
</tr>
<tr>
<td>Russian-lhcyrkoi</td>
<td>russian-lhcyrkoi.mld</td>
<td>544</td>
</tr>
<tr>
<td>Russian-lhcyrwin</td>
<td>russian-lhcyrwin.mld</td>
<td>545</td>
</tr>
<tr>
<td>Samin</td>
<td>samin.mld</td>
<td>545</td>
</tr>
<tr>
<td>Scottish</td>
<td>scottish.mld</td>
<td>546</td>
</tr>
<tr>
<td>Serbian</td>
<td>serbian.mld</td>
<td>546</td>
</tr>
<tr>
<td>Serbianb</td>
<td>serbianc.mld</td>
<td>547</td>
</tr>
<tr>
<td>Slovak</td>
<td>slovak.mld</td>
<td>547</td>
</tr>
<tr>
<td>Slovene</td>
<td>slovene.mld</td>
<td>548</td>
</tr>
<tr>
<td>Spanish</td>
<td>spanish.mld</td>
<td>548</td>
</tr>
<tr>
<td>Spanish2</td>
<td>spanish2.mld</td>
<td>549</td>
</tr>
<tr>
<td>Spanish3</td>
<td>spanish3.mld</td>
<td>549</td>
</tr>
<tr>
<td>Spanish4</td>
<td>spanish4.mld</td>
<td>550</td>
</tr>
<tr>
<td>Swahili</td>
<td>swahili.mld</td>
<td>550</td>
</tr>
<tr>
<td>Swedish</td>
<td>swedish.mld</td>
<td>551</td>
</tr>
<tr>
<td>Swedish2</td>
<td>swedish2.mld</td>
<td>551</td>
</tr>
<tr>
<td>Thai</td>
<td>thai.ml[d</td>
<td>o]</td>
</tr>
<tr>
<td>Turkish</td>
<td>turkish.mld</td>
<td>552</td>
</tr>
<tr>
<td>Uighur</td>
<td>uighur.mld</td>
<td>552</td>
</tr>
<tr>
<td>Uighur2</td>
<td>uighur2.mld</td>
<td>553</td>
</tr>
<tr>
<td>Uighur3</td>
<td>uighur3.mld</td>
<td>553</td>
</tr>
<tr>
<td>UkEnglish</td>
<td>UKenglish.mld</td>
<td>553</td>
</tr>
<tr>
<td>Ukraineb</td>
<td>ukraineb.mld</td>
<td>553</td>
</tr>
<tr>
<td>Ukraineb</td>
<td>ukraineb.mld</td>
<td>554</td>
</tr>
<tr>
<td>Uppersorbian</td>
<td>uppersorbian.mld</td>
<td>554</td>
</tr>
<tr>
<td>USenglish</td>
<td>USenglish.mld</td>
<td>555</td>
</tr>
<tr>
<td>Uxorbian</td>
<td>usorbian.mld</td>
<td>555</td>
</tr>
<tr>
<td>Vietnam</td>
<td>vietnam.mld</td>
<td>555</td>
</tr>
<tr>
<td>Vietnamese</td>
<td>vietnamese.mld</td>
<td>556</td>
</tr>
<tr>
<td>Welsh</td>
<td>welsh.mld</td>
<td>556</td>
</tr>
</tbody>
</table>
13.1 Overview

This chapter shows the code of each .mld file. A .mld file is a minitoc language definition file, which defines the titles of the mini-tables for a given language. It contains often some comments about its origin, if you need further details.

For some languages, I have added a map (and a flag) of the country or area where the language is spoken, if it is not trivial. The origin of each map is given by an URL to the graphic file or to the WEB page where I found it. Note that the [294] and [229] Web sites are useful sources. Maps from [229] are under the Creative Commons License, see http://creativecommons.org/licenses/by-nc-sa/1.0/deed.en_GB. The site http://www.expatries.senat.fr/pays.html allows to look at the maps of many countries (but not of France!). The Perry-Castañeda Library Map Collection [395] (The University of Texas at Austin, http://www.lib.utexas.edu/maps) contains countless maps.

Many free maps were also found by a search in the vast Wikipedia (i.e. http://en.wikipedia.org, http://fr.wikipedia.org, http://de.wikipedia.org, http://simple.wikipedia.org, etc.). If you are curious and brave, you can also find many maps and documents about Eastern Europa and about Asia at http://www.hunmagyar.org; that site give many historical informations.

A .mld file is loaded either via a package option in the \usepackage command for the minitoc package (or a global option for the document), either via the command:

\mtcselectlanguage{(language)}

\ptctitle \plftitle \pltitle
\mtctitle \mlftitle \mlttitle
\stctitle \slftitle \slttitle

Each .mld file must define the nine following commands (for the mini-tables of contents, mini-lists of figures and mini-lists of tables, at the part, chapter and section levels):
Many .mld files require special fonts adequate for the corresponding language; as this is a language-dependent issue, the user must set up the correct language and font context for each language, like using the babel package [54, 60, 61, 74], the CJK system [127, 297, 298], the HrTEX system [266, in korean], the Antomega system [272], the ArabTeX [276, 277], BangTeX [362], Devanagari for TeX [364], ethiopTeX [162], 1, Guarani [45], Malayalam [4], mormal [5], MonTeX [137, 140], or ArmTeX [142] packages. Note that it is often the english name of the language which is used to name the corresponding .mld file.

But for some oriental languages 2, the source of the titles use some exotic encodings, difficult to manipulate in a .dtx file, the .mld file is then just a wrapper loading a .mlo file 3, not generated by the .dtx files in the current version of minitoc package. To go around this limitation, the minitoc.ins file uses filecontents environments to generate the .mlo files. The adequate input encoding must be set up by the user before loading the .mld file via the \mtcselectlanguage command.

Since version 49, the minitoc package checks the presence of the language .mld file (and of the language .mlo file if necessary) for each language option of the package, before validating the option. If a .mld or .mlo file is missing, the corresponding language option is not enabled and a warning message is written in the document.log file. But the presence of the english.mld file is mandatory, because english is the default language. If some .mld or .mlo files are missing, the list of this files is given in the .log file. You should find these files on CTAN.

13.2 “Acadian” language: acadian.mld

The acadian language 4 is just french, so we load the french.mld file (see section 13.60 on page 497):

\mtcselectlanguage {acadian}
\ProvidesFile{acadian.mld}[2004/12/14]\mtcselectlanguage{french}\mtcselectlanguage{/acadian}

13.3 “Acadien” language: acadien.mld

The “acadien” language 4 is just french (“acadien” is the french term for “acadian”), so we load the french.mld file (see section 13.60 on page 497):

---

1 By Mohammad Ghaosi (ghodsi@rose.ipm.ac.ir) and the FarsiTeX Project Group. See the FarsiTeX site at http://www.farsitex.org
2 Mainly for chinese, farsi (iranian), hangûl (korean), hanja (korean), japanese, malayalam-omega, thai, and russian variants.
3 The extension .mlo means minitoc language object.
4 Spoken in Acadia and some parts of the south of the USA, like Louisiana.
13.4 “Afrikaan” language: afrikaan.mld

The titles for the “afrikaan” language\(^5\) come from the dutch.dtx file (by Johannes L. Braams and Stoffel Lombard) in the babel package [55, 60, 61]:

\begin{verbatim}
\def\ptctitle{Inhoudsopgawe}%
\def\plftitle{Lys van figure}%
\def\plttitle{Lys van tabelle}%
\def\mtctitle{Inhoudsopgawe}%
\def\mlftitle{Lys van figure}%
\def\mlttitle{Lys van tabelle}%
\end{verbatim}

13.5 “Afrikaans” language: afrikaans.mld

The term “afrikaans” is a synonym of “afrikaan”, so we just load afrikaan.mld (see section 13.4):

\begin{verbatim}
\def\ptaftitle{Inhoudsopgawe}%
\def\plftitle{Lys van figure}%
\def\plttitle{Lys van tabelle}%
\def\mtctitle{Inhoudsopgawe}%
\def\mlftitle{Lys van figure}%
\def\mlttitle{Lys van tabelle}%
\end{verbatim}

13.6 “Albanian” language: albanian.mld

The albanian language (shqip) is spoken in Albania and some regions of Macedonia, Montenegro, Serbia and Kosovo. The titles for the “albanian” language are taken from the albanian.dtx file (with a contribution of Adi Zaimi) in the babel package [60, 61, 101]:

\begin{verbatim}
\def\ptaftitle{Inhoudsopgawe}%
\def\plftitle{Lys van figure}%
\def\plttitle{Lys van tabelle}%
\end{verbatim}

\(^5\) Spoken in South Africa and Namibia, it has dutch origins; compare with section 13.44 on page 489. See also http://www.tlfq.ulaval.ca/axl/afrique/afrikaans.htm in [294].
13.7 “American” language: american.mld

The “american” language is just like “english” (the languages themselves have some differences, like the hyphenation rules, some spellings and phonetics), so we just load english.mld (see section 13.45 on page 490):

\mtcselectlanguage{american}
\ProvidesFile{american.mld}[2004/12/14]

13.8 “Arab” language: arab.mld

The titles for the “arab” language (al-‘Arabiyyah) are taken from the ArabTeX package [276, 277] (by Klaus LAGALLY), which should be used, with the associated fonts. The arabic language is spoken in: Algeria, Bahrain, Egypt, Iraq, Jordan, Kuwait, Lebanon, Libya, Mauritania, Morocco, Oman, Palestinian territories, Qatar, Saudi Arabia, Sudan, Syria, Tunisia, United Arab Emirates, Western Sahara, Yemen by a majority; it is also the liturgical language of Islam.

\mtcselectlanguage{arab}
\ProvidesFile{arab.mld}[1999/03/16]
13.9 “Arab2” language: arab2.mld

The titles for the “arab2” language are taken from the ArabTEX package [276, 277] (by Klaus Lagally), which should be used, with the associated fonts. It is a variant of the “arab” language.

13.10 “Arabi” language: arabi.mld

The titles for the “arabi” language are taken from the Arabi package [243] (by Youssef Jabri), which should be used, with the associated fonts.
13.11 “Arabic” language: arabic.mld

The “arabic” language is a synonym for “arab”, so we just load arab.mld (see section 13.8 on page 474):

\mtcselectlanguage{arabic}
\ProvidesFile{arabic.mld}[2005/02/10]\mtcselectlanguage{arab}%
\mtcselectlanguage{arabic}

13.12 “Armenian” language: armenian.mld

The titles for the “armenian” language (hayeren) are taken from the ArmTeX package [142] (by Serguei DACHIAN, Arnak DALALYAN and Vartan AKOPIAN), which should be used, with the associated fonts. The armenian language is spoken in Armenia, in a part of Azerbaidjan and in the armenian diaspora.

\mtcselectlanguage{armenian}
\ProvidesFile{armenian.mld}[1999/06/28]%
\ProvidesFile{armenian.mld}[1999/06/28]%
\ProvidesFile{armenian.mld}[1999/06/28]%
\def\ptctitle{Bovandakuthyun}%
\def\plftitle{Patkerneri cank}%
\def\plttitle{Aghyusakneri cank}%
\def\mtctitle{Bovandakuthyun}%
\def\mlftitle{Patkerneri cank}%
\def\mlttitle{Aghyusakneri cank}%
\def\stctitle{Bovandakuthyun}%
\def\slftitle{Patkerneri cank}%
\def\slttitle{Aghyusakneri cank}%
\mtcselectlanguage{armenian}

13.13 “Australian” language: australian.mld

The “australian” language is just like “english”, so we just load english.mld (see section 13.45 on page 490):

\mtcselectlanguage{australian}
\ProvidesFile{australian.mld}[2006/01/11]\mtcselectlanguage{english}%
\mtcselectlanguage{australian}
13.14 “Austrian” language: austrian.mld

For the mini-table titles, the “austrian” language is like the “german” language, so we load german.mld (see section 13.67 on page 500):

\mtcselectlanguage{german}

13.15 “Bahasa” language: bahasa.mld

The “bahasa” language is just like “bahasai”, so we just load bahasai.mld (see section 13.16):

\mtcselectlanguage{bahasai}

13.16 “Bahasai” language: bahasai.mld

The titles of the mini-tables for the “bahasai” language (bahasa indonesia / bahasa meyalu) are taken from the file bahasa.dtx (by Jörg Knappen and Terry Mart) in the babel package [60, 61, 82]. Specific fonts are needed. See also section 13.17 on the following page. The word “bahasa” means “language” in bahasa. For other names for this language, see sections 13.15 and 13.89 to 13.90 on page 511.
13.17 “Bahamas” language: bahasam.mld

The titles of the mini-tables for the “bahasam” language (Bahasa Malaysia) \(^7\) are taken from the file bahasam.dtx (by Jörg KNAPPEN, Terry MART and Bob MARGOLIS) in the babel package [60, 61, 83]. Specific fonts are needed. See also section 13.15 on the page before. For other names for this language, see sections 13.118 on page 524 and 13.129 on page 529.

\begin{verbatim}
\ProvidesFile{bahasam.mld}[2006/12/19]%
%% Bahasa Malaysia titles from bahasam.dtx in the babel package
%% Knappen, Jörg & Mart, Terry & Margolis, Bob
\def\ptctitle{Kandungan}\
\def\plftitle{Senarai Gambar}\
\def\plttitle{Senarai Jadual}\
\def\mtctitle{Kandungan}\
\def\mlftitle{Senarai Gambar}\
\def\mlttitle{Senarai Jadual}\
\def\stctitle{Kandungan}\
\def\slftitle{Senarai Gambar}\
\def\slttitle{Senarai Jadual}\
\end{verbatim}

13.18 “Bangla” language: bangla.mld

The titles for the “bangla” (bengali) language \(^8\) are taken from the BangTeX package [362] (by Palash Baran Pal); they need specific fonts (the bengali alphabet is derived from sanskrit).

\begin{verbatim}
\ProvidesFile{bangla.mld}[2006/03/31]%
%% Bangla titles from BangTeX. Needs specific fonts.
\def\ptctitle{suu\*c*ipotRo}% <-----
\def\plftitle{cho\*b*ir ta\*l*ika}%
\def\plttitle{cho\*k*er ta\*l*ika}%
\def\mtctitle{suu\*c*i}%
\def\mlftitle{cho\*b*ir ta\*l*ika}%
\def\mlttitle{cho\*k*er ta\*l*ika}%
\def\stctitle{suu\*c*i}%
\def\slftitle{cho\*b*ir ta\*l*ika}%
\def\slttitle{cho\*k*er ta\*l*ika}%
\end{verbatim}

---

\(^7\) Spoken in Indonesia and Malaysia, with different pronunciations and titles but the same writing. Bahasam is the Malaysian variant.

\(^8\) Spoken in Bangladesh and some parts of India, like Occidental Bengal (19), Orissa (21), Assam (18), Bihar (10) and Tripura (16).
13.19 “Basque” language: basque.mld

The titles for the “basque” language\(^9\) (euskara) are taken from the basque.dtx file in the babel package [60–62], by Juan M. Aguirregabiria and Julio Sánchez, with help from Zunbeltz Izaola Azkona. It seems that 8 bits fonts are preferable.

\[
\begin{align*}
\text{\def\ptctitle{Gaien Aurkibidea}}
\text{\def\plftitle{Irudien Zerrenda}}
\text{\def\plttitle{Taulen Zerrenda}}
\text{\def\mtctitle{Gaien Aurkibidea}}
\text{\def\mlftitle{Irudien Zerrenda}}
\text{\def\mlttitle{Taulen Zerrenda}}
\text{\def\stctitle{Gaien Aurkibidea}}
\text{\def\slftitle{Irudien Zerrenda}}
\text{\def\slttitle{Taulen Zerrenda}}
\end{align*}
\]

13.20 “Bengali” language: bengali.mld

The “bengali” language is a synonym for the “bangla” language, so we load the file bangla.mld (see section 13.18 on the preceding page):

\[
\begin{align*}
\text{\mtcselectlanguage{bangla}}
\end{align*}
\]

13.21 “Bicig” language: bicig.mld

The titles for the “bicig” language\(^{10}\) are taken from the MonTeX package [137, 140]. This language requires specific fonts. See also sections 13.22 to 13.23 on the following page, and 13.130 on page 529.

---

9 Spoken in the basque country, in the north of Spain and south-west of France.

10 The bicig is a written form of the mongolian language. It is also known as Uighur or Bichig. See also section 13.178 on page 552.
13.22 “Bicig2” language: bicig2.mld

The titles for the “bicig2” language\textsuperscript{11} are taken from the \textsc{MonTeX} package \cite{137, 140}. This language requires specific fonts. See also sections 13.21 on the page before, and 13.23, and 13.130 on page 529.

\textsuperscript{11}The \texttt{bicig}, or uighur, is a written form of the mongolian language, \texttt{bicig2} is a variant. See also section 13.179 on page 553.

13.23 “Bicig3” language: bicig3.mld

The titles for the “bicig3” language\textsuperscript{12} are taken from the \textsc{MonTeX} package \cite{137, 140}. This language requires specific fonts. See also sections 13.21 to 13.22 on pages 479–480 and 13.130 on page 529.

\textsuperscript{12}The \texttt{bicig}, or uighur, is a written form of the mongolian language, \texttt{bicig3} is a variant. See also section 13.180 on page 553.
13.24 “Bithe” language: bithe.mld

The titles for the “bithe” language are taken from the MonTEX package [137, 140]. This language requires specific fonts. See also sections 13.127 on page 528 and 13.130 on page 529. The Manju writing, or bithe system is a close relative of the Mongolian system; the basical letter shapes are the same. Yet for Manju, a set of diacritics (dots and circles) was designed to eliminate all the ambiguities of Mongolian.

13.25 “Brazil” language: brazil.mld

The titles for the “brazil” language (português brasileiro or português do Brasil) are taken from the portuges.dtx file (for portuguese titles by Jose Pedro RAMALHETE) in the babel package [60, 61, 92]:

The bithe is a written form of the manju variant of the mongolian language.

It is the main portuguese dialect spoken in Brazil. Note that these titles are different in Brazil and in Portugal. Arnaldo Viegas de Lima contributed to brazilian translations. See section 13.148 on page 537.
13.26 “Brazilian” language: brazilian.mld

\mtcselectlanguage The “brazilian” language is just like “brazil”, so we just load brazil.mld (see section 13.25 on the page before):

\mtcselectlanguage{brazil}

13.27 “Breton” language: breton.mld


\mtcselectlanguage{breton}
13.28 “British” language: british.mld

The “british” language is just like “english”, so we just load english.mld (see section 13.45 on page 490):

\mtcselectlanguage{english}

13.29 “Bulgarian” language: bulgarian.mld

The titles for the “bulgarian” language (български) are taken from the bulgarian.dtx file in the babel package [60, 61, 67]; they require specific cyrillic fonts. See also section 13.30.

13.30 “Bulgarianb” language: bulgarianb.mld

The titles for the “bulgarianb” (upper bulgarian) language are taken from the russianb.dtx file (by Olga G. Lapko, Vladimir Volovich, Werner Lemberg, and Irina A. Makhovaya) of the
The babel package [60, 61, 84, 286]; they require specific cyrillic fonts. See also section 13.29 on the preceding page.

The titles for the "buryat" language are taken from the MonTeX package [137, 140]. This language requires specific fonts. See also section 13.130 on page 529.

\[\text{Spoken in some regions of Mongolia and in the Buryatia republic, near Lake Baikal.}\]
13.32 “Buryat2” language: buryat2.mld

\mnr The titles for the “buryat2” language (a variant for the “buryat” language, see section 13.31 on the preceding page) are taken from the MonTeX package [137, 140]. This language requires specific fonts. See also section 13.130 on page 529.

\ProvidesFile{buryat2.mld}[1999/03/16] % Buryat2 titles. Needs special fonts.
\def\ptctitle{{\mnr Aguulga}} %
\def\plftitle{{\mnr Zuraga"i jagsaalt}} %
\def\plttitle{{\mnr X"usn"ag"at"a"i jagsaalt}} %
\def\mtctitle{{\mnr Aguulga}} %
\def\mlftitle{{\mnr Zuraga"i jagsaalt}} %
\def\mlttitle{{\mnr X"usn"ag"at"a"i jagsaalt}} %
\def\stctitle{{\mnr Aguulga}} %
\def\slftitle{{\mnr Zuraga"i jagsaalt}} %
\def\slttitle{{\mnr X"usn"ag"at"a"i jagsaalt}} %
\endinput

13.33 “Canadian” language: canadian.mld

\mtcselectlanguage The “canadian” language  (note the final “ian”) is just the english language spoken in Canada. We just load the file english.mld (see section 13.45 on page 490):

\ProvidesFile{canadian.mld}[2004/12/14] %
\mtcselectlanguage{english}
\endinput

13.34 “Canadien” language: canadien.mld

\mtcselectlanguage The “canadien” language  (note the final “ien”) is just the french language spoken in Canada. We just load the file french.mld (see section 13.60 on page 497):

\ProvidesFile{canadien.mld}[2004/12/14] %
\mtcselectlanguage{french}
\endinput
13.35 “Castillan” language: castillan.mld

The “castillan” language is better known as “spanish”, but is spoken mainly in Castile, a part of central Spain. We just load the spanish.mld file (see section 13.169 on page 548):

\mtcselectlanguage{spanish}

13.36 “Castillian” language: castillian.mld

“Castillian” is just the english name for “castillan”, so we just load the spanish.mld file (see section 13.169 on page 548):

\mtcselectlanguage{spanish}

13.37 “Catalan” language: catalan.mld

The titles for the “catalan” language (català, valencià) are taken from the catalan.dtx file (adapted from spanish by Gonçal Badenes and Jörg Knappen) in the babel package [60, 61, 64]:

\mtcselectlanguage{catalan}

---

17 Spoken in Catalunya, the eastern part of Spain, around Barcelona, and in Roussillon, in France.
13.38 “Chinese1” language: chinese1.ml[d|o]

There are several variants for the Chinese language. The “chinese1” language uses titles taken from the Bg5.cap file in the CJK system [127, 297, 298] (by Werner Lemberg). Special fonts are needed, of course. See also section 13.39. See [418] about the history of China and the Chinese language. The titles for the “chinese1” language contain characters that cannot be easily generated, hence we load chinese1.mlo.

13.39 “Chinese2” language: chinese2.ml[d|o]

The “chinese2” language uses titles taken from the Bg5.cpx file in the CJK system [127, 297, 298] (by Werner Lemberg). Special fonts are needed, of course. See also section 13.38. The titles for the “chinese2” language contain characters that cannot be easily generated, hence we load chinese2.mlo.

13.40 “Croatian” language: croatian.mld

The titles for the “croatian” language (hrvatski) are taken from the file croatian.dtx file (by Alan Pac) in the babel package [60, 61, 89]:

[13] — Language definition (.mld) and object (.mlo) files 487
13.41 “Czech” language: czech.mld

The titles for the “czech” language (čeština, český jazyk) are taken from the file czech.dtx (contributions by Miloš V. LOKAIČEK) in the babel package [60, 61, 87]:

```
\ProvidesFile{czech.mld}[2007/12/04]
%% Czech titles from czech.dtx (babel). Lokaj\'i\v{c}ek, Milo\'v{s} V.
\def\ptctitle{Obsah}\%
\def\plftitle{Seznam obr\'\accent'97\(z\)aj}\%
\def\plttitle{Seznam tabulek}\%
\def\mtctitle{Obsah}\%
\def\mlftitle{Seznam obr\'\accent'97\(z\)aj}\%
\def\mlttitle{Seznam tabulek}\%
\def\stctitle{Obsah}\%
\def\slftitle{Seznam obr\'\accent'97\(z\)aj}\%
\def\slttitle{Seznam tabulek}\%
```

13.42 “Danish” language: danish.mld

The titles for the “danish” language are taken from the danish.dtx file (by Henning LARSEN) in the babel package [60, 61, 85]:

```
\ProvidesFile{danish.mld}[2007/12/18]
%% Danish titles from danish.dtx (babel). Larsen, Henning (larsen@cernvm.cern.ch)
\def\ptctitle{Indhold}\%
\def\plftitle{Figurer}\%
\def\plttitle{Tabeller}\%
\def\mtctitle{Indhold}\%
\def\mlftitle{Figurer}\%
\def\mlttitle{Tabeller}\%
\def\stctitle{Indhold}\%
\def\slftitle{Figurer}\%
\def\slttitle{Tabeller}\%
```

The danish (dansk) language is spoken in Denmark, in the Faeroe Islands and in Greenland.
13.43 “Devanagari” language: devanagari.mld

The titles for the “devanagari” language are taken from the devanagari.sty and captions.dn files (by Anshuman Pandey, C. V. Radhakrishnan, Zdeněk Wagner, John Smith, Kevin Carmody, Richard Mahoney and Dominik Wujastyk) in the Devanāgari package [364] (Devanāgari). See also section 13.85 on page 510.

Specific fonts are required. The home page of the package is http://devnag.sarovar.org. See also [148] about the hindi language.

13.44 “Dutch” language: dutch.mld

The titles for the “dutch” language are taken from the dutch.dtx file (by Johannes L. Braams) in the babel package [55, 60, 61]:

The dutch language (nederlands) is spoken in the Netherlands and a part of Belgium.
13.45 “English” language: english.mld

The titles for the “english” language are taken from the english.dtx file (by Johannes L. Braams) in the babel package [56, 60, 61]. The presence of the english.mld file is mandatory, because english is the default language. See also sections 13.7 on page 474, 13.13 on page 476, 13.28 on page 483, 13.33 on page 485, 13.134 on page 531, 13.181 on page 553, and 13.185 on page 555.

13.46 “English1” language: english1.mld

\ifnum\value{part}=1\relax
\else Table of Contents of Part-\Roman{part}\fi

13.47 “English2” language: english2.mld

The titles for the “english2” language are again taken from the english.dtx file (written by Johannes L. Braams) in the babel package [56, 60, 61], with adaptations at the part level.
13.48 “Esperant” language: esperant.mld

The titles for the “esperant” (espéranto) language are taken from the esperanto.dtx file (by Marti Ruiz-Altaba and Jörg Knappen) in the babel package [60, 61, 94]. The esperanto artificial language was created in the 1877–1885 years by Doctor Ludwig Lejzer Zamenhof (1859–1917) of Warsaw, Poland.

13.49 “Esperanto” language: esperanto.mld

The “esperanto” and “esperant” languages are synonyms, so we just load the esperant.mld file (see section 13.48):

13.50 “Estonian” language: estonian.mld

The titles for the “estonian” language are taken from the estonian.dtx file (by Enn Saar) in the babel package [60, 61, 95]:

---


21 Estonian (eesti keel) is not a baltic language, but a language from the uralian family.
13.51 “Ethiopia” language: ethiopia.mld

The titles for the “ethiopia” language (amharic, እማር𝗶 Vaults) are taken from the ethiop package [44] (written by Berhanu BEYENE, Manfred KUDELEK, Olaf KUMMER, and Jochen MEITZINGER). Specific fonts are needed. See also section 13.53 on the next page.

13.52 “Ethiopian” language: ethiopian.mld

The “ethiopian” language is just a synonym for the “ethiopia” language, so we just load the ethiopia.mld file (see section 13.51).
13.53 “Ethiopian2” language: ethiopian2.mld

The titles for the “ethiopian2” language (for Omega) are taken from the ethiop package [44] (by Berhanu Beyene, Manfred Kudlek, Olaf Kummer, and Jochen Metzinger). Specific fonts are needed. See also section 13.51 on the page before.

\begin{verbatim}
\ProvidesFile{ethiopian2.mld}[2006/01/30]
\def\ptctitle{^^^^12ed^^^^12d8^^^^1275}\
\def\plftitle{^^^^12e8^^^^1225^^^^12d5^^^^120e^^^^127d ^^^^121b^^^^12cd^^^^132b}\
\def\plttitle{^^^^12e8^^^^1230^^^^1295^^^^1320^^^^1228^^^^12e5
\def\mtctitle{^^^^12ed^^^^12d8^^^^1275}\
\def\mlftitle{^^^^12e8^^^^1225^^^^12d5^^^^120e^^^^127d ^^^^121b^^^^12cd^^^^132b}\
\def\mlttitle{^^^^12e8^^^^1230^^^^1295^^^^1320^^^^1228^^^^12e5
\def\stctitle{^^^^12ed^^^^12d8^^^^1275}\
\def\slftitle{^^^^12e8^^^^1225^^^^12d5^^^^120e^^^^127d ^^^^121b^^^^12cd^^^^132b}\
\def\slttitle{^^^^12e8^^^^1230^^^^1295^^^^1320^^^^1228^^^^12e5}
\end{verbatim}

13.54 “Farsi1” language: farsi1.ml[d|o]

\begin{verbatim}
\mtcloadmlo
\ProvidesFile{farsi1.mld}[2005/09/13]\mtcloadmlo{farsi1}
\mtcloadmlo{farsi1}
\end{verbatim}

There are several variants for the farsi language, spoken in Iran and Afghanistan. The “farsi1” language uses titles taken from the \texttt{farsi.sty} file in the \texttt{FarsiTEX} [162] system, by Mohammad Ghodsi, Behdad Esfahbod, Roozbeh Pournader, Hassan Abolhassani, and others. Special fonts are needed, of course. See also section 13.55 on the following page. The titles for the “farsi1” language contain characters that cannot be easily generated, hence we load \texttt{farsi1.mlo}.

\begin{verbatim}
\mtcloadmlo
\ProvidesFile{farsi1.mld}[2005/09/13]\mtcloadmlo{farsi1}
\mtcloadmlo{farsi1}
\end{verbatim}

\[22\text{By Mohammad Ghodsi (ghodsi@rose.ipm.ac.ir) and the FarsiTEX Project Group. See the FarsiTEX site at http://www.farsitex.org}\]
There are several variants for the farsi language, spoken in Iran and Afghanistan. The “farsi2” language uses titles taken from the farsi.sty file in the FarsiTEX system [162] by Mohammad Ghodsi, Roozbeh Pournader, Behdad Esfahbod, Hassan Abolhassani, and others. Special fonts are needed, of course. See also section 13.54 on the page before.

\mtcloadmlo The titles for the “farsi2” language contain characters that cannot be easily generated, hence we load farsi2.mlo.

\ProvidesFile{farsi2.mld}[2005/09/13]\mtcloadmlo{farsi2} % From farsi.sty (FarsiTeX project: http://www.farsitex.org). Dr Mohammad Ghodsi, Roozbeh Pournader (roozbeh@sharif.edu), Hassan Abolhassani, & others.

\FR There are several variants for the farsi language, spoken in Iran and Afghanistan. The “farsi3” language uses titles taken from the farsi.ldf file in the Arabi system[243], by Youssef Jabri. Special fonts are needed, of course.

\ProvidesFile{farsi3.mld}[2006/07/27] % From farsi.ldf of the Arabi system by Youssef Jabri.

\FR The titles for the “farsi3” language contain characters that cannot be easily generated, hence we load farsi3.mlo.

\ProvidesFile{farsi3.mld}[2006/07/27] % From farsi.ldf of the Arabi system by Youssef Jabri.

\FR The titles for the “farsi3” language are taken from the farsi.dtx file (by Mikko Kanerva and Keranen Reino) in the babel package [60, 61, 80]. See also section 13.58 on the following page.

By Mohammad Ghodsi (ghodsi@rose.ipm.ac.ir) and the FarsiTEX Project Group. See the FarsiTEX site at http://www.farsitex.org
13.58  “Finnish2” language: finnish2.mld

The titles for the “finnish2” language are taken from a variant proposed by the finnish.dtx file (by Mikko Kanerva and Keranen Reino) in the babel package [60, 61, 80]. See also section 13.57 on the page before.

13.59  “Francais” language: francais.mld

The “francais” (français) language is a synonym for the “french” language, so we load the file french.mld (see section 13.60 on the following page):
13.60 “French” language: french.mld

The titles for the “french” language are taken from the frenchb.dtx file (by Daniel Flipo) in the babel package [60, 61, 75]. See also sections 13.2 to 13.3 on page 472, 13.34 on page 485, 13.59 on the page before, and 13.63 to 13.65 on pages 498–499.

13.61 “French1” language: french1.mld

The titles for the “french1” language are taken from the frenchb.dtx (by Daniel Flipo) file in the babel package [60, 61, 75], with some adaptations for the part-level titles.
The titles for the "french2" language are taken from the frenchb.dtx file (by Daniel Flipo) in the babel package [60, 61, 75], with some adaptations for the part-level titles. See also section 9.5.8 on page 273, for the subtle distinction between “deuxième” and “seconde”. See the mtc-2nd.tex example file in section 4.2 on page 92.

This is an example of a .mld file needing some support from code in the minitoc package.

The "frenchb" language is a synonym for the "french" language, so we load the french.mld file. See section 13.60 on the preceding page.
13.64 “Frenchle” language: frenchle.mld

\mtcselectlanguage

The “frenchle” language is a synonym for the “french” language, so we load the french.mld file. See section 13.60 on page 497. See also [179].

13.65 “Frenchpro” language: frenchpro.mld

\mtcselectlanguage

The “frenchpro” language is a synonym for the “french” language, so we load the french.mld file. See section 13.60 on page 497. See also [180, 181].

13.66 “Galician” language: galician.mld

The titles for the “galician” language (galego) are taken from the galician.dtx file, (by Manuel Carriba and Javier A. Mu´gica de Rivera) derived from the spanish.dtx file (by Javier Bezos) in the babel package [60, 61, 70, 71]:

Spoken in Galice, in the north-west part of Spain, around Santiago de Compostela.
13.67 “German” language: german.mld

The titles for the “german” language (deutsch) are taken from the babel package [60, 61]. See also the section 13.14 on page 477.

\ProvidesFile{german.mld}[1999/03/16]%
\% German titles
\def\ptctitle{Inhaltsangabe}%
\def\plftitle{Figuren}%
\def\plttitle{Tabellen}%
\def\mtctitle{Inhaltsangabe}%
\def\mlftitle{Figuren}%
\def\mlttitle{Tabellen}%
\def\stctitle{Inhaltsangabe}%
\def\slftitle{Figuren}%
\def\slttitle{Tabellen}%

13.68 “Germanb” language: germanb.mld

The “germanb” language is a variant for the “german” language. The titles come from germanb.dtx (by Johannes L. Braams and Bernd Raichle) in the babel package [60, 61, 90]:

\ProvidesFile{germanb.mld}[2006/01/13]%
\% German titles (variant) from germanb.dtx (babel). Braams, Johannes-L. & Raichle, Bernd
\def\ptctitle{Inhaltsverzeichnis}%
\def\plftitle{Abbildungsverzeichnis}%
\def\plttitle{Tabellenverzeichnis}%
\def\mtctitle{Inhaltsverzeichnis}%
\def\mlftitle{Abbildungsverzeichnis}%
\def\mlttitle{Tabellenverzeichnis}%
\def\stctitle{Inhalt}%
\def\slftitle{Abbildungen}%
\def\slttitle{Tabellen}%
13.69 “Germanb2” language: germanb2.mld

The “germanb2” language is a variant for the “german” language, with short titles. See also section 13.68 on the preceding page. The titles are taken from the file germanb.dtx (by Johannes L. Braams and Bernd Raichle) in the babel package [60, 61]:

8715 ⟨∗germanb2⟩
8716 \ProvidesFile{germanb2.mld}[2007/12/18]%
8717 \def\ptctitle{Inhalt}%
8718 \def\plftitle{Abbildungen}%
8719 \def\plttitle{Tabellen}%
8720 \def\mtctitle{Inhalt}%
8721 \def\mlftitle{Abbildungen}%
8722 \def\mlttitle{Tabellen}%
8723 \def\stctitle{Inhalt}%
8724 \def\slftitle{Abbildungen}%
8725 \def\slttitle{Tabellen}%
8726 ⟨/germanb2⟩

13.70 “Greek” language: greek.mld

The titles for the “greek” language (modern greek, νέα αληθινά) are taken from the greek.dtx file (by Apostolos Syropoulos) in the babel package [60, 61, 98, 427]. Greek fonts are required.

8728 ⟨∗greek⟩
8729 \ProvidesFile{greek.mld}[2007/12/18]%
8730 \def\ptctitle{Perieq'omena}%
8731 \def\plftitle{Kat'alogos Sqhm'atwn}%
8732 \def\plttitle{Kat'alogos Pin'akwn}%
8733 \def\mtctitle{Perieq'omena}%
8734 \def\mlftitle{Kat'alogos Sqhm'atwn}%
8735 \def\mlttitle{Kat'alogos Pin'akwn}%
8736 \def\stctitle{Perieq'omena}%
8737 \def\slftitle{Kat'alogos Sqhm'atwn}%
8738 \def\slttitle{Kat'alogos Pin'akwn}%
8739 ⟨/greek⟩
13.71 “Greek-mono” language: greek-mono.mld

The titles for the “greek-mono” language are taken from the omega-greek.ldf file (by Alexej M. Kryukov and Dmitry Ivanov) in the Antomega project [272]:

```latex
\localgreek
\\ProvidesFile{greek-mono.mld}[2005/02/08]%
%% from omega-greek.ldf (Antomega project). Needs Omega.
%% Alexej M. Kryukov & Dmitry Ivanov
\def\ptctitle{\localgreek
^{\hspace{0.2cm}03a0\hspace{0.2cm}03b5\hspace{0.2cm}03c1\hspace{0.2cm}03b9\hspace{0.2cm}03b5\hspace{0.2cm}03c7\hspace{0.2cm}03cc\hspace{0.2cm}03bc\hspace{0.2cm}03b5\hspace{0.2cm}03bd\hspace{0.2cm}03b1}}%
\def\plftitle{\localgreek
^{\hspace{0.2cm}039a\hspace{0.2cm}03b1\hspace{0.2cm}03c4\hspace{0.2cm}03ac\hspace{0.2cm}03bb\hspace{0.2cm}03bf\hspace{0.2cm}03b3\hspace{0.2cm}03bf\hspace{0.2cm}03c2\hspace{0.2cm}03c3\hspace{0.2cm}03c7\hspace{0.2cm}03bc\hspace{0.2cm}03ac\hspace{0.2cm}03c4\hspace{0.2cm}03c9}}%
\def\mtctitle{\localgreek
^{\hspace{0.2cm}03a0\hspace{0.2cm}03b5\hspace{0.2cm}03c1\hspace{0.2cm}03b9\hspace{0.2cm}03b5\hspace{0.2cm}03c7\hspace{0.2cm}03cc\hspace{0.2cm}03bc\hspace{0.2cm}03b5\hspace{0.2cm}03bd\hspace{0.2cm}03b1}}%
\def\mlftitle{\localgreek
^{\hspace{0.2cm}039a\hspace{0.2cm}03b1\hspace{0.2cm}03c4\hspace{0.2cm}03ac\hspace{0.2cm}03bb\hspace{0.2cm}03bf\hspace{0.2cm}03b3\hspace{0.2cm}03bf\hspace{0.2cm}03c2\hspace{0.2cm}03c3\hspace{0.2cm}03c7\hspace{0.2cm}03bc\hspace{0.2cm}03ac\hspace{0.2cm}03c4\hspace{0.2cm}03c9}}%
\def\stctitle{\localgreek
^{\hspace{0.2cm}03a0\hspace{0.2cm}03b5\hspace{0.2cm}03c1\hspace{0.2cm}03b9\hspace{0.2cm}03b5\hspace{0.2cm}03c7\hspace{0.2cm}03cc\hspace{0.2cm}03bc\hspace{0.2cm}03b5\hspace{0.2cm}03bd\hspace{0.2cm}03b1}}%
\def\slftitle{\localgreek
^{\hspace{0.2cm}039a\hspace{0.2cm}03b1\hspace{0.2cm}03c4\hspace{0.2cm}03ac\hspace{0.2cm}03bb\hspace{0.2cm}03bf\hspace{0.2cm}03b3\hspace{0.2cm}03bf\hspace{0.2cm}03c2\hspace{0.2cm}03c3\hspace{0.2cm}03c7\hspace{0.2cm}03bc\hspace{0.2cm}03ac\hspace{0.2cm}03c4\hspace{0.2cm}03c9}}%
\end{verbatim}
```

13.72 “Greek-polydemo” language: greek-polydemo.mld

The titles for the “greek-polydemo” language are taken from the file omega-greek.ldf (by Alexej M. Kryukov and Dmitry Ivanov) in the Antomega project [272]:

```latex
\localgreek
```

\footnote{Monotonic greek, from a recent (1982) but strongly contested – and contestable – reform of the greek language.}

\footnote{Polytonic demotic (popular) greek, for classical greek.}
13.73  “Greek-polykatha” language:  
greek-polykatha.mld

The titles for the “greek-polykatha” language are taken from the omega-greek.ldf file (by Alexej M. Kryukov and Dmitry Ivanov) in the Antomega project [272]:

28 Polytonic greek, « kathaverousa » (purified) style, a form of the Greek language created during the early xix-th century by Adamantios Korais, to purify the language from the Byzantine and non-greek vocabulary. It has now been obsoleted by the demotic (popular) greek, but it has left a very noticeable trace in the modern Greek language.
13.74 “Guarani” language: guarani.mld

The “guarani” (guaraní) language is the main language spoken in Paraguay. Very often, a mixture of Guarani and Spanish, known as Jopará or Yopará, is spoken. The titles are taken from the guarani.ldf file by Javier Bezos [45]. A special input encoding (win-gn.def) is needed. These files are available on the CTAN archives.
The Korean language was originally written using the Chinese characters; it is now mainly written in Hangûl, the Korean writing system, optionally incorporating Hanja to write Sino-Korean words [453]. See [214, page 150], [216] and [365].

The titles for the “hangul1” language (korean in hangûl script, first variant) are taken from the file hangul1.cap of the CJK system [127, 297, 298] (by Werner Lemberg). Special fonts are needed, of course.

See also sections 13.76 to 13.82 on pages 505–508.

\mtcloadmlo The titles for the “hangul1” language contain characters that cannot be easily generated, hence we load hangul1.mlo.

The titles for the “hangul2” language (korean in hangûl script, second variant) are taken from the file hangul2.cpx of the CJK system [127, 297, 298] (by Werner Lemberg). Special fonts are needed, of course.

See also sections 13.75 and 13.77 to 13.82 on pages 506–508.

\mtcloadmlo The titles for the “hangul2” language contain characters that cannot be easily generated, hence we load hangul2.mlo.
13.77  “Hangul3” language: hangul3.ml[d|o]

The titles for the “hangul3” language (korean in hangul script, third variant) are taken from the file hangul2.cap of the CJK system [127, 297, 298] (by Werner Lemberg). Special fonts are needed, of course. See also sections 13.75 to 13.76 on the preceding page and 13.78 to 13.82 on pages 506–508.

\mtcloadmlo  The titles for the “hangul3” language contain characters that cannot be easily generated, hence we load hangul3.mlo.

13.78  “Hangul4” language: hangul4.ml[d|o]

The titles for the “hangul4” language (korean in hangul script, fourth variant) are taken from the file hangul2.cpx of the CJK system [127, 297, 298] (by Werner Lemberg). Special fonts are needed, of course. See also sections 13.75 to 13.77 on pages 505–506, and 13.79 to 13.82 on pages 507–508.

\mtcloadmlo  The titles for the “hangul4” language contain characters that cannot be easily generated, hence we load hangul4.mlo.
13.79  “Hangul-u8” language: hangul-u8.ml[d|o]

The titles for the “hangul-u8” language (korean in hangûl script, for Lambda Λ) are taken from the file u8hangul.tex of the HLaTeX system [266, in korean] by Un Koaunghi. Special fonts are needed, of course. Input encoding is UTF-8.

See also sections 13.75 to 13.78 on pages 505–506, and 13.80 to 13.82 on pages 507–508. See [214, page 150], [216] and [365].

\mtcloadmlo The titles for the “hangul-u8” language contain characters that cannot be easily generated, hence we load hangul-u8.mlo.

13.80  “Hanja1” language: hanja1.mld.ml[d|o]

The titles for the “hanja1” language (korean in the old script hanja, first variant) are taken from the file hanja.cpx of the CJK system [127, 297, 298] (by Werner Lemberg). Special fonts are needed, of course.

See also sections 13.75 to 13.79 on pages 505–507, and 13.81 to 13.82 on the next page.

\mtcloadmlo The titles for the “hanja1” language contain characters that cannot be easily generated, hence we load hanja1.mlo.
13.81 “Hanja2” language: hanja2.mld[dl]

The titles for the “hanja2” language (Korean in the old script hanja, second variant) are taken from the file hanja.cap of the CJK system [127, 297, 298] (by Werner LEMBERG). Special fonts are needed, of course. See also sections 13.75 to 13.80 on pages 505–507, and 13.82.

\mtcloadmlo The titles for the “hanja2” language contain characters that cannot be easily generated, hence we load hanja2.mlo.

13.82 “Hanja-u8” language: hanja-u8.mld[dl]

The titles for the “hanja-u8” language (korean in hanja script, for Lambda Λ) are taken from the file u8hanja.tex of the HLaTeX system [266, in korean] by Un KOAUNGHI. Special fonts are needed, of course. Input encoding is UTF-8. See also sections 13.75 to 13.81 on pages 505–508. See [214, page 150], [216] and [365].

\mtcloadmlo The titles for the “hanja-u8” language contain characters that cannot be easily generated, hence we load hanja-u8.mlo.

13.83 “Hebrew” language: hebrew.mld

The titles for the “hebrew” language (ivrit) are taken from the ArabTeX package [276, 277] (by Klaus Lagally), with the associated fonts. See also section 13.84 on the next page. See the hebrew alphabet (alefbet): http://www.jewfaq.org/graphics/hebrew.gif.
13.84 “Hebrew2” language: hebrew2.mld

The titles for the “hebrew2” language are taken from the file hebrew.dtx (by Boris Lavva and Rama Porrat) in the babel package [60, 61, 86], which should be used, with the associated fonts and encodings. See also section 13.83 on the preceding page.
13.85 “Hindi” language: hindi.mld

The “hindi” language is just like “devanagari”, so we just load devanagari.mld (see section 13.43 on page 489):

\mtcselectlanguage{devanagari}

The titles for the “hindi-modern” language are taken from the captions.dn file (by Anshuman Pandey, C.V. Radhakrishnan, Zdeněk Wagner, John Smith, Kevin Carmody, Richard Mahoney and Dominik Wujastyk) in the Devanāgarī package [364] (Devanāgarī) after conversion. See also section 13.43 on page 489. Specific fonts are required. The home page of the package is http://devnag.sarovar.org.

The “hungarian” language is a synonym of the “magyar” language, so we load magyar.mld. See section 13.115 on page 523.
13.88 “Icelandic” language: icelandic.mld

The titles for the “icelandic” language (íslenska) are taken from the icelandic.dtx file (by Einar Árnason) in the babel package [60, 61, 63]. See also [236].

13.89 “Indon” language: indon.mld

The “indon” language is just like “bahasai”, so we just load bahasai.mld (see section 13.16 on page 477):

13.90 “Indonesian” language: indonesian.mld

The “indonesian” language is just like “bahasai”, so we just load bahasai.mld (see section 13.16 on page 477):
13.91 “Interlingua” language: interlingua.mld

The titles for the “interlingua” language are taken from the interlingua.dtx file (by Peter Kleiweg) in the babel package [60, 61, 81]. Interlingua is an auxiliary language, built from the common vocabulary of Spanish/Portuguese, English, Italian and French, with some normalisation of spelling. The grammar is very easy, more similar to English’s than to neolatin languages. See also:

- Union Interlinguiste de France: http://www.interlingua.com.fr/
- interlingua-english dictionary: http://www.interlingua.com/ied/
- interlingua grammar (in french): http://filip.ouvaton.org/ia/gram/entra1.html
- some sites in interlingua: http://www.dmoz.org/World/Interlingua
- other sites about interlingua: http://www.cle.unicamp.br/wcp3/interlingua.htm

13.92 “Irish” language: irish.mld

The titles for the “irish” language (gaeilge) come from the irish.dtx file (by Johannes L. Braams, Marion Gunn and Fraser Grant) in the babel package [57, 60, 61]:

The site http://www.interlingua.com is mostly written in interlingua (as is http://interlingua.altervista.org), in case you want to read some sample of it.
13.93 “Italian” language: italian.mld

The titles for the “italian” language (italiano) come from the file italian.dtx (by Maurizio Codogno and Claudio Beccari) in the babel package [60, 61, 73]. See also section 13.94.

13.94 “Italian2” language: italian2.mld

The titles for the “italian2” language are the same as for the “italian” language, except at the part level (“Contenuto”). See also section 13.93.
13.95 “Japanese” language: `japanese.ml[d|o]`

There are several variants for the Japanese titles. The titles for a first variant of the “Japanese” language have been found (by a Google search) on the Web site of Professor Toshiki KUMAZAWA. But see also other variants in sections 13.96 to 13.100 on pages 514–516.

The titles for the “Japanese” language contain characters that cannot be easily generated, hence we load `japanese.mlo`.

13.96 “Japanese2” language: `japanese2.ml[d|o]`

The titles for the “Japanese2” language (Japanese, second variant) are taken from file JIS.cap of the CJK system [127, 297, 298] (by Werner LEMBERG). Special fonts are needed, of course. See also sections 13.95, and 13.97 to 13.100 on pages 515–516.

The titles for the “Japanese2” language contain characters that cannot be easily generated, hence we load `japanese2.mlo`.

---

30http://www.biwako.shiga-u.ac.jp/sensei/kumazawa/tex/minitoc.html
13.97 “Japanese3” language: *japanese3.ml*[d|o]

The titles for the “japanese3” language (japanese, third variant) are taken from file JIS.cpx of the CJK system [127, 297, 298] (by Werner Lemberg).

Special fonts are needed, of course. See also sections 13.95 to 13.96 on the page before, and 13.98s+mld+japanese6. The titles for the “japanese3” language contain characters that cannot be easily generated, hence we load *japanese3.mlo*.

\mtcloadmlo  The titles for the “japanese3” language contain characters that cannot be easily generated, hence we load *japanese3.mlo*.

13.98 “Japanese4” language: *japanese4.ml*[d|o]

The titles for the “japanese4” language (japanese, fourth version) are taken from file SJIS.cap of the CJK system [127, 297, 298] (by Werner Lemberg). Special fonts are needed, of course. See also sections 13.95 to 13.97 on pages 514–515, and 13.99 to 13.100 on the next page.

\mtcloadmlo  The titles for the “japanese4” language contain characters that cannot be easily generated, hence we load *japanese4.mlo*.
13.99 “Japanese5” language: \texttt{japanese5.mlo}

The titles for the “japanese5” (Japanese, fifth variant) language are taken from file \texttt{SJIS.cpx} of the CJK system [127, 297, 298] (by Werner Lemberg). Special fonts are needed, of course. See also sections 13.95 to 13.98 on pages 514–515, and 13.100. The titles for the “japanese5” language contain characters that cannot be easily generated, hence we load \texttt{japanese5.mlo}.

\texttt{mtcloadmlo} The titles for the “japanese5” language contain characters that cannot be easily generated, hence we load \texttt{japanese5.mlo}.

13.100 “Japanese6” language: \texttt{japanese6.mlo}

The titles for the “japanese6” (Japanese, sixth variant) language have been found (by a Google search) on the Web site of Professor Toshiki Kumazawa\footnote{http://www.biwako.shiga-u.ac.jp/sensei/kumazawa/tex/minitoc.html}. See also sections 13.95 to 13.99 on pages 514–516.

\texttt{mtcloadmlo} The titles for the “japanese6” language contain characters that cannot be easily generated, hence we load \texttt{japanese6.mlo}.

13.101 “Kannada” language: \texttt{kannada.mld}

The Kannada (“kannada”) (or Kannara) language is a dravidian language spoken in the Karnataka state (main town: Bangalore) of India. Titles are taken in the \texttt{kanlel.sty} package file from the KannadaTeX project\footnote{http://Sarovar.org/projects/kannadatex} by C. S. Yogananda and K. K. Subramaniam. Specific fonts are required. See the alphabet here: \url{http://www.omniglot.com/writing/kannada.htm}.

\footnote{http://www.biwako.shiga-u.ac.jp/sensei/kumazawa/tex/minitoc.html}
“Khalkha” language: khalkha.mld

“khalkha” is a synonym for “xalx”, so we just load xalx.mld (see sections 13.190 to 13.192 on pages 557–558):

```
\ProvidesFile{khalkha.mld}[2005/11/16]\mtcselectlanguage{khalkha}%
```

“Latin” language: latin.mld

The titles for the “latin” language (medieval) are taken from the latin.dtx file (by Claudio Beccari, Raffaella Tabacco, and Krzysztof Konrad Zelechowski) in the babel package [60, 61, 65]. See also section 13.104 on the next page. The latin language is still used by the Catholic Church and the Vatican for archives and some texts.

```
\ProvidesFile{latin.mld}[2006/01/13]%
```
13.104 “Latin2” language: latin2.mld

The titles for the “latin2” language (Latin, medieval, abbreviated variant) are taken from the \texttt{latin.dtx} (by Claudio Beccari, Raffaella Tabacco, and Krzysztof Konrad Żelechowski) file in the \texttt{babel} package [60, 61, 65], but abbreviated. See also section 13.103 on the preceding page.

13.105 “Latinc” language: latinc.mld

The titles for the “latinc” language (classical Latin) are taken from the \texttt{latin.dtx} file (by Claudio Beccari and Krzysztof Konrad Żelechowski) in the \texttt{babel} package [60, 61, 65]. See also section 13.106 on the next page.
13.106 “Latinc2” language: latinc2.mld

The titles for the “latinc2” language (classical latin, abbreviated variant) are taken from the latin.dtx (by Claudio Beccari and Krzysztof Konrad Zelechowski) file in the babel package [60, 61, 65], but abbreviated. See also section 13.105 on the preceding page.

13.107 “Latvian” language: latvian.mld

The titles for the “latvian” language 33 (latviešu valoda) come from the latvian.ldf file (by Alexej M. Kryukov and Dmitry Ivanov) in the Antomega project [272]. See also section 13.109 on the next page.

33Note that “latvian” is the original name for “letton”. 
13.108 “Latvian2” language: latvian2.mld

The titles for the “latvian2” language come from the latvian.ldf file (by Andris Lasis and Ivars Drikis) at http://home.lanet.lv/~drikis/TeX/2e/latvian.ldf. See also section 13.107 on the preceding page.

13.109 “Letton” language: letton.mld

The “letton” language is a synonym for the “latvian” language, so we just load latvian.mld. See section 13.107 on the page before.

13.110 “Letton2” language: letton2.mld

The “letton2” language is a synonym for the “latvian2” language, so we just load latvian2.mld. See section 13.108 on the preceding page.
13.111 “Lithuanian” language: *lithuanian.mld*

The titles for the “lithuanian” language (*lietuvių kalba*) are taken from the *lithuanian.ldf* file
(by Sigitas Tolusis) for the *babel* package [60, 61]. See also section 13.112.

```
\ProvidesFile{lithuanian.mld}[2007/12/04]%
\Lithuanian titles from lithuanian.ldf
\% in http://www.vtex.lt/tex/download/zip/babel.zip
\% by Tolusis, Sigitas (sigitas@vtex.lt)
\def\ptctitle{Turinys}%
\def\plftitle{Paveiksl\k{u}s\k{a}ra\v{s}as}%
\def\plttitle{Lentel\k{u}s\k{a}ra\v{s}}%
\def\mtctitle{Turinys}%
\def\mlftitle{Paveiksl\k{u}s\k{a}ra\v{s}as}%
\def\mlttitle{Lentel\k{u}s\k{a}ra\v{s}}%
\def\stctitle{Turinys}%
\def\slftitle{Paveiksl\k{u}s\k{a}ra\v{s}as}%
\def\slttitle{Lentel\k{u}s\k{a}ra\v{s}}%
\end{lithuanian2}
```

13.112 “Lithuanian2” language: *lithuanian2.mld*

The titles for the “lithuanian2” language (variant) are taken from the *lithuanian.ldf* file, found in http://www.vtex.lt/tex/littex/littex-20070713.tar.gz, (by Sigitas Tolusis) for the *babel* package [60, 61]. See also section 13.111. The L7x encoding and the Latin Modern fonts are needed.

```
\ProvidesFile{lithuanian2.mld}[2007/12/04]%
\Lithuanian titles (variant) from lithuanian.ldf
\% in http://www.vtex.lt/tex/littex/littex-20060928.tar.gz
\% by Tolusis, Sigitas (sigitas@vtex.lt)
\def\ptctitle{Turinys}%
\def\plftitle{Iliustracij\k{u} s\k{a}ra\v{s}as}%
\def\plttitle{Lenteli\k{u} s\k{a}ra\v{s}}%
\def\mtctitle{Turinys}%
\def\mlftitle{Iliustracij\k{u} s\k{a}ra\v{s}as}%
\def\mlttitle{Lenteli\k{u} s\k{a}ra\v{s}}%
\def\stctitle{Turinys}%
\def\slftitle{Iliustracij\k{u} s\k{a}ra\v{s}as}%
\def\slttitle{Lenteli\k{u} s\k{a}ra\v{s}}%
\end{lithuanian2}
```

13.113 “Lowersorbian” language: lowersorbian.mld

The titles for the “lowersorbian” language\(^{35}\) (\textit{dolnoserbski, dolnoservšćina}) are taken from the \texttt{lsorbian.dtx} file (by Eduard Werner) in the \texttt{babel} package \cite{60, 61, 99}. See also section 13.184 on page 554. A shorter language name is \texttt{lsorbian} (see section 13.114).

\begin{verbatim}
\ProvidesFile{lowersorbian.mld}[2006/02/28]
%% Lower sorbian titles from lsorbian.dtx (babel) by Werner, Eduard
\def\ptctitle{Wop\'simje\'se}
\def\plftitle{Zapis wobrazow}
\def\plttitle{Zapis tabulkow}
\def\mtctitle{Wop\'simje\'se}
\def\mlftitle{Zapis wobrazow}
\def\mlttitle{Zapis tabulkow}
\def\stctitle{Wop\'simje\'se}
\def\slftitle{Zapis wobrazow}
\def\slttitle{Zapis tabulkow}
\end{verbatim}

13.114 “Lsorbian” language: lsorbian.mld

The “lsorbian” language is a synonym for “lowersorbian”, so we just need to load \texttt{lowersorbian.mld}. See section 13.113.

\begin{verbatim}
\ProvidesFile{lsorbian.mld}[2007/12/04]\mtcselectlanguage{lowersorbian}
\end{verbatim}

13.115 “Magyar” language: magyar.mld

The titles for the “magyar” language are taken from the \texttt{magyar.dtx} (by József Bérces and Árpád Bíró, with help from Attila Koppanyi) file in the \texttt{babel} package \cite{60, 61, 66}. A synonym of “magyar” is “hungarian” (see section 13.87 on page 510). See also sections 13.116 to 13.117 on pages 523–524 for variants.

\begin{verbatim}
\ProvidesFile{magyar.mld}[2006/03/08]\mtcselectlanguage{magyar}
\end{verbatim}

\textsuperscript{35} Lower sorbian. Sorbian, or wendisch, is a member of the west slavic subgroup of indo-european languages spoken in Lower Lusatia in the german \textit{länder} of Saxony and Brandenburg. The Sorbs are descendents of the Wends, the german name for the slavic tribes who occupied the area between the Elbe and Saale rivers in the west and the Odra (Oder) river in the east during the medieval period (vi-th century).
13.116 “Magyar2” language: magyar2.mld

The titles for the “magyar2” language are taken from a variant proposed in the magyar.dtx file of the babel package [60, 61] (by József Bérces, Árpád Bíró, and Attila Köppanyi). See also sections 13.115 and 13.117 on the following page.

13.117 “Magyar3” language: magyar3.mld

The titles for the “magyar3” language (third variant of magyar) are taken from the magyar.dtx file (by József Bérces, Árpád Bíró, and Attila Köppanyi) in the babel package [60, 61, 66]. See also sections 13.115 to 13.116 on the page before.

\ProvidesFile{magyar3.mld}[2006/03/08]
% Magyar3 titles (variant) from magyar.dtx (babel).
% Bíró, Árpád & Bérces, József

\ProvidesFile{magyar2.mld}[2008/04/03]
% Magyar2 titles (variant) from magyar.dtx (babel).
% Bíró, Árpád & Bérces, József

The situation of the magyar language in the babel package is not clear; some experimental versions exist.
13.118 “Malay” language: malay.mld

\mtcselectlanguage The “malay” language is just like “bahasam”, so we just load bahasam.mld (see section 13.17 on page 478):

9274 ⟨∗malay⟩
9275 \ProvidesFile{malay.mld}[2006/01/11]\mtcselectlanguage{bahasam}⟨malay⟩
9276 ⟨/malay⟩

13.119 “Malayalam-b” language: malayalam-b.mld

\mm The titles for the “malayalam-b” language are taken from the malayalam package [4] by A.J. Alex. The Malayalam language is spoken from the western coast of Malabar to the extreme southern India, mainly in the Kerala state. It is one of the dravidian languages strongly bound to the Tamil language. The alphabet and the script are dated from the 8th or 9th centuries. This language option requires specific fonts (depending on the option of the malayalam package). It should be used with the following options of the malayalam package: aathira, ambili, anahka, ashtamudi, aswathi, ayilyambold, bhanu, bhavana, chippi, gauri, gopika, indulekha, ISMkarthika, ISMkaumudi, ISMravivarma, jaya, kaumudi, kottakkal, makam, malavika mridula, payippad, periyar, ravivartha, sabari, sarada, sruthy, and triruvathira. See also sections 13.120 to 13.126 on pages 525–528.

9277 ⟨∗malayalam-b⟩
9278 \ProvidesFile{malayalam-b.mld}[2007/12/04]\mm
9279 \def\ptctitle{\X{<68>\X{<197>\X{<83>\X{<161>}<119>}}\X{<78>\X{<110>\X{<123>\X{<88>\X{<167>\X{<196>}}}}}}
9280 \def\plftitle{\X{<116>\X{<83>\X{<95>\X{<110>\X{<102>\X{<123>\X{<73>\X{<196>}}}}}}
9281 \def\plttitle{\X{<116>\X{<83>\X{<95>\X{<110>\X{<102>\X{<123>\X{<73>\X{<196>}}}}}}
9282 \def\mtctitle{\X{<68>\X{<197>\X{<83>\X{<161>}<119>}}\X{<78>\X{<110>\X{<123>\X{<88>\X{<167>\X{<196>}}}}}}
9283 \def\mlftitle{\X{<116>\X{<83>\X{<95>\X{<110>\X{<102>\X{<123>\X{<73>\X{<196>}}}}}}
9284 \def\mlttitle{\X{<116>\X{<83>\X{<95>\X{<110>\X{<102>\X{<123>\X{<73>\X{<196>}}}}}}
9285 \def\stctitle{\X{<68>\X{<197>\X{<83>\X{<161>}<119>}}\X{<78>\X{<110>\X{<123>\X{<88>\X{<167>\X{<196>}}}}}}
9286 \def\slttitle{\X{<116>\X{<83>\X{<95>\X{<110>\X{<102>\X{<123>\X{<73>\X{<196>}}}}}}
9287 \def\slttitle{\X{<116>\X{<83>\X{<95>\X{<110>\X{<102>\X{<123>\X{<73>\X{<196>}}}}}}

37 There is a great variety of fonts for malayalam; hence I have attempted to limit the number of .mld files.
13.120 “Malayalam-keli” language: malayalam-keli.mld

The titles for the “malayalam-keli” language, with the “Keli” fonts, are taken from the malayalam package [4] by A.J. Alex. This language requires specific fonts. See also sections 13.119 on the page before and 13.121 to 13.126 on pages 525–528.

13.121 “Malayalam-keli2” language: malayalam-keli2.mld

The titles for the “malayalam-keli2” language are taken from the malayalam package [4] by A.J. Alex. This language requires specific fonts (keli second variant). See also sections 13.119 to 13.120 on pages 524–525 and 13.122 to 13.126 on pages 526–528.
“Malayalam-mr” language: malayalam-mr.mld


“Malayalam-omega” language: malayalam-omega.mld

This is the Malayalam language implementation “malayalam-omega” based on Lambda (Λ) (the version of LATEX for Omega) via the ormal package [5] by A.J. Alex of the Malayalam-Omega project. As the titles contain characters in a special encoding, we must load a .mlo file. A lot of fonts are available via options of the ormal package. See also sections 13.119 to 13.122 on pages 524–526 and 13.124 to 13.126 on pages 527–528.

“Malayalam-rachana” language: malayalam-rachana.mld

The titles for the “malayalam-rachana” language, with the traditionnal “Rachana” fonts (old lipi), are taken from the malayalam package [4] by A.J. Alex. This language requires specific fonts. See also sections 13.119 to 13.123 on pages 524–526 and 13.125 to 13.126 on pages 527–528.
13.125 “Malayalam-rachana2” language: 
malayalam-rachana2.mld

The titles for the “malayalam-rachana2” language, with the reformed “Rachana” fonts (new lipi), are taken from the malayalam package [4] by A.J. Alex. This language requires specific fonts. See also sections 13.119 to 13.124 on pages 524–527 and 13.126 on the following page.

13.126 “Malayalam-rachana3” language: 
malayalam-rachana3.mld

The titles for the “malayalam-rachana3” language are taken from the malayalam package [4] by A.J. Alex. This language requires specific fonts (rachana). See also sections 13.119 to 13.125 on pages 524–527.
13.127 “Manju” language: manju.mld

The “manju” language is a synonym for “bithe”, so we just load bithe.mld (see section 13.24 on page 481):

\mtcselectlanguage{bithe}

13.128 “Mexican” language: mexican.mld

The titles for the “mexican” language (español mexicano) are taken from the mexican.ldf file (by Luis Rivera) in http://mirror.ctan.org/language/spanish/nonstandard/mx/. Mexican is a spanish (castillan) dialect. The title of the parttocs is shorter for articles. See also section 13.172 on page 550.
13.129 “Meyalu” language: meyalu.mld

The “meyalu” language is just like “bahasam”, so we just load bahasam.mld (see section 13.17 on page 478):

\mtcselectlanguage{meyalu}
\ProvidesFile{meyalu.mld}[2006/01/13]\mtcselectlanguage{bahasam}%
\mtcselectlanguage{meyalu}

13.130 “Mongol” language: mongol.mld

The titles for the “mongol” language are taken from the MonTEX package [137, 140] (by Oliver Corff and Dorjpalam Dorj). This language requires specific fonts. See also sections 13.21 to 13.24 on pages 479–481, 13.31 to 13.32 on pages 484–485, and 13.190 to 13.192 on pages 557–558.

\mtcselectlanguage{mongol}
\ProvidesFile{mongol.mld}[1999/03/16]%
\def\ptctitle{{\mnr Garqig}}%
\def\plftitle{{\mnr Zurgi"in jagsaalt}}%
\def\plttitle{{\mnr X"usn"agti"in jagsaalt}}%
\def\mtctitle{{\mnr Garqig}}%
\def\mlftitle{{\mnr Zurgi"in jagsaalt}}%
\def\mlttitle{{\mnr X"usn"agti"in jagsaalt}}%
\def\stctitle{{\mnr Garqig}}%
\def\slftitle{{\mnr Zurgi"in jagsaalt}}%
\def\slttitle{{\mnr X"usn"agti"in jagsaalt}}%
\mtcselectlanguage{mongol}

13.131 “Mongolb” language: mongolb.mld

This is another variant for the mongolian titles, taken from the mongolian.dtx file [26] (by Dorjgotov Batmunkh) for the babel package [60, 61] (hence the final “b” in “mongolb”).

\mtcselectlanguage{cyr}
The titles for the “mongolb” language use cyrillic characters and the X2 and T2 encodings and are derived from the russianb.dtx file (by Olga G. Lapko, Vladimir Volovich and Werner Lemberg).
This is an other name for the “mongolb” language, because the babel package \cite{60, 61} uses the name “mongolian”. We just load mongolb.mld. See section \ref{13.131} on the preceding page.
13.133 “Naustrian” language: naustrian.mld

The “naustrian” language is a synonym of the “ngermanb” language (a revised version of the germanb variant of the german language), so we just load the ngermanb.mld file. See also section 13.136 on the next page.

13.134 “Newzealand” language: newzealand.mld

The “newzealand” language is just like “english”, so we just load english.mld (section 13.45 on page 490):

13.135 “Ngerman” language: ngerman.mld

The “ngerman” language is a synonym of the “ngermanb” language\(^{39}\), so we just load the ngermanb.mld file. See also section 13.136 on the next page.

13.136 “Ngermanb” language: ngermanb.mld

The titles for the “ngermanb” language\(^{40}\) are taken from the file ngermanb.dtx file (by Bernd Raichle and Walter Schmidt) in the babel package [60, 61, 91]. See also sections 13.133 on the preceding page, and 13.135 on the page before.

\(^{39}\) A revised version of the germanb variant of the german language.

\(^{40}\) A variant of the german language, with revised spelling.
[13] — Language definition (.mld) and object (.mlo) files

13.137 “Ngermanb2” language: ngermanb2.mld

The titles for the “ngermanb2” language (revised spelling and short titles) are taken from the ngermanb.dtx file (by Bernd Raichle and Walter Schmidt) in the babel package [60, 61, 91], and abbreviated. See also section 13.136.

13.138 “Norsk” language: norsk.mld

The titles for the “norsk” language (or bokmål, “language of the kingdom”) are taken from the norsk.dtx file (by Johannes L. Braams, Håvard Helstrup, Alv Kjetil Holme, Per Steinar Iversen, Terje Engeset Petterst and Rune Kleveland) in the babel package [58, 60, 61], with help from Dag Langmyhr. See also section 13.140 on the next page.
13.139 “Norsk2” language: norsk2.mld

The titles for the “norsk2” language (or bokmål, “language of the kingdom”) are taken from the babel package [58, 60, 61], with help from Dag Langmyhr, and abbreviated.

13.140 “Nynorsk” language: nynorsk.mld

The titles for the “nynorsk” language 41 are taken from the norsk.dtx file (by Johannes L. Braams, Hävard Helstrup, Alv Kjetil Holme, Per Steinar Iversen, Terje Engeset Petterst and Rune Kleveland) in the babel package [58, 60, 61], with help from Dag Langmyhr. See also section 13.138 on the preceding page.

---

41 Created around 1800 by Ivar Åssen to make a real independent and national norvegian language, in reaction to danish, from the various dialects spoken in the country. But nynorsk has never gained much popularity outside rural regions.
13.141 “Nynorsk2” language: nynorsk2.mld

The titles for the “nynorsk2” language are variants of the titles of the “nynorsk” language. See also section 13.140.

13.142 “Occitan” language: occitan.mld

The occitan language\(^{42}\) is still spoken in the south of France, from Limoges (Lètmòges), Bordeaux (Bordèu) and Toulouse (Tołosa\(^{43}\)) to Marseille (Marseilha) and Nice (Niça), with many local variants. This bilingual street sign in Toulouse (Tołosa), like many such signs found in historical parts of the city, is maintained primarily for its antique charm; it is typical of what little remains of the “lenga d’oc” in southern French cities. See also [122].

\(^{42}\)I used the site [http://www.panoccitan.org/diccionari.aspx](http://www.panoccitan.org/diccionari.aspx) for the translations.

\(^{43}\)Per Tołosa toijorn mai!
13.143 “Occitan” language: occitan2.mld

The occitan2 language provides an example of variants for the occitan titles.

13.144 “Polish” language: polish.mld

The titles for the “polish” language (język polski) are taken from the polish.dtx file (by Elmar Schalück and Michael Janich) in the babel package [60, 61, 96]. See also sections 13.145 to 13.146 on pages 536–537.
13.145 “Polish2” language: polish2.mld

The titles for the “polish2” language are taken from the omega-polish.ldf (by Alexej M. Kryukov and Dmitry Ivanov) in the Antomega project. See also sections 13.144 and 13.146 on the next page.

13.146 “Polski” language: polski.mld

The titles for the “polski” language (variant for polish) are taken from the polski.dtx file (by Mariusz Olko and Marcin Woliński) in the polski package. See also sections 13.144 to 13.145 on the preceding page.

44“Polish2” is a variant of “polish”.
The name “portuges” is another spelling for “portuguese” (see section 13.148), so we just load portuguese.mld:

\ProvidesFile{portuguese.mld}[2005/06/07]\mtcselectlanguage{portuguese}%

The titles for the “portuguese” language (português) are taken from the portuges.dtx file (by Jose Pedro Ramalhete) in the babel package [60, 61, 92]. The portuguese language is spoken in Portugal (with the islands of Azores and Madeira), in Brazil, and in former portuguese colonies like Angola, Guinea-Bissau, Mozambique, Cape Verde Islands, São Tomé and Príncipe Islands, East Timor, and some old trading posts like Macao and Goa. See also section 13.25 on page 481, because the titles are different in Brazil, even if the language is also portuguese.
13.149 “Romanian” language: romanian.mld

The titles for the “romanian” language (română) come from the romanian.dtx file (by Umstatter Horst and Robert Juhasz) in the babel package [60, 61, 78]. See also sections 13.150 to 13.151 on pages 538–539.

13.150 “Romanian2” language: romanian2.mld

The titles for the “romanian2” language come from the romanian.dtx file (by Adrian Rezus and Bernd Raichle) in the RomanianTEX package [397]. See also sections 13.149 and 13.151 on the following page. Alas, RomanianTEX is not compatible with the babel package [60, 61].
13.151 “Romanian3” language: romanian3.mld

The titles for the “romanian3” language come from the romanian.dtx file (by Adrian Rezus and Bernd Raichle) in the RomanianTEX package [397]. See also sections 13.149 to 13.150 on the page before. Alas, RomanianTEX is not compatible with the babel package [60, 61].

\ProvidesFile{romanian3.mld}[2006/08/03]%
%% Romanian titles from RomanianTeX (romanian.dtx) variant.
%% Adrian Rezus (adriaan@cs.kun.nl)
%% Bernd Raichle (raichle@azu.Informatik.Uni-Stuttgart.de)
def\ptctitle{Tabla de materii}%
def\plttitle{Indice de figuri}%
def\mtctitle{Tabla de materii}%
def\mlttitle{Indice de figuri}%
def\stctitle{Tabla de materii}%
def\slttitle{Indice de figuri}%

13.152 “Russian” language: russian.mld

\ProvidesFile{russian.mld}[1999/03/16]%
%% Russian titles
%% Russian titles
\def\ptctitle{Oglavlenie}%
def\plftitle{Pere\textcz\textmz\ en\textcz\textmz\ risunkov}%
def\plttitle{Pere\textcz\textmz\ en\textcz\textmz\ tablic}%
def\mtctitle{Oglavlenie}%
def\mlftitle{Pere\textcz\textmz\ en\textcz\textmz\ risunkov}%
def\mlttitle{Pere\textcz\textmz\ en\textcz\textmz\ tablic}%
def\stctitle{Oglavlenie}%
def\slftitle{Pere\textcz\textmz\ en\textcz\textmz\ risunkov}%
def\slttitle{Pere\textcz\textmz\ en\textcz\textmz\ tablic}%

\textcz The titles
\textmz for the “russian” language (russkiy yazyk) are taken from the babel package [60, 61]. Specific cyrillic fonts are required.
13.153 “Russian2m” language: russian2m.mld

The titles for the “russian2m” language (“russian2m” is a modern variant of “russian”) are taken from the russian2m.ldf file (by Alexej M. Kryukov and Dmitry Ivanov) in the Antomega project [272]. Specific cyrillic fonts are required. See also section 13.152 on the preceding page.

13.154 “Russian2o” language: russian2o.mld

The titles for the “russian2o” language (“russian2o” is an old variant of “russian”) are taken from the omega-russian.ldf file (by Alexej M. Kryukov and Dmitry Ivanov) in the Antomega project [272]. Specific cyrillic fonts are required. See also section 13.152 on page 539.
13.155 "Russianb" language: russianb.mld

\cyr  The titles for the "russianb" language ("russianb" is a variant of "russian") are taken from the russianb.dtx file (by Olga G. Lapko, Vladimir Volovich, Werner Lemberg, and Irina A. Makhovera) in the babel package [60, 61, 84, 286]. Specific cyrillic fonts are required. See also section 13.152 on page 539. The parttoc title varies depending on the presence of chapters defined or not by the document class.
13.156 “Russianc” language: russianc.mld

The titles for the “russianc” language (“russianc” is a variant of “russian”, used in the part of Mongolia under Russian influence) are taken from the file russian.def in the MonTEX package [137, 140]. Specific Cyrillic fonts are required. See also section 13.152 on page 539.

13.157 “Russian-cca” language: russian-cca.ml[d|o]

They are several variants for the Russian titles with the cmcyrlat fonts. The titles for a first variant of the “russian-cca” are taken from the russian.sty (by Victor Boyko and Vadim Maslov) file in the cmcyrlat package [53].
The titles for the “russian-cca” language contain characters that cannot be easily generated, hence we load russian-cca.mlo.

13.158 “Russian-cca1” language: russian-cca1.ml[d;o]

They are several variants for the russian titles with the cmcyrlt fonts. The titles for the “russian-cca1” language are taken from the cmcyrlt.sty file (by Vadim Maslov, Alexander Harin and Vadim V. Zhytnikov) in the cmcyrlt package[222].

The titles for the “russian-cca1” language contain characters that cannot be easily generated, hence we load russian-cca1.mlo.

13.159 “Russian-lh” language: russian-lh.ml[d;o]

The russian titles for the LH fonts (“russian-lh” language) are taken from the russian.sty file (by Sergei O. Naumov) in the LH package [342].

The titles for the “russian-lh” language contain characters that cannot be easily generated, hence we load russian-lh.mlo.
13.160 **“Russian-lhcyralt” language:**

`russian-lhcyralt.ml[d|o]`

The Russian titles for the LHCYRALT fonts (“russian-lhcyralt” language) are taken from the `lhcyralt.sty` file (by Vadim V. Zhytnikov) in the `lhcyr` package [487].

The titles for the “russian-lhcyralt” language contain characters that cannot be easily generated, hence we load `russian-lhcyralt.mlo`. The input encoding is ALT (code page CP866).

\mtcloadmlo
\ProvidesFile{russian-lhcyralt.mld}[2006/03/10]\mtcloadmlo{russian-lhcyralt}%
\% Russian-lhcyralt titles from lhcyralt.sty in the LHCYR package
\% LHCYRALT fonts in special encoding ALT (CP866).
\% Vadim V. Zhytnikov (vvzhy@td.lpi.ac.ru)
\% (/russian-lhcyralt)

13.161 **“Russian-lhcyrkoi” language:**

`russian-lhcyrkoi.ml[d|o]`

The Russian titles for the LHCYRKOI fonts (“russian-lhcyrkoi” language) are taken from the `lhcyrkoi.sty` file (by Vadim V. Zhytnikov) in the `lhcyr` package [487].

The titles for the “russian-lhcyrkoi” language contain characters that cannot be easily generated, hence we load `russian-lhcyrkoi.mlo`. The input encoding is KOI-8.

\mtcloadmlo
\ProvidesFile{russian-lhcyrkoi.mld}[2006/03/13]\mtcloadmlo{russian-lhcyrkoi}%
\% Russian-lhcyrkoi titles from lhcyrkoi.sty in the LHCYR package
\% LHCYRKOI fonts in special encoding KOI-8. Vadim V. Zhytnikov (vvzhy@td.lpi.ac.ru)
\% (/russian-lhcyrkoi)

13.162 **“Russian-lhcyrwin” language:**

`russian-lhcyrwin.ml[d|o]`

The Russian titles for the LHCYRWIN fonts (“russian-lhcyrwin” language) are taken from the `lhcyrwin.sty` file (by Vadim V. Zhytnikov) in the `lhcyr` package [487].

The titles for the “russian-lhcyrwin” language contain characters that cannot be easily generated, hence we load `russian-lhcyrwin.mlo`. The input encoding is CP1251.
13.163 “Samin” language: samin.mld

The titles for the “samin” language come from the samin.dtx file (by Regnor Jernsletten) in the babel package [60, 61, 79]. Specific fonts are required. Note that several Sámi dialects/languages are spoken in Finland, Norway, Sweden, and on the Kola Peninsula (Russia). The alphabets differ, so there will eventually be a need for more .dtx files for, e.g., Lule and South Sámi. Hence the (artificial) name samin.dtx (and not sami.dtx or the like) in the North Sámi case. These dialects and languages are part of the Finnic group. See also http://en.wikipedia.org/wiki/Sápmi_(area).

13.164 “Scottish” language: scottish.mld

The titles for the “scottish” language (gaelic scottish, gàidhlig) come from the scottish.dtx file (by Fraser Grant) in the babel language [60, 61, 76]:
13.165 “Serbian” language: serbian.mld

The titles for the “serbian” (serbocroatian) (srpski jezik, srpskohrvatski jezik) language are taken from the serbian.dtx file (by Dejan Muhamedagić and Jankovic Slobodan) in the babel package [60, 61, 88]. Serbocroatian is spoken by Serbs, Croats and Chernogors, but only Serbs and Chernogors use the cyrillic alphabet (a variant). See also section 13.166 on the following page.

13.166 “Serbian” language: serbianc.mld

The titles for the “serbianc” language have been gently provided by Marko Ėehaja and Frank Küster. Cyrillic fonts are required. Serbocroatian is spoken by Serbs, Croats and Chernogors, but only Serbs and Chernogors use the cyrillic alphabet (a variant). See also section 13.165 on the page before.

46 The “serbianc” language is written with cyrillic characters.
13.167 “Slovak” language: slovak.mld

The titles for the “slovak” language (slovenčina, sloven˘ k˘ jazyk) are taken from the slovak.dtx file (Jana Chlebîková and Tobias Schlemmer) in the babel package [60, 61, 72].

13.168 “Slovene” language: slovene.mld

The slovene language (slovenšˇ cina, slovenski jezik) is spoken in Slovenia, but somewhat also in Italy (Frioul), in Austria (Carinthia and Styria), in Hungary (Szlovénviék and Porabje), in West Germany and Sweden. The titles for the “slovene” language come from the slovene.dtx file (by Danilo Zavrtanik and Leon Žlajpah) in the babel package [60, 61, 102]:
13.169 “Spanish” language: spanish.mld

The titles for the “spanish” (español, castellano) language are taken from the spanish.dtx file (by Javier Bezos, initially by Julio Sánchez) in the babel package [48, 60, 61]. Note that the “spanish” language is in fact “castillan” (see section 13.35 on page 486). But note also that other languages are spoken in Spain: “basque” (section 13.19 on page 479), “catalan” (section 13.37 on page 486), and “galician” (section 13.66 on page 499). Note that “spanish2” is a version of “spanish” with shorter titles (see section 13.170 on the following page). And “spanish3” (see section 13.171 on the next page) is a version for the Antomega [272] project; some titles are different. And “spanish4” is a variant of “spanish” where \ptctitle is shorter for articles (section 13.172 on page 550).

% 9886 \ProvidesFile{spanish.mld}\[2008/04/03]\%
% Spanish titles from spanish.dtx (babel) by Bezos, Javier & CervanTeX
9887 \expandafter\ifx\csname chapter\endcsname chapter\relax
9888 \def\ptctitle{\'Indice}\else\def\ptctitle{\'Indice general}\fi
9889 \def\plftitle{\'Indice de figuras}\%
9890 \def\plttitle{\'Indice de tablas}\%
9891 \def\mtctitle{\'Indice}\%
9892 \def\mlftitle{\'Indice de figuras}\%
9893 \def\mlttitle{\'Indice de tablas}\%
9894 \def\stctitle{\'Indice}\%
9895 \def\slftitle{\'Indice de figuras}\%
9896 \def\slttitle{\'Indice de tablas}\%
9897 (/spanish)
9900 (/spanish)
13.170 “Spanish2” language: spanish2.mld

The titles for the “spanish2” language are taken from the spanish.dtx file in the babel package [48, 60, 61], but made shorter for chapter and section levels. See section 13.169 on the page before.

13.171 “Spanish3” language: spanish3.mld

The titles for the “spanish3” language are taken from the omega-spanish.ldf file (by Alexej M. Kryukov) of the Antomega project [272]. See section 13.169 on the preceding page.

13.172 “Spanish4” language: spanish4.mld

The titles for the “spanish4” language are taken from the spanish.dtx file (by Javier Bezos) from the CervanTeX package [47]. The title of the parttoc is shorter for articles. See also section 13.169 on page 548.
13.173  “Swahili” language: swahili.mld

The titles for the “swahili” language (kiswahili) are taken from the obsolete swahili.tex file\(^47\), with adaptations and corrections given on the comp.text.tex news group (messages 57662, 57713, and 57717) by Giancarlo Bassi and Enrico Gregorio. Swahili is the main Bantu language and is spoken in East Africa: Tanzania, Kenya, Uganda, Rwanda, Burundi, Zanzibar and in the area of the Great Lakes in the Democratic Republic of Congo (Congo-Kinshasa, formerly Zaire) and in the Republic of Congo (Congo-Brazzaville), in the north of Mozambique and the south of Somalia\(^48\). See [135, page 991].

\(^{47}\)http://mirror.ctan.org/obsolete/macros/latex209/contrib/ml/swahili.tex

13.174 “Swedish” language: swedish.mld

The titles for the “swedish” (svenska) language come from the swedish.dtx file (by Sten Hellman and Erik Östhols, with a correction by Jan Michael Rynning) in the babel package [60, 61, 77]. The swedish language is spoken in Sweden and in some regions of Finland like the Åland Islands. See also section 13.175.

13.175 “Swedish2” language: swedish2.mld

The titles for the “swedish2” language (variant for swedish) are taken from the rapport.doc file (by Sven Mattisson) in the SLATEX package [318]. See also section 13.174.

13.176 “Thai” language: thai.ml[d|o]
The titles for the “thai” language come from the thaicjk.1df file (by Werner Lemberg) and use fonts of the CJK system [127, 297, 298]. The thailatex package [320] (by Suraphant Meknavin, Theppitak Karoonboonyanan, Chanop Silpa-Anan and Veerathanabutr Poonlap) provides the same titles in its thai.1df file.

\mtcloadmlo The titles for the “thai” language contain characters that cannot be easily generated, hence we load thai.mlo. See also [255].

13.177 “Turkish” language: turkish.mld

The turkish language (türkçe) is spoken mainly in Turkey and in Cyprus. The titles for the “turkish” language are taken from the turkish.dtx file (by Mustafa Burç, Pierre A. MacKay and Turgut Uyar) in the babel package [60, 61, 68].

13.178 “Uighur” language: uighur.mld

\mtcselectlanguage The “uighur” and “bicig” languages are synonyms, so we just load the bicig.mld file (see section 13.21 on page 479):
13.179 “Uighur2” language: uighur2.mld

\mtcselectlanguage  The “uighur2” and “bicig2” languages are synonyms, so we just load the bicig2.mld file (see section 13.22 on page 480):

10003 ⟨+uighur2⟩
10004 \ProvidesFile{uighur2.mld}[2006/05/31]\mtcselectlanguage{bicig2}%
10005 ⟨/uighur2⟩

13.180 “Uighur3” language: uighur3.mld

\mtcselectlanguage  The “uighur3” and “bicig3” languages are synonyms, so we just load the bicig3.mld file (see section 13.21 on page 479):

10006 ⟨+uighur3⟩
10007 \ProvidesFile{uighur3.mld}[2006/05/31]\mtcselectlanguage{bicig3}%
10008 ⟨/uighur3⟩

13.181 “UKenglish” language: UKenglish.mld

\mtcselectlanguage  The “UKenglish” language is just like “english” (“UK” is for “United Kingdom”), so we just load english.mld (see section 13.45 on page 490):

10009 ⟨+UKenglish⟩
10010 \ProvidesFile{UKenglish.mld}[2005/07/11]\mtcselectlanguage{english}%
10011 ⟨/UKenglish⟩

13.182 “Ukraineb” language: ukraineb.mld

\mtcselectlanguage  The “ukraineb” language is a synonym for “ukrainian”, so we just load ukrainian.mld. See section 13.183 on the following page.
13.183 “Ukrainian” language: ukrainian.mld

The titles for the “ukrainian” language (ukrayins’ka mova) come from the ukraineb.dtx file (by Olga G. Lapko, Andrij M. Shvaika, Vladimir Volovich, and Werner Lemberg) in the babel package [60, 61, 97]. Cyrillic fonts are required. Another language name is ukraineb (see section 13.182 on the page before).

13.184 “Uppersorbian” language: uppersorbian.mld

The titles for the “uppersorbian” language\footnote{Upper sorbian. Sorbian, or wendisch, is a member of the west slavic subgroup of indo-european languages spoken in Upper Lusatia in the german länder of Saxony and Brandenburg. The Sorbs are descendents of the Wends, the german name for the slavic tribes who occupied the area between the Elbe and Saale rivers in the west and the Odra (Oder) river in the east during the medieval period (vr-th century).} (hornjoserbsce, hornjoserbš´ ciba) are taken from the usorbian.dtx file (by Eduard Werner) in the babel package [60, 61, 100]. See
also section 13.113 on page 522. A shorter language name is usorbian (see section 13.186 on the next page).

\mtcselectlanguage\ProvidesFile{uppersorbian.mld}[2006/02/38]%
\def\ptctitle{Wobsah}%
\def\plftitle{Zapis wobrazow}%
\def\plttitle{Zapis tabulkow}%
\def\mtctitle{Wobsah}%
\def\mlftitle{Zapis wobrazow}%
\def\mlttitle{Zapis tabulkow}%
\def\stctitle{Wobsah}%
\def\slftitle{Zapis wobrazow}%
\def\slttitle{Zapis tabulkow}%
\mtcselectlanguage{uppersorbian}
\mtcselectlanguage

13.185 “USenglish” language: USenglish.mld

\mtcselectlanguage The “USenglish” language (“US” is for “United States (of America)” is just like “english”\footnote{It should be true for the mini-table titles; the languages themselves have some differences, like the hyphenation rules, see \url{http://en.wikipedia.org/wiki/American_English}.}, so we just load english.mld (see section 13.45 on page 490):

\mtcselectlanguage\ProvidesFile{USenglish.mld}[2005/07/11]\mtcselectlanguage{english}%
\mtcselectlanguage

13.186 “Usorbian” language: usorbian.mld

\mtcselectlanguage The “usorbian” language is a synonym for “uppersorbian”, so we just have to load uppersorbian.mld. See section 13.184 on the page before.

\mtcselectlanguage\ProvidesFile{usorbian.mld}[2007/12/04]\mtcselectlanguage{uppersorbian}%
\mtcselectlanguage


The titles for the “vietnam” language (tiếng việt) are taken from the vietnam package \cite{vietnam} (by Werner Lemberg and Thế Thành Hàn). Vietnamese fonts are required; see \cite{vietnam, vietnam}.
The Vietnamese language is spoken in Vietnam and in the Vietnamese diaspora. See also section 13.188 on the following page.

13.188 “Vietnamese” language: vietnamese.mld

The “Vietnamese” language is just a synonym for the “vietnam” language. So we just load vietnam.mld. Vietnamese fonts are required. See also section 13.187 on the page before.

13.189 “Welsh” language: welsh.mld

The titles for the “Welsh” language (cymraeg) come from the welsh.dtx file (by Johannes L. Braams) in the babel package [59–61]:

The vietnamese language is spoken in Vietnam and in the vietnamese diaspora. See also section 13.188 on the following page.
The titles for the “xalx” language are taken from the Mon\TeX package \cite{137,140} (by Oliver Corff and Dorjpalam Dorj). Xalx (Khalkha) is the name of the Mongolian nationality residing in Mongolia proper.

Their dialect forms the basis of Mongolian written with Cyrillic letters. See also sections 13.102 on page 517 and 13.191 to 13.192 on pages 557–558.

The titles for the “xalx2” language are taken from the Mon\TeX package \cite{137,140} (by Oliver Corff and Dorjpalam Dorj). This is a variant for the “xalx” language (see section 13.190).
The titles for the “xalx3” language are taken from the MonTeX package [137, 140] (by Oliver Corfū and Dorjpalam Dorj). This is an other variant for the “xalx” language (see section 13.190 on the preceding page).

\xalx  The titles for the “xalx3” language are taken from the MonTeX package [137, 140] (by Oliver Corfū and Dorjpalam Dorj). This is an other variant for the “xalx” language (see section 13.190 on the preceding page).

\ProvidesFile{xalx3.mld}[2006/03/31]
\% Mongol (xalx3) titles
\def\ptctitle{xalx{Soderjanie}}
\def\plftitle{xalx{Spisok risunkow}}
\def\plttitle{xalx{Spisok tablic}}
\def\mtctitle{xalx{Soderjanie}}
\def\mlftitle{xalx{Spisok risunkow}}
\def\mltttitle{xalx{Spisok tablic}}
\def\stctitle{xalx{Soderjanie}}
\def\slltitle{xalx{Spisok risunkow}}
\def\sltttitle{xalx{Spisok tablic}}
\end
Part III

Complements
Contents of the Third Part

<table>
<thead>
<tr>
<th>Bibliography</th>
<th>561</th>
<th>Acknowledgments</th>
<th>634</th>
</tr>
</thead>
<tbody>
<tr>
<td>Changes history</td>
<td>597</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
This bibliography contains many URLs; you must be aware that some of them might be inaccessible because they are obsolete, or because their site is down or encounters some unexpected problem. Note also that the response of some sites may be slow (several seconds). For instance, the entries [257–259], from the http://www.geocities.com/kijoo2000/ site, are very difficult to contact.

The URLs beginning with “https:” to the TUGboat site may have a restricted access to the TUG members during one year after publication. Being a member of TUG is useful and cheap!

Some URLs may contain an extension not supported by your Web browser; in such cases, you should try to access to the document manually. An example is [29], whose extension is .ps.gz.

Some URLs are too long for some tools; when possible, I shortened the URL to display only the contents the directory, as for [209], or by using an URL to the catalogue entry (as for [243] and [272]); when not possible, you should try to access to the document manually.


mailto:indicTeX@gmail.com. Available from: 

Mark Alford. How to create Japanese language documents under GNU/Linux using \LaTeX [online], January 2007. Available from: 


American Mathematical Society. Instructions for Preparation of Papers and 


Jacques Andrè and Yannis Haralambous. Fontes numériques. Document numérique, 

http://hapax.qc.ca.

Walter Appel, Céline Chevalier, Emmanuel Cornet, Sébastien Desreux, Jean-Julien 
Fleck, and Paul Pichaureau. \LaTeX pour l’impatient. In Céline Chevalier, editor, 
mailto:contact@H-K.fr, see http://www.h-k.fr/liens/tp/latex_pour_l_impatient.html.

Jouko Arponen. Practical \LaTeX Guide [online], August 2004. Available from: 

Donald Arseneau. The notoccite package, July 2000. mailto:asnd@reg.triumf.ca. 

Donald Arseneau. The placeins package, June 2002. mailto:asnd@reg.triumf.ca. 
Available from: http://mirror.ctan.org/macros/latex/contrib/placeins/. 29, 88, 217, 
219, 234, 251, 417, 424, 441.

Donald Arseneau. The cite package, November 2003. mailto:asnd@reg.triumf.ca. 
Available from: http://mirror.ctan.org/macros/latex/contrib/cite/cite.sty. 627.

Donald Arseneau. The overcite package, May 2003. mailto:asnd@reg.triumf.ca. 

Donald Arseneau. The wrapfig package, January 2003. mailto:asnd@reg.triumf.ca. 
Bibliography


Bibliography


Bibliography


[57] Johannes L. Braams. Irish support from the \texttt{babel} system, March 2005. With help from Marion Gunn and Fraser Grant, Available from: http://mirror.ctan.org/macros/latex/required/babel/irish.dtx. 512


[63] Johannes L. Braams and Einar Árnason. Icelandic support from the babel system, March 2005. mailto:babel@bramms.cistron.nl, mailto:einar@if.lif.hi.is. Available from: http://mirror.ctan.org/macros/latex/required/babel/icelandic.dtx. 511


[65] Johannes L. Braams and Claudio Beccari. Latin support from the babel system, March 2008. mailto:babel@bramms.cistron.nl, mailto:claudio.beccari@polito.it, with help from Krzysztof Konrad Żelechowski. (mailto:kkz@alta.mimuw.edu.pl). Available from: http://mirror.ctan.org/macros/latex/required/babel/latin.dtx. 517, 518, 519


[68] Johannes L. Braams and Mustafa Burc. Turkish support from the babel system, March 2005. mailto:babel@bramms.cistron.nl, mailto:rz6001@rziris01.rz.uni-hamburg.de, with help from Pierre A. MacKay and Turgut Uyar (mailto:uyar@cs.itu.edu.tr). Available from: http://mirror.ctan.org/macros/latex/required/babel/turkish.dtx. 552
Bibliography


[71] Johannes L. Braams, Manuel Carriba, and Javier A. Múgica de Rivera. Galician support from the babel system, October 2007. mailto:babel@bramms.cistron.nl, mailto:mccarriba@eunetcom.net. Available from: http://mirror.ctan.org/macros/latex/required/babel/galician.dtx. 499


[73] Johannes L. Braams, Maurizio Codogno, and Claudio Beccari. Italian support from the babel system, March 2008. mailto:babel@bramms.cistron.nl, mailto:mau@beatles.csielt.stet.it, mailto:beccari@polito.it. Available from: http://mirror.ctan.org/macros/latex/required/babel/italian.dtx. 513


[76] Johannes L. Braams and Fraser Grant. Scottish support from the babel system, March 2005. mailto:babel@bramms.cistron.nl, mailto:fraser@cernvm. Available from: http://mirror.ctan.org/macros/latex/required/babel/scottish.dtx. 546

[77] Johannes L. Braams and Sten Hellman. Swedish support from the babel system, March 2005, mailto:babel@bramms.cistron.nl, mailto:hellman@cernvm.cern.ch; enhancements for version 2.0 by Erik Östhols (mailto:erik_osthols@yahoo.com). Available from: http://mirror.ctan.org/macros/latex/required/babel/swedish.dtx. 551


[80] Johannes L. Braams, Mikko Kanerva, and Keranen Reino. Finnish support from the babel system, October 2007. mailto:babel@bramms.cistron.nl, mailto:kanerva@cernvm, mailto:keranen@cernvm. Available from: http://mirror.ctan.org/macros/latex/required/babel/finnish.dtx. 495, 496


[82] Johannes L. Braams, Jörg Knappen, and Terry Mart. Bahasa Indonesia support from the babel system, March 2008. mailto:babel@bramms.cistron.nl, mailto:joerg.knappen@alpha.ntp.springer.de, mailto:mart@vkpmzd.kph.uni-mainz.de. Available from: http://mirror.ctan.org/macros/latex/required/babel/bahasa.dtx. 477

[83] Johannes L. Braams, Jörg Knappen, Terry Mart, and Bob Margolis. Bahasa Malaysia support from the babel system, March 2008. mailto:babel@bramms.cistron.nl, mailto:joerg.knappen@alpha.ntp.springer.de, mailto:mart@vkpmzd.kph.uni-mainz.de, mailto:bob.margolis@ntlworld.com. Available from: http://mirror.ctan.org/macros/latex/required/babel/bahasam.dtx. 478

[84] Johannes L. Braams, Olga G. Lapko, Vladimír Volovich, and Werner Lemberg. Russian support from the babel system, March 2008. mailto:babel@bramms.cistron.nl, mailto:cyrtug@mir.msk.su, mailto:TeX@vvv.vsu.ru et mailto:wl@gnu.org. Available from: http://mirror.ctan.org/macros/latex/required/babel/russianb.dtx. 484, 541


[94] Johannes L. Braams, Marti Ruiu-Altaba, and Jörg Knappen. *Esperanto support from the babel system*, October 2007. mailto:babel@bramms.cistron.nl, mailto:ruizaltb@cernvm.cern.ch, mailto:knappen@vkpmzd.kph.uni-mainz.de. Available from: http://mirror.ctan.org/macros/latex/required/babel/esperanto.dtx. 492


Bibliography


[116] David P. Carlisle. The file \LaTeX{} \texttt{atxdoc.dtx} for use with \BIBL{\LaTeX}2e, August 1999. Available from: http://mirror.ctan.org/macros/latex/base/atxdoc.dtx. 25, 213, 222
http://mirror.ctan.org/macros/latex/contrib/hyperref/xr-hyper.sty. 241

Available from: http://www.ifi.uio.no/it/latex-links/ifthen.pdf. 219

mailto:carlisle@cs.man.ac.uk. Available from:

mailto:carlisle@cs.man.ac.uk. Available from:
http://mirror.ctan.org/macros/latex/contrib/color.dtx. 115

[121] David P. Carlisle and Sebastian Rahtz. *Back referencing from bibliographical
citations*, October 2006. Available from:


[123] Winston Chang. *$\TeX_2$g Cheat Sheet*, October 2006. Available from:

[124] Jean-Pascal Chauvet. *About Kanji* [online], June 1999. Available from:
http://www2.gol.com/users/jpc/Japan/Kanji/. mailto:jpc@gol.com.

[125] Pehong Chen and Michael A. Harrison. *Index preparation and processing*. *Software

[126] Otfried Cheong. *Using TrueType Fonts and Unicode in $\TeX$* [online], April 2007.
mailto:otfried@tclab.kaist.ac.kr and http://tclab.kaist.ac.kr/~otfried/

Typography. 38, 44, 212, 237, 238, 472, 487, 505, 506, 507, 508, 514, 515, 516, 552

[128] Pai H. Chou. *How to make $\TeX$ (te$\TeX$) handle unicode and CJK in MacOSX* [online],
mailto:chou@ece.uci.edu. 215

[129] André Chuvin, René Létoile, and Sébastien Peyrouse. *Histoire de l’Asie centrale

mailto:sds@cs.cmu.edu, mailto:cochran@ieee.org. Available from:
http://mirror.ctan.org/obsolete/macros/latex/contrib/subfigure/. 33, 88, 140, 228, 237

mailto:sds@cs.cmu.edu, mailto:cochran@ieee.org. Available from:
http://mirror.ctan.org/macros/latex/contrib/captcont/. 54, 76, 88, 210, 254, 277
Bibliography

mailto:sds@cs.cmu.edu, mailto:cochran@ieee.org. Available from: 
http://mirror.ctan.org/macros/latex/contrib/subfig/subfig.pdf. 33, 88, 146, 210, 228, 
237, 395

Imprimerie Nationale, August 2002.


[135] Bernard Comrie, editor. The Worlds’s Major Languages. Oxford University Press, 
USA, June 1990. 550

langues (L’origine et le développement des langues dans le monde). Acropole 

mmailto:corff@zedat.fu-berlin.de. Available from: 
http://mirror.ctan.org/language/mongolian/montex/doc/mlsquick.pdf. 44, 205, 225, 
472, 479, 480, 481, 484, 485, 529, 542, 557, 558

from: http://userpage.fu-berlin.de/~corff/im/Buch/Pentaglot.html. 
mmailto:corff@zedat.fu-berlin.de. 226

[139] Oliver Corff. Infosystem Mongolei [online], July 2004. Available from: 
http://userpage.fu-berlin.de/~corff/infromong.html. An Internet-based Journal on 
Mongolian Affairs. 225

[140] Oliver Corff and Dorjpalam Dorji. MonTeX – Mongolian for BiT\TeX\ $\varepsilon$, Implementation 
Level System Documentation, July 2002. mmailto:corff@zedat.fu-berlin.de. Available 
from: http://mirror.ctan.org/language/mongolian/montex/doc/montex.pdf. 44, 205, 
225, 472, 479, 480, 481, 484, 485, 529, 542, 557, 558

[141] Olivier Dabène and Aurélie Boissière. Atlas de l’Amérique latine: violences, 
démocratie participative et promesse de développement. In Atlas/Monde. Éditions 
Autrement, June 2006.

[142] Sergueï Dachian, Arnak Dalalyan, and Vartan Akopian. Arm\TeX\: a System for Writing 
in Armenian with \TeX\ and Bi\TeX\ (written in armenian), June 1999. 
mmailto:Serguei.Dachian@univ-lemans.fr, 
http://www-scf.usc.edu/~vakopian/programs/progs.html. Available from: 

[143] Patrick W. Daly. Graphics and Colour with Bi\TeX\. Max Plank Institut für Aeronomie; 

Bibliography


[165] Simon Fear. *Publication quality tables in \LaTeX\ (with the booktabs package)*, March 2003. In the absence of the author, contact Danie Els, mailto:dnjels@sun.ac.za. Available from: http://mirror.ctan.org/macros/latex/contrib/booktabs/booktabs.pdf. 21, 210


Bibliography


[217] Yannis Haralambous and John Plaice. *\Omega, a \TeX extension including Unicode and featuring lex-like filtering processes*. In Włodzimierz Bzyl and Tomasz Płaś-Przezchlewski, editors, *Proceedings of the European \TeX Conference*, pages 153–166, Gdańsk, Poland, 1994. GUST.
Bibliography


Bibliography


[239] Hiroya Ikeda. *p\(\LaTeX\)Ε* \(\LaTeX\)Ε2Ε. In japanese, mailto:ike@kobitosan.net, September 2001. Available from: http://www.kobitosan.net/ike/latex/handhtml/index.html. 234


---

This document is somewhat obsolete.


[288] \LaTeX3 Project Team. *\LaTeX2ε Kernel (Output Routine)*, July 2000. Available from: http://mirror.ctan.org/macros/latex/base/ltoutput.dtx. 30, 234, 441


mailto:loreti@padova.infn.it and http://wwwcdf.padova.infn.it/~loreti/mlo.html, University of Padova – Department of Physics, Via F. Marzolo, 8 – 35131 PADOVA – Italy. Available from: http://mirror.ctan.org/macros/latex/contrib/sfheaders/=. 54, 76, 88, 237, 254, 276


[318] Sven Mattisson. *SL\TeX: Swedish speaking version of \LaTeX* [online], November 1989. Available from: http://mirror.ctan.org/language/swedish/slatex/slatex1.tar.gz. mailto:sven@lde.lu.se, Department of Applied Electronics, University of Lund, Box 118, S-221 00 Lund, Sweden. 44, 237, 551
mailto:rowland.mcdonnell@physics.org. Available from: 
http://mirror.ctan.org/macros/latex/contrib/sectsty/.. 53, 70, 88, 129, 236, 252, 275

[320] Surapant Mekavin, Theppitak Karoonboonyanan, Chanop Silpa-Anan, and Veerathanabutr Poonlap. The thailatex package, March 2006. Find thai.dtx in thailatex-0.4.0.tar.gz, mailto:surapan@nectec.or.th. Available from: 
ftp://linux.thai.net/pub/ThaiLinux/software/thailatex/. 238, 552

[321] Frank Mittelbach. \$\$$\textit{e}$$\textup{-}\TeX\$: Guidelines for future $\TeX$. TUGboat, 11(3):337–345, September 1990. Available from: 


[323] Frank Mittelbach. $\$$\textit{Bi}$$\La$$\TeX$$ \textup{2}$ Encoding Interfaces, June 1995. Available from: 
http://www.latex-project.org/papers/encoding-concepts.pdf. 215, 229

[324] Frank Mittelbach. Producing slides with $\$$\textit{Bi}$$\La$$\TeX$$ \textup{2}$, August 1997. Available from: 
http://mirror.ctan.org/macros/latex/base/slides.dtx. 213

mailto:Frank.Mittelbach@latex-project.org. Available from: 
http://mirror.ctan.org/macros/latex/base/required/tools/multicol.pdf. 91, 115, 229

http://mirror.ctan.org/macros/latex/base/required/tools/varioref.pdf. 140

[327] Frank Mittelbach. The \texttt{doc} and \texttt{shortverb} packages, January 2004. Available from: 
http://mirror.ctan.org/macros/latex/base/doc.dtx. 21

mailto:frank.mittelbach@latex-project.org, The current maintainer is Werner Lemberg, mailto:wl@gnu.org. Available from: 
http://mirror.ctan.org/language/vietnamese/vntex/tex/latex/vntex/dblaccnt.sty. 214, 617


[339] Michael A. Morrison. \textit{\LaTeX\ Links} [online], December 1999. Available from: http://www.nhn.ou.edu/~morrison/LaTeX/index.shtml, mailto:morrison@mail.nhn.ou.edu, Department of Physics & Astronomy, University of Oklahoma.


Bibliography


[397] Adrian Rezuş and Bernd Raichle. Romanian \TeX{}, October 1996. mailto:adriaan@cs.kun.nl. Available from: http://mirror.ctan.org/language/romanian/RomanianTeX/romanian.dtx. 538, 539


Bibliography


Bibliography


Bibliography

mailto:nlct@cmp.uea.ac.uk, School of Computing Sciences, University of East Anglia, Norwich. NR4 7TJ, United Kingdom. Available from: 

mailto:nlct@cmp.uea.ac.uk, School of Computing Sciences, University of East Anglia, Norwich. NR4 7TJ, United Kingdom. Available from: 
http://theoval.sys.uea.ac.uk/~nlct/latex/admin/admin_a4.pdf.

mailto:nlct@cmp.uea.ac.uk, School of Computing Sciences, University of East Anglia, Norwich. NR4 7TJ, United Kingdom. Available from: 

mailto:nlct@cmp.uea.ac.uk, School of Computing Sciences, University of East Anglia, Norwich. NR4 7TJ, United Kingdom. Available from: 
http://theoval.sys.uea.ac.uk/~nlct/latex/minexample/minexample.pdf. 57, 222

mailto:nlct@cmp.uea.ac.uk, School of Computing Sciences, University of East Anglia, Norwich. NR4 7TJ, United Kingdom. Available from: 

mailto:nlct@cmp.uea.ac.uk, School of Computing Sciences, University of East Anglia, Norwich. NR4 7TJ, United Kingdom. Available from: 

mailto:nlct@cmp.uea.ac.uk, School of Computing Sciences, University of East Anglia, Norwich. NR4 7TJ, United Kingdom. Available from: 

mailto:nlct@cmp.uea.ac.uk, School of Computing Sciences, University of East Anglia, Norwich. NR4 7TJ, United Kingdom. Available from: 

http://mirror.ctan.org/macros/latex/contrib/taupin/varsects.sty. 53, 73, 88, 240, 253, 275


52 TWS-TDS = TUG Working Group on a \TeX\ Directory Structure (TWG-TDS).


Changes history

* version 00

- 1990/10/01
  - Original version, by Nigel Ward.
- 1991/11/01
  - Revised to reuse `\chapter`, `\section`, `\subsection` commands transparently, generate toc-file-name automatically, assorted other cleanup (by Dan Jurafsky).

* version 01

- 1993/06/01
  - New design, to avoid allocating a newwrite, or file descriptor, for each chapter (a deadly sin!) (Jean-Pierre F. Drucbert).
  - Added `\chapterend` to terminate the scope of a minitoc. *(If you forgot putting `\chapterend` at the end of each chapter, an entry for the next chapter will appear in each minitoc.)* (Thanks to Yufan Hu).
  - Replaced the `minipage` environment by a `verse` environment, to allow a minitoc being split across pages.
  - All the layout of the minitoc is in the command, so if somebody wants to redefine that layout, he has just to rewrite it (and only it).
  - You can inhibit the minitoc for the next chapter by preceding it with `\minitocno`. *(`\minitocyes` is useless for the user, because it is implicit after the `\chapter` pseudo-chapters).*
  - Problems: you *must* have `\chapterend` to terminate each chapter with a minitoc. How about avoiding this constraint?
  - The depth of the minitoc is user-adjustable with the counter `minitocdepth` (similar to `tocdepth` for the table of contents).
  - At least three passes (3) of \LaTeX are necessary to get correct minitocs (the first pass creates the `.mtc(X)` files, the second uses them (but they may contain wrong page numbers) and recreates them, the third should be ok).
  - Works with `\chapter{xxx}{yyy}` and floating bodies. Works with two columns (but the minitoc is composed in one column; how to make it to spread over the two columns?).
  - Some mods added to work with `xr` (external references). `xr` version 5 is much more tolerant.
Changes history

⋆ version 02
• 1993/07/05
  ○ Added compatibility with hangcaption (the package hangcaption (if present) must be loaded before the minitoc package). Beware to options modifying \@caption.

※ version 03
• 1993/07/09
  ○ Version 3 not released (buggy).

⋆ version 04
• 1993/07/09
  ○ Added \if@realch to avoid contents lines from pseudo-chapters to go into the toc.
  ○ The package mtcff allows you to use a \LaTeX{} document with minitoc commands and to make them transparent: just replace the minitoc package by mtcff.

⋆ version 05
• 1993/07/13
  ○ Added a selection mechanism to not write spurious things in the minitocs.

⋆ version 06
• 1993/07/15
  ○ Fixed problems about chapters in the toc, removed obsolete \caption stuff (filters are much better) added compatibility with toch (toch makes a table of chapters; if used, must be loaded before minitoc; in fact, it is the ancestor of the shorttoc [155] package).

⋆ version 07
• 1993/07/22
  ○ (major differences) Completely rewritten, using tricks from xr (the version 5, by David P. Carlisle). The info for minitocs is directly stolen from the .toc file. \chapterend and \minitocno are suppressed, \minitoc, \dominitoc and \faketableofcontents added.

⋆ version 08
• 1993/07/29
  ○ Spacing adjustments.
Changes history

* version 09
  • 1993/08/04
    ○ Added mods for MS-DOS (search for MS-DOS, uncomment; search for UNIX, comment out). MS-DOS allows only 3 characters for extensions in file names (what a pity!).

* version 10
  • 1993/08/05
    ○ Works now with appendices. Detects now the obsolete versions of \texttt{latex.tex} (\texttt{@inputcheck} or \texttt{reset@font} not defined).

* version 11
  • 1993/08/18
    ○ Added \texttt{mtcSfont}, font for section entries, \texttt{mtcSSfont} for subsection entries, \texttt{mtcSSSfont} for subsubsection entries, \texttt{mtcPfont} for paragraph entries, \texttt{mtcSPfont} for subparagraph entries.

* version 12
  • 1993/12/16
    ○ Use \texttt{kern} in place of \texttt{vspace}, and added penalties (\texttt{nopagebreak}) to avoid a page break just before last \texttt{mtc@rule}.
    ○ Also added a \texttt{samepage} environment.
    ○ Removed old commented out lines from previous versions.

* version 13
  • 1993/12/17
    ○ Added \texttt{minilof} and \texttt{minilot} stuff. For MS-DOS, uncomment the definition of \texttt{SHORTEXT}.

* version 14
  • 1994/01/03
    ○ Corrected space under \texttt{minitoc/lof/lot} and added a \texttt{raggedright} setting to avoid "underfull" warnings.
    ○ Corrected some spacing problems (avoiding ~'s). \texttt{mtifont} is changed from \texttt{\normalsize\bf} to \texttt{\large\bf}.
    ○ Some modifications suggested by Donald Arsenneau (thanks): \texttt{@newread} becomes \texttt{newread}, not outer version of \texttt{newread}; \texttt{empty} replaced by \texttt{relax} in the spare definition of \texttt{reset@font}.
    ○ Removed the setting of \texttt{clubpenalty} and \texttt{widowpenalty} to 10000 (done by \texttt{samepage}), and \texttt{noindent}.
    ○ Simplified processing of optional argument in \texttt{minitoc, minilof} and \texttt{minilot}. 
Changes history

* version 15
  * 1994/01/27
    - Added `\parttoc`, `\partlof` and `\partlot` for books, `\secttoc`, `\sectlof` and `\sectlot` for articles, with some commands and parameters parallel to those for mini-tables.
  * 1994/01/27
    - Added `\parttoc`, `\partlof` and `\partlot` for books, `\secttoc`, `\sectlof` and `\sectlot` for articles, with some commands and parameters parallel to those for mini-tables.

* version 16
  * 1994/02/02
    - Bug fixes (typos).

* version 17
  * 1994/06/23
    - ‘n’ (null) synonym of ‘e’ (empty) in the optional argument of `\minitoc`, `\dominitoc`, and siblings.
    - Compatibility with “L\TeX\ 2\epsilon”. Thanks to Denis B. Roegel (who found the problem) and Frank Mittelbach (who gave the hints to solve).

* version 18
  * 1994/06/26
    - Make `\minitoc` really compatible with L\TeX\ 2\epsilon.
    - Introduce the language files as options. Many thanks to Michel Goossens (via Frank Mittelbach) who was inspired by the code of the babel package (by Johannes L. Braams).

* version 19
  * 1994/08/16
    - Added stuff for numbering of chapters (parts, sections) not starting at 1. \firstchapteris etc. commands added.
    - `\mtcrule`, `\nomtcrule` etc. commands added.
    - Corrected a bug in `\c@mti`.
    - Corrected `mtcswegian.sty` (Jan Michael Rynning).
    - Corrected appendix in articles.

* version 20
  * 1994/08/25
    - Corrected spacing before and after minitocs and siblings.
    - Added the `\mtcpagenumbers` and `\nomtcpagenumbers` commands (and siblings) to make minitocs with/without page numbers. Default: with page numbers.
Changes history

○ Corrected (difficult bug) appendix in articles.
○ Corrected vertical spacing.
○ Corrected a problem with chapters numbered with uppercase roman numbers.

* version 21

• 1994/09/07
  ○ Corrected typos in minitoc and minitoc.tex.

* version 22

• 1994/10/10
  ○ Corrected typos in minitoc.

* version 23

• 1994/11/08
  ○ Added a missing line in \sectlof.
  ○ Works with document classes resetting chapter (or section) number at each part (thanks to Denis B. Roegel).
  ○ Added the notion of “absolute numbering” for the mini-tables.
  ○ Removed stuff for \firstchapteris and co. These commands are obsolete.
  ○ Removed appendix stuff.

* version 24

• 1994/12/21
  ○ The \protect commands have been removed from the .toc, .lot and .lot files, so some internal macros have been corrected to be compatible with the \TeX{} 2ε release of December 1994. Thanks to Denis B. Roegel who did the work.

* version 25

• 1996/09/13
  ○ Updated mtcnorsk.sty and added mtcynorsk.sty on a suggestion from Dag Langmyhr.

* version 26

• 1996/11/14
  ○ Language specific files are now named language.mld (replacing names of the form mtc\language.sty) because they are not packages and it makes shorter names.
  ○ Added breton, estonian, germanb, greek, irish, russianb, scottish, lower and upper sorbian; renamed “esperanto” by “esperant” like in the babel package.
Changes history

★ version 27

- 1996/12/20
  - Corrections for starred sectioning commands.
  - english.mld loaded as default language.
  - Added vietnam.mld and arab.mld.
  - Renamed minitocoff into mtcoff to keep the name short.

★ version 28

- 1997/10/29
  - Added the afrikaan(s), brazil, and ethiopia(n) languages.
  - Added autoconfiguration of extensions.
  - Added the shorttext package option.
  - Added coffee stuff.
  - Added \addstarred stuff (for starred chapter stuff).
  - Fixed bug in parttocs.
- 1998/06/15
  - A typo corrected by Donald Arseneau:
    \{\let@dottedtocline\@undottedtocline\}\}
    should probably be
    \{\let\dottedtocline\@undottedtocline\}\}
    (a backslash was missing after \let). Thanks to him.
  - Added the bahasa language.
- 1998/12/03
  - Added the tight and loose package options.

★ version 29

- 1999/03/16
  - Added the bicig, buryat, mongol and russianc languages.
- 1999/06/28
  - Added the armenian language (from ArmTeX [142]).
- 1999/07/23
  - Added the dotted/undotted package options (default: dotted).
- 1999/07/29
  - Added lithuanian.mld.

★ version 30

- 1999/12/06
  - Added the basque, ngermanb, serbian, ukraineb, and welsh languages.
  - Corrected a bug in \sltname definition (mlt should be slt).
Changes history

* version 31
  • 2000/04/04
    ○ Added compatibility with the \hyperref{} package, thanks to Heiko Oberdiek, who has also simplified some code and fixed the infamous \chapter* bug.

* version 32
  • 2000/08/08
    ○ Added very (too) numerous new commands for the mini-table features: \beforeparttoc, \beforepartlof, \beforepartlot, \afterparttoc, \afterpartlof, \afterpartlot, \thispageparttocstyle, \thispagepartlofstyle, and \thispagepartlotstyle.
    ○ Documentation improved by Stefan Ulrich.
    ○ \nomtcrule corrected.

* version 33
  • 2000/12/07
    ○ Added new adjustment commands: \mtcaddchapter, \mtcaddsection, and \mtcaddpart. These commands add stuff in the .toc, .lof and .lot files for the \chapter* (\section* and \part*) problem. From a suggestion by Karl F. Everitt.
  • 2000/12/08
    ○ Corrected a feature in \mtcaddchapter and co. with a blank optional argument.

* version 34
  • 2001/02/26
    ○ Added bulgarian.mld, hebrew.mld, icelandic.mld, latin.mld, and samin.mld.
  • 2001/03/09
    ○ Added \mtcselectlanguage.
  • 2001/06/01
    ○ Fixed the estonian package option (missing).
  • 2001/07/04
    ○ Added the interlingua language.
Changes history

* version 36

- 2002/02/11
  - Corrected an interaction with `\tableofcontents` which creates a `\chapter*` or a `\section*`, perturbing mtc/stc counters (problem signalled by Frank MITTELBACH).

- 2002/02/18
  - Corrected a spacing problem with empty titles (problem signalled by Frank MITTELBACH).
  - Workaround for the `\parttoc-\chapter*` problem.

- 2002/02/19
  - Added `\mtcskip` and `\mtcskipamount`.

- 2002/02/27
  - Fixed test for empty files.

- 2002/03/13
  - Added the bangla language.

- 2002/03/15
  - Reduced depth of `\mtc@strutbox`.

* version 37

- 2003/01/24
  - Version #37 dropped.

* version 38

- 2003/01/24
  - `pt` becomes `\@pt` and `\pt` becomes `\z@`.
  - `\hrule` and `\vrule` replaced by `\rule` (LATEX).
  - Added `\mtc@zrule` for zero-dims rules.
  - Added the frenchb language (synonym of french).

- 2003/01/30
  - Changed the test for empty titles.
  - Added the `flsection` and `flsectionb` package options.

- 2003/01/31
  - The `tight` and `loose` package options are applied to `\parttoc` (Thomas LEONHARDT).

- 2003/02/07
  - Package options `flsection` and `flsectionb` removed and replaced by the `insection` package option (like `flsectionb`).

- 2003/02/11
  - Corrected numbering of SLF, SLT.

- 2003/02/20
  - Added `frenchle` and `frenchpro` language options (synonyms of `french`).
Changes history

- Corrected sectocs, at least.

- 2003/03/18
  - Corrected some vertical spacings and struts (I added some mods by Frank Mittlebach, many thanks to him.). A lot of cleaning remains to do, but the release seems to be needed now.

★ version 39

- 2003/04/09
  - Modern font commands for compatibility with the memoir class.
  - \nomtcpagenumbers and memoir class.

- 2003/06/08
  - Added \@filesfalse and \mtc@hook@beforeinputfile for the notoccite package (requested by Donald Arsenseau); added the notoccite package option (loads the notoccite package).

- 2004/09/08
  - Added language options and .mld files for dialects: canadian (english), acadian, acadien, canadien (french), naustrian, ngerman (ngermanb).
  - Added comments in .mld files using special fonts.
  - Documentation: added a paragraph about making a TOC for appendices, eventually not listed in the main TOC.

- 2004/09/17
  - Corrections in the documentation; corrections about rules.

★ version 40

- 2004/12/09
  - Added the japanese and castillian languages.
  - Removed the test on the presence of the multicol package in minitoc.tex, because multicol is a required package.
  - Added a figure in minitoc.tex about the need of three compilations.
  - Added some infos in minitoc.bug.
  - Added a paragraph about a problem with the appendix package.

- 2004/12/13
  - Updated minitoc-fr.bib and minitoc.bib.

- 2004/12/14
  - Added the hints package option. This option is still experimental; your advice is welcome.

- 2004/12/20
  - Added minitoc-fr.pdf (french documentation in PDF format).
Changes history

∗ version 41

• 2005/01/05
  ◦ Corrections in documentation.
  ◦ Message added if some sectioning commands are not available.
  ◦ Replaced \typeout commands in minitoc by the \PackageInfo or \PackageWarning commands; with the line number when useful (\@gobble if no line number). Hence, the package is less verbose (\PackageInfo writes only in the document.log file, not on the terminal).

• 2005/01/06
  ◦ Added the \mtcsetfont (Benjamin Bavary) and \mtcsettitlefont commands, with a much simpler syntax.

• 2005/01/10
  ◦ Added bibliography.

• 2005/01/11
  ◦ AMS classes: amsart and amsproc are incompatible with minitoc, amsbook needs precautions.

• 2005/01/12
  ◦ Added \mtcsetformat.

• 2005/01/18
  ◦ Added \mtcsettitle.
  ◦ Added a hint for recommending the insection package option.

• 2005/01/19
  ◦ Added a hint about the presence of \dominitoc and co.
  ◦ Added a hint about consistency of \dominitoc/minitoc and co.
  ◦ Improved documentation about hints.

• 2005/01/20
  ◦ Added a hint about using short extensions with more that 99 parts or 99 chapters or 99 sections.

• 2005/01/25
  ◦ \ptifont: \Huge\bfseries becomes \LARGE\bfseries.

• 2005/01/26
  ◦ Added \mtcsetpagenumbers.

• 2005/01/28

• 2005/02/02
  ◦ Added \mtcsetrules.

• 2005/02/03

Changes history

○ Added \plfrule, \noplfrule, \mlfrule, \nomlfrule, \slfrule, \noslfrule, \pltrule, \nopltrule, \mltrule, \nomltrule, \sltrule, \osltrule.

• 2005/02/04
  ○ Added the \mtchideinmaintoc environment.

• 2005/02/08
  ○ Added latvian.mld, letton.mld, greek-mono.mld, greek-polydemo.mld, greek-polykatha.mld, polish2.mld, russian2m.mld, and russian2o.mld as new language files.

• 2005/02/09
  ○ Added the \mtchideinmainlof and \mtchideinmainlot environments.

• 2005/02/10
  ○ Added tests on the \mtchideinmain\* environments.

• 2005/02/14
  ○ Added \mtcfixindex.

※ version 42

• 2005/02/14
  ○ Version 42 not released.
  ○ Replaced “language” by “langue” in the french documentation.

• 2005/02/15
  ○ Fixed a minor typo.

• 2005/02/16
  ○ Upgraded \mtcfixindex.

• 2005/02/21
  ○ Added \mtcsettitle, forgotten to be inserted in v41.

※ version 43

• 2005/02/21
  ○ Version 43: consolidation of v40, v41 and v42.

• 2005/02/24
  ○ Fixed a bug in \mtcsetformat.
  ○ Fixed a bug in mtcoff.sty about \mtcfixindex.

• 2005/03/02
  ○ Fixed the \mtcset... macros.
  ○ Moved history to the end of package code.
  ○ Added the INSTALL file and a chapter about installation.

• 2005/03/07
  ○ Fixed a typo (Benjamin Bayart).
  ○ Completed the hint about consistency of \dominitoc/\minitoc and co.
Changes history

• 2005/03/08
  ○ Added a hint about consistency of \minitoc and \tableofcontents.
• 2005/03/09
  ○ Added comments about fonts.
• 2005/03/10
  ○ Corrections in documentation.
• 2005/03/11
  ○ Added \mtcsetfeature.
• 2005/03/14
  ○ Added bulgarianb.mld (upper bulgarian).
• 2005/03/15
  ○ Added *[-\baselineskip] after the \ after the top rule of each part level mini-table.
• 2005/03/16
  ○ Corrections in the arguments of \mtcsetfeature.
• 2005/03/18
  ○ Removed \markboth for minitocs (...) and sectocs (...).
• 2005/03/21
  ○ Added spanish2.mld.
• 2005/03/22
  ○ Added a hint for the abstract package.
• 2005/04/07
  ○ Corrected the stc@verse environment.
  ○ Added finnish2.mld, latin2.mld, and magyar2.mld.
• 2005/04/08
  ○ Renamed portuges.mld as portuges.mld.
• 2005/04/12
  ○ Correction in \mtcskip.
  ○ First version in \dtex format.
• 2005/04/14
  ○ Removed \ypart, \ychapter, \ysection, and stuff; unused.
• 2005/05/11
  ○ Corrected a typo in \@dosectlot.
  ○ Added \mtcfixglossary.
  ○ Print the documentation with “oneside” to have all marginal notes on left.
  ○ Added the (extended to 54 floats) code of morefloats (Don Hosek) to allow more marginpars and floats.
  ○ Added minitoc.ist to format the index correctly.
• 2005/05/26
Changes history

- Fixed rules in parttocs, partlofs and partlots.
- 2005/05/30
  - Fixed chapter-level entries in parttocs, when page numbers must be removed.
  - Added a hint about the sectsty package (must be loaded before minitoc).
- 2005/06/01
  - Added a hint about attempts to insert empty mini-tables.
  - Added a hint about the use of obsolete commands.
  - The mini-lists of figures or tables should not be printed empty even if tocdepth < 1.
- 2005/06/02
  - Added the notion of depth for mini-tables of figures/tables.
  - Added \mtcsetdepth.
  - The hints option is the default and no more considered as experimental.
- 2005/06/03
  - Added an error message in \mtcsetdepth if the counter is not available.
- 2005/06/06
  - Added portuges.mld, which loads portugues.mld.
- 2005/06/07
  - Added three variants for the malayalam language: malayalam-keli.mld, malayalam-rachana.mld, and malayalam-rachana2.mld.
- 2005/06/14
  - Added method for bilingual documentation.
- 2005/06/15
  - Added minitoc-fr.ist to format correctly the index in french.
- 2005/06/16
  - Changed “Liste des Tables” by “Liste des Tableaux” in french.mld, and in the french documentation, to stick to the choices of the babel package.
- 2005/06/17
  - The file minitoc-fr.dtx is now generated by minitoc.ins.
- 2005/06/21
  - Added “OUI”, “NON”, “oui”, “non”, “O”, and “o” as true/false keywords.
  - Compacted the code about detection of short/long extensions.
- 2005/06/22
- 2005/06/23
  - Correctly set the \ifFTR flag to have the names of months in the right language in the bibliography.
- 2005/06/29
Changes history

- Set the flag `\mtcoffwarn@true` in `mtcoff` if a command `\mtcadd...` is found.

- 2005/07/01
  - Added `castillian.mld`.
  - Renamed `portugues.mld` as `portuguese.mld`.

- 2005/07/11
  - Added `brazilian.mld`, `british.mld`, `UKenglish.mld`, and `USenglish.mld`.

- 2005/07/12
  - Suppressed “General:” in the changes history.

- 2005/07/13
  - Replaced some `\PackageWarning` commands by `\PackageInfo`.

- 2005/07/18
  - Restoring the correspondence of each language option with a `.mld` file.

- 2005/07/20
  - Improving the `mtchideinmainlof` and `mtchideinmainlot` environments.

- 2005/07/21
  - Removing unused some flags `\if@mtc@setpagenumbers@act@` and `\if@mtc@setrules@act@`.
  - Added the `\decrementptc`, `\decrementmtc`, and `\decrementstc` commands.

- 2005/07/22
  - Corrected a bug in `mtcoff`.
  - Improved some messages in `mtcoff`.
  - Added a test on the version of the `placeins` package.

- 2005/08/23
  - Added a note about `\FloatBarrier`.

- 2005/08/24
  - Added a note about an alignment problem in the `minitoc`. Updated `minitoc.bug`.
  - Made two versions of the `mtchideinmainlof` and `mtchideinmainlot` environments, depending on the presence of the corresponding depth counter.
  - The `memoir` class is incompatible if too recent.

- 2005/08/25
  - Added a comment about the position of the `\do...` preparation commands.
  - Corrections in the `mtchideinmainlof` and `mtchideinmainlot` environments.

- 2005/08/26
  - Added `guarani.mld`.

- 2005/08/29
  - Added `\incrementptc`, `\incrementmtc`, and `\incrementstc`.
Changes history

◦ Added an optional argument to \adjustptc, \adjustmtc, and \adjuststc.
◦ Added the k-tight and k-loose package options.

• 2005/09/02
◦ Added a patch for the recent version of the memoir class.

• 2005/09/06
◦ Added spanish3.mld.

• 2005/09/08
◦ Use \mtcselectlanguage in language options and in “secondary” .mld files.

• 2005/09/09
◦ Added \mtcloadmlo to be used in some .mld files to load a .mlo file.

• 2005/09/12
◦ Added a test to forbid direct calls of \mtcloadmlo by the user.

• 2005/09/13
◦ Added farsi1.mld, farsi1.mlo, farsi2.mld, and farsi2.mlo.
◦ Added a note about the rubber tool.

• 2005/09/15
◦ Added mtcglo.ist to format the glossary.

• 2005/09/16
◦ Removed the page numbers in the glossary. Done in the *mk scripts.

★ version 44

• 2005/09/26
◦ Changes history (glossary) typeset in RaggedRight.

• 2005/09/27
◦ Added germanb2.mld, ngermanb2.mld, norsk2.mld, and nynorsk2.mld.

• 2005/09/28
◦ New method for history: embedded lists on 3 levels.
◦ Removed mtcglo.ist.

• 2005/09/29
◦ Cleaned the *mk scripts.
◦ Added the listfiles package option.

• 2005/09/30
◦ Corrected typos.
◦ Added the name of the .maf file in the message of the listfiles package option.
◦ Improved the cleaning in the *mk scripts, using a .maf file.

• 2005/10/03
◦ Load the patch for the memoir class only if necessary; do not load it if memoir is dated after 2005/09/25.
Changes history

- Added a remark in the FAQ chapter (and minitoc.bug) about precautions to take with the starred sectioning commands.
  - 2005/10/04
    - Added the nolistfiles package option.
    - Added a hint about the caption, caption2, ccaption, and mcaption packages (they must be loaded before minitoc).
  - 2005/10/05
    - Fixed typos in the documentation.
    - Fixed some marginal notes in the commented code.
  - 2005/10/06
    - Minor corrections in the documentation.
    - Use the xargs Unix command in the *mk scripts to remove the auxiliary files.
  - 2005/10/07
    - Minor corrections in the documentation.
    - Added a short intro to the “Frequently Asked Questions” chapter and to minitoc.bug.
  - 2005/11/02
    - Minor corrections in the documentation.
  - 2005/11/04
    - Minor corrections in the documentation.
  - 2005/11/07
    - Begin adding the “Jargon” chapter.
  - 2005/11/08
    - Added the french L\TeX Companion [332].
  - 2005/11/09
    - Continuing the “Jargon” chapter.
    - Adding minitoc.pre in minitoc.l.
    - Adding a note about the need of running imk before emk or fmk.
  - 2005/11/10
    - Fixed typos in the documentation.
    - Added a note about a problem with minitoc, hyperref and memoir.
    - Continuing the “Jargon” chapter.
  - 2005/11/14
    - Fixed typos in the documentation.
    - Continuing the “Jargon” chapter.
    - Improve the notes about the memoir class.
  - 2005/11/15
    - Continuing the “Jargon” chapter.
    - Improve the notes about the memoir class.
    - Added \plfsfont, \pltbsfont, \mlfsfont, \mlltbsfont, \slfsfont, \sltbsfont for subfigures and subtables entries in the mini-tables.
Changes history

- 2005/11/16
  - Continuing the “Jargon” chapter.
  - Fixed a bug about fonts for subfigures and subtables entries in the mini-
    tables.
  - Added bicig2.mld, bithe.mld, manju.mld, xalx.mld, and khalkha.mld.
- 2005/11/17
  - Continuing the “Jargon” chapter.
  - Added testing via internal \texttt{quarks} commands in \texttt{mtcsetfont}.
- 2005/11/18
  - Continuing the “Jargon” chapter.
  - Fixed typos in the documentation.
- 2005/11/21
  - Continuing the “Jargon” chapter.
- 2005/11/22
  - Continuing the “Jargon” chapter.
- 2005/11/23
  - Continuing the “Jargon” chapter.
  - Updating the bibliography.
- 2005/11/24
  - Continuing the “Jargon” chapter.
- 2005/11/25
  - Continuing the “Jargon” chapter.
  - Changed “table” into “tableau” in the french doc, where necessary.
  - Updating the bibliography.
- 2005/11/28
  - Continuing the “Jargon” chapter.
  - Fixed typos in the documentation.
- 2005/11/29
  - Continuing the “Jargon” chapter.
  - Fixed typos in the documentation.
  - Fixed typos in the bibliography.
  - Updating the bibliography.
- 2005/11/30
  - Continuing the “Jargon” chapter.
  - Avoid some warnings “Token not allowed” from pdftex.
- 2005/12/01
  - Continuing the “Jargon” chapter.
- 2005/12/02
  - Continuing the “Jargon” chapter.
  - Reordering a long sequence of citations.
Changes history

- Added “mailto:” in the mailing URLs.

2005/12/05
- Fixed typos in the documentation.
- Continuing the “Jargon” chapter.
- Added a hint about the \texttt{varsects} package (must be loaded \texttt{before} \texttt{minitoc}).

2005/12/06
- Continuing the “Jargon” chapter.
- Correcting an hyperlink in the bibliography (for the \texttt{xt} package).
- Attempting to avoid broken URLs, using \texttt{quote}, footnotes and \texttt{\par}.

2005/12/07
- Continuing the “Jargon” chapter.
- Updating the bibliography.
- Corrections of layout (some headers, a table).
- In the warning message of the hint about a number of mini-tables greater than 99 (if short extensions), give the effective number.
- Reduce the width of some info, warning or error messages.

2005/12/08
- Corrections of layout (some headers).

2005/12/09
- Corrections of french quotes.
- Added some PDF options.
- Continuing the “Jargon” chapter.
- Corrected an URL to the \texttt{AMS} in the bibliography.

2005/12/19
- Made some messages shorter (mainly by removing stars).

2005/12/21
- Correction of typos.
- Added some labels.
- Added a chapter with the (explained) messages. Not yet sorted.
- The documentation needs 4 \LaTeX{} runs.

2005/12/22
- Made some messages shorter.
- Corrections in the list of messages.
- Updating the bibliography.

2005/12/23
- Improving the placement of floats on pages of floats: to the top.

2006/01/03
- Corrections in the documentation (thanks to Markus Gleiszner).
- Added \texttt{addsec.tex}

2006/01/04
Changes history

- Corrected the flag `\ifundottedmtc`.
- Correction to make `addsec.tex` work.

- **2006/01/05**
  - Added "*" as keyword for the first argument of `\mtcsetpagenumbers` and `\mtcsetrules` (asked by Markus Gleiszner).
  - Removed "\MessageBreak" from the index.

- **2006/01/06**
  - Continuing the “Jargon” chapter.
  - Corrected the bibliography entry about BangTEX.
  - Updated the bibliography.
  - Used the `afterpage` package [115] in the documentation to fix a float positioning problem.

- **2006/01/09**
  - Corrections in the documentation.
  - Fixing a float positioning problem.

- **2006/01/10**
  - Corrections in the documentation.
  - Continuing the “Jargon” chapter.
  - Updated the bibliography.
  - Added the bahasam language.
  - Added the albanian language.
  - Added the hebrew2 language.

- **2006/01/11**
  - Updated the bibliography.
  - Updated the documentation for the albanian, bahasa, bahasam, and hebrew2 languages.
  - Updated `french.mld` (removing abusive uppercase letters).
  - Corrected the `italian.mld` file. Added the italian2 language.
  - Added the australian and newzealand languages (english).
  - Renamed the bahasa language as bahasai; bahasa is synonym of bahasai.
  - Added the malay and meyalu languages, synonyms of bahasam.
  - Added the indon and indonesian languages, synonyms of bahasai.

- **2006/01/12**
  - Updated the bibliography.
  - Updated the acknowledgements.
  - Added references to the new bibliographic entries.

- **2006/01/13**
  - Fixed an instability in page breaks in the documentation of `japanese3.mld`.
  - Added comments in some `.mld` files.
  - Added `magyar3.mld`.
  - Updated `lithuanian.mld`. 
Changes history

- 2006/01/16
  - Correction in \mtcaddsection.

- 2006/01/17
  - Correction in \mtcfindex and \mtcfindexglossary.
  - Updated the bibliography.
  - Limitation of the initial depth of displayed bookmarks.

- 2006/01/18
  - Added some comments in point 34 of the FAQ (and in minitoc.bug) about the initialization of fonts.
  - Added romanian2.mld and romanian3.mld.
  - Updated the bibliography.

- 2006/01/19
  - Updated the bibliography.
  - Load some packages before hyperref.
  - Added spanish4.mld.

- 2006/01/23
  - Corrected the table about default titles.
  - Corrected the keywords for \mtcsetfont.
  - Added lowersorbian.mld, uppersorbian.mld, and ukrainian.mld.

- 2006/01/24
  - Updated documentation for lowersorbian.mld, uppersorbian.mld, and ukrainian.mld.

- 2006/01/25
  - Corrections in the documentation.
  - Updated the bibliography.

- 2006/01/26
  - Added a hint about the KOMA-Script classes [343, 344, 399], and an entry in the FAQ chapter (and in minitoc.bug).

- 2006/01/27
  - Updated the bibliography.
  - Added a note in documentation of serbian.mld and serbianc.mld.

- 2006/01/30
  - Added ethiopian2.mld (for Omega).

- 2006/01/31
  - Simplifications in the “Messages” chapter.
  - Corrections in the “Jargon” chapter.

- 2006/02/01
  - Corrections in the documentation.
  - Added the “Postface” chapter.

- 2006/02/02
Changes history

- Corrections in the “Postface” chapter.
- Updated the bibliography.
- 2006/02/06
  - Corrections in the documentation.
  - Updated the bibliography.
  - Added package dblaccent [328] for the “The pdfTeX Program” entry in the bibliography. Its author’s first name needs a double accent (Thế Thành Hân); je l’ai aussi utilisé pour composer d’autres mots vietnamiens.
- 2006/02/07
  - Corrections in the documentation.
  - Updated the bibliography.
- 2006/02/09
  - Corrections in the documentation.
- 2006/02/10
  - Corrections in the documentation.
  - Updated the bibliography.
- 2006/02/13
  - Added malayalam-omega.mld and malayalam-omega.mlo.
  - Updated the bibliography.
- 2006/02/14
  - Added kannada.mld.
  - Updated the bibliography.
- 2006/02/15
  - Corrections in russianb.mld and spanish.mld.
  - Corrections in the documentation and the bibliography.
  - Place `\mtcfixglossary` before `\mtcfixindex`.
- 2006/02/16
  - Added a citation from Donald Arseneau.
  - Updated the bibliography.
  - Updated the acknowledgments.
- 2006/02/17
  - Updated the bibliography.
  - Updated the jargon.
- 2006/02/20
  - Added u8hangul.mld, u8hangul.mlo, u8hanja.mld, and u8hanja.mlo.
- 2006/02/21
  - Renamed languages u8hangul and u8hanja into hangul-u8.ml[d|o] and hanja-u8.ml[d|o].
  - Updated the bibliography.
- 2006/02/22
Changes history

- Added a hint about repeated preparation commands.
- Moved up the declaration of some flags relative to the hints option.
- Added `\mtcprepare`.

- 2006/02/23
  - Updated the bibliography.

- 2006/02/24
  - Updated the bibliography.

- 2006/02/27
  - Corrections in the documentation.
  - Added `minitoc.pre` to class 6.

- 2006/02/28
  - Corrections in the documentation.
  - Corrected `irish.mld`, `lsorbian.mld` and `usorbian.mld`.
  - Added `polski.mld`.

- 2006/03/01
  - Hints about the `jura` class and the `alphanum` package, incompatible with `minitoc`.

- 2006/03/02
  - Use bibliographic styles with an URL field, built with the help of `urlbst` [196].

- 2006/03/06
  - Update the bibliography.

- 2006/03/08
  - Corrections in `magyar.mld`, `magyar2.mld`, and `magyar3.mld`.
  - Added `russian-cca.mld`, `russian-cca1.mld`, and `russian-1h.mld`, with their `.mlo` files.

- 2006/03/09
  - Update the bibliography.

- 2006/03/10
  - Update the bibliography.
  - Added `russian-lhcyrlalt.mld`, `russian-lhcyrkoi.mld`, and `russian-lhcyrwin.mld`, with their `.mlo` files.

- 2006/03/13
  - Corrections in the documentation.

- 2006/03/14
  - Added the `mtcmess` package.

- 2006/03/16
  - The messages are now numbered.
  - Update the bibliography.

- 2006/03/20
Changes history

- Corrections in the documentation.
  - 2006/03/21
    - Update the bibliography.
  - 2006/03/22
    - Update the jargon.
  - 2006/03/28
    - Corrections in the documentation.
    - Update the jargon.
  - 2006/03/29
    - Added FAQ 37 about `.mld` files and `babel`.
    - Added `french1.mld` and `french2.mld`.
    - Update the jargon.
  - 2006/03/30
    - Added `english1.mld` and `english2.mld`.

★ version 45

- 2006/03/31
  - Suppression of the PostScript versions of the documentation.
  - Added `arab2.mld`, `bicig3.mld`, `buryat2.mld`, `xalx2.mld`, and `xalx3.mld`.
- 2006/04/03
  - Corrections in the documentation.
- 2006/04/04
  - Added `swedish2.mld`.
  - The `insection` package option loads also the `flafter` package.
- 2006/04/05
  - Corrections in the documentation.
  - Added `lamed.eps` and `lamed.pdf` as images for the `Lamed` logo (built from `lamed.tex`).
  - Reordering of the chapters in the user’s manual (part I).
- 2006/04/06
  - Use `sectsty` to better format section titles.
- 2006/04/07
  - Corrections in the documentation.
  - Added `\ifmctsecondpart` to check if the document has exactly 2 parts.
- 2006/04/10
  - Corrections in the documentation.
- 2006/04/11
  - Corrections in the documentation.
- 2006/04/12
Changes history

- Corrections in the documentation.

  - 2006/04/13
    - In the \texttt{insection} package option, load the \texttt{flafter} package \textit{before} the \texttt{placeins} package.
    - Added a figure about the float barriers.

- 2006/04/27
  - Added notes in FAQ 20, about the use with the \texttt{appendix} package.
  - Added comments about the \texttt{insection} option.
  - Update the bibliography.
  - Begin correction of the \texttt{mtchideinmaintoc} environment.

- 2006/05/02
  - Added notes in \texttt{minitoc.bug}, point 20, about the use with the \texttt{appendix} package.
  - End correction of the \texttt{mtchideinmaintoc} environment.
  - Analogous corrections in the \texttt{mtchideinmainlof} and \texttt{mtchideinmainlot} environments.
  - Update the bibliography.

- 2006/05/03
  - Added notes about the \texttt{mtchideinmainlof} and \texttt{mtchideinmainlot} environments.
  - Added \texttt{hide1.tex} and \texttt{hide2.tex}.
  - Added a hint about the \texttt{fncychap} package (must be loaded \textit{before} \texttt{minitoc}).
  - Added a hint about the \texttt{quotchap} package (must be loaded \textit{before} \texttt{minitoc}).
  - Update the bibliography.

- 2006/05/04
  - Update the bibliography.
  - Added a hint about the \texttt{romannum} package (must be loaded \textit{before} \texttt{minitoc}).
  - Added a hint about the \texttt{sfheaders} package (must be loaded \textit{before} \texttt{minitoc}).
  - Added a hint about the \texttt{alnumsec} package (must be loaded \textit{before} \texttt{minitoc}).
  - Corrections in the documentation.

- 2006/05/05
  - Corrections in the documentation.

- 2006/05/24
  - Corrections in the documentation.
  - Renamed \texttt{hide.tex} to \texttt{hide1.tex}.
  - Update the bibliography.

- 2006/05/30
  - Corrections in the documentation.
  - Use \texttt{\MakeUpperCase} in \texttt{\markboth} for page styles.

- 2006/05/31
  - Update the bibliography.
Changes history

○ Added a hint about the captcont package (must be loaded before minitoc).
○ Corrections in the documentation.
○ Added uighur.mld, uighur2.mld, and uighur3.mld (as synonyms for the bicig variants).

• 2006/06/01
  ○ Corrections in the documentation.
  ○ Added description of MonTeX in the jargon.
  ○ Added a comment about the \ink script in INSTALL and the “Installation” chapter.
  ○ Added an entry about “package” in the jargon.
  ○ Added the \mtc-apx.tex example file.
  ○ Added FAQ 44 and the \mtcgapbeforeheads and \mtcgapafterheads commands.

• 2006/06/02
  ○ Corrections in the documentation.
  ○ Update the bibliography.
  ○ Added the \gaps.tex example file.

• 2006/06/05
  ○ Corrections in the documentation.
  ○ Update the bibliography.

• 2006/06/06
  ○ Corrections in the documentation.
  ○ Update the bibliography.

• 2006/06/08
  ○ Corrections in the documentation.
  ○ Spacing correction in french2.mld.

• version 46

• 2006/06/09
  ○ Corrections in the documentation and the bibliography.

• 2006/06/21
  ○ Update the bibliography.
  ○ Comment about the \thailatex package.

• 2006/06/22
  ○ Generate some example files with \minitoc.dtx/minitoc.ins.
  ○ Added chapter “Example files”.

• 2006/06/23
  ○ Renamed chapter “Example files” as “Examples of documents”.
  ○ Use the \lipsum package \cite{lipsum} in some of the examples of documents.
  ○ Update the bibliography.

• 2006/06/27
Changes history

○ Update the examples of documents.
○ Added the second.tex example file.

• 2006/06/29
○ Added the amem.tex, mem.tex and mem1.tex example files.

• 2006/06/30
○ Added the fo1.tex, fo2.tex and scr.tex example files.

• 2006/07/03
○ Added the subf.tex example file.
○ Corrections about the depth of minilofs, minilots and siblings.

• 2006/07/04
○ Added the tsfc.tex and tbi.tex example files.
○ Corrections in the bibliography.

• 2006/07/07
○ Corrections of typos.
○ Corrections in the bibliography.
○ Added the 2c.tex and mtc-bo.tex example files.
○ Correction in french2.mld.

• 2006/07/10
○ Correction in minilots and minilofs (and siblings) about depth.
○ Added the hop.tex and cri.tex example files.
○ Update the bibliography.

• 2006/07/11
○ Added the livre.tex, ch0.tex, tlc.tex and mu.tex example files.
○ Update the bibliography.

• 2006/07/12
○ Update the jargon.

• 2006/07/13
○ Corrections in the documentation.
○ The not released versions are flagged by ※ in place of ⋆.

• 2006/07/17
○ The “About this document” section becomes a starred first chapter.

• 2006/07/18
○ Added the hir.tex and hia.tex example files.

• 2006/07/19
○ Update the bibliography.
○ Corrections in add.tex and addsec.tex for the index.
○ Added the xmk script to typeset the examples into PDF documents.
○ Updated the scripts to treat the examples.

• 2006/07/20
○ Do not forget \jobname.mtc0 in the list of files.
○ In the scripts, the backup directory (OLD) is now /tmp/‘whoami’/OLD.
○ In the scripts, the repartition directories (CL[0–9]) are now /tmp/‘whoami’/CL[0–9].
Changes history

★ version 47

- 2006/07/26
  - Corrections in the documentation and the bibliography.
  - Update the bibliography.
- 2006/07/27
  - Added arabi.mld and farsi3.mld (from the Adbi system [243]).
  - Update the bibliography.
- 2006/07/28
  - Update the jargon.
- 2006/07/27
  - Fixed \l@xsection.
  - Fixed some spacings in mini-tables.
- 2006/08/01
  - Added a \kernafter... vertical kern between each minitable and its bottom rule.
  - Added point 45 of the FAQ.
- 2006/08/03
  - Minor correction in warning message F0008.
  - Update the bibliography.
  - Fixed a bug in romanian2.mld and romanian3.mld.
  - Shortened the result of some example documents by using the report class in place of the book class (hence using one side printing).

★ version 48

- 2006/08/04
  - Fixed typos.
- 2006/08/22
  - Update the bibliography.
  - No preamble in add.bib.
- 2006/08/23
  - Corrections in the TOC formatting.
  - Increasing \textwidth.
  - Correction of the preamble problem in add.bib and all generated files.
- 2006/08/24
  - Remove comments about spurious lines in preamble of generated files.
  - Added devanagari.mld and hindi.mld.
- 2006/08/25
  - Update the bibliography.
  - Added hindi-modern.mld.
  - Corrected the \name macro (for the documentation).
Changes history

- 2006/08/28
  - Corrections in the bibliography.
  - Correction (conversion) in hindi-modern.mld.

- 2006/08/29
  - Added error E0036 if english.mld is not found to set the default titles.

- 2006/08/31
  - Update the bibliography.
  - Modified the plainurl.bst to have family names of authors and editors in small caps and years in old style digits. Titles are in emphasis. The frplain1.bst style is also updated.

- 2006/09/01
  - Update the bibliography.
  - The bibliographic styles plainurl.bst and frplain1.bst are renamed en-mtc.bst and fr-mtc.bst.

- 2006/09/05
  - Update the bibliography.
  - Renamed add.bib to mtc-add.bib.
  - Renamed add.tex to mtc-add.tex.
  - Renamed addsec.tex to mtc-ads.tex.
  - Renamed 2c.tex to mtc-2c.tex.
  - The listfiles package option is now active by default.

- 2006/09/07
  - Renamed app-mem.tex to mtc-amm.tex.
  - Renamed apx.tex to mtc-apx.tex.
  - Renamed bo.tex to mtc-bo.tex.
  - Renamed ch0.tex to mtc-ch0.tex.
  - Renamed cri.tex to mtc-cri.tex.
  - Renamed fol1.tex to mtc-fol1.tex.
  - Renamed fo2.tex to mtc-fo2.tex.
  - Renamed gaps.tex to mtc-gap.tex.
  - Renamed hia.tex to mtc-hia.tex.
  - Renamed hir.tex to mtc-hir.tex.
  - Renamed hide1.tex to mtc-hil1.tex.
  - Renamed hide2.tex to mtc-hi2.tex.
  - Renamed hop.tex to mtc-hop.tex.
  - Renamed livre.tex to mtc-liv.tex.
  - Renamed mem.tex to mtc-mem.tex.
  - Renamed mem1.tex to mtc-mml.tex.
  - Renamed mini-art.tex to mtc-art.tex.
  - Renamed minitoc-ex.tex to mtc-bk.tex.
  - Renamed mu.tex to mtc-mu.tex.
  - Renamed scr.tex to mtc-scr.tex.
Changes history

- Renamed second.tex to mtc-2nd.tex.
- Renamed subf.tex to mtc-sbf.tex.
- Renamed tbi.tex to mtc-tbi.tex.
- Renamed tlc.tex to mtc-tlc.tex.
- Renamed tsfc.tex to mtc-tsf.tex.

- **2006/09/08**
  - Updated the bibliography (added the Pentaglot).
  - Corrected the format of two tables about NFSS.
  - Example documents in alphabetical order in their chapter.

- **2006/09/11**
  - Updated the bibliography.

- **2006/09/12**
  - Added a figure about systems derived from \TeX{} and \LaTeX{}.

- **2006/09/13**
  - Added the mtc-syn.tex example document file.

*version 49*

- **2006/09/14**
  - Slightly modified the layout of the list of files ("Installation" chapter).
  - Simplifications in the scripts.
  - Updated the bibliography.

- **2006/09/18**
  - Updated the bibliography.
  - Added point 46 in the FAQ and example file mtc-tlo.tex.

- **2006/09/26**
  - Updated the bibliography.
  - Corrections in the bibliography and the bibliographic styles.

- **2006/09/29**
  - Better error messages about undefined preparation and insertion commands.
  - Updated the bibliography.
  - Added "+" and "-" as synonyms for "on" and "off", respectively.

- **2006/10/20**
  - Corrections in the bibliography.
  - Fixed typos.
  - Updated the bibliography.
  - Added a table of some encodings.

- **2006/10/31**
  - Suppressed the "Summary" entry in the summary, but added it in the Table of Contents.
  - Improving some tables.
  - Added the japanese6.mld and japanese6.mlo files.
Changes history

○ Updated the bibliography.

• 2006/11/03
  ○ Corrections in the bibliography.
  ○ Corrections in formatting a citation from Donald Arseneau.
  ○ Combine four figures in one (with sub-figures).
  ○ Added (in the memento) a table of the classes and packages which are incompatible or need precautions with minitoc.
  ○ Added a hint about the hangcaption package (must be loaded before minitoc).

• 2006/11/06
  ○ Completed the list of the standard classes.

• 2006/11/09
  ○ Added a validation of the language options with the presence of the .mld and .mlo files.
  ○ Added notes about the mandatory presence of the english.mld file.

• 2006/11/13
  ○ The validation of the language options writes only informative messages in the document.log file and, if necessary, gives only one warning message.

⋆ version 50

• 2006/11/17
  ○ Updated the bibliography.
  ○ Added the tmk script and a table describing a TDS structure for minitoc.
  ○ Added an item about the TDS in the jargon.
  ○ Updated the INSTALL file and the “Installation” chapter.

• 2006/11/29
  ○ Added the warning message W0094 with the list of the missing minitoc languages files (.mld and .mlo).
  ○ Corrections in the bibliography.
  ○ Updated the INSTALL file and the “Installation” chapter.
  ○ Changed the names of the scratch directories in some scripts.
  ○ Updated the bibliography.
  ○ Added the file minitoc.tds.zip (a ZIP archive of a TDS-compliant hierarchy of all files of the package) to the distribution.

⋆ version 51

• 2006/12/18
  ○ Improving the index: packages and classes, scripts, tools, names, examples, extensions, options, language options.
Changes history

• 2006/12/20
  ○ Improving the index: names.
  ○ Updated some .mld files with names of the authors of titles.

• 2007/01/09
  ○ Miscellaneous corrections.
  ○ The names of some internal macros are shortened to fit into the margin.
  ○ Added a `\ProvidesFile` command to the example files.
  ○ Indexing the environments (not perfect).
  ○ Indexing the files.
  ○ Renamed the file “catalog” into “CATALOG”.
  ○ Indexing the counters and depth counters.
  ○ The example files are in their own directory in the (proposed) TDS hierarchy.

* version 52

• 2007/01/11
  ○ Correction of index ordering.
  ○ Default option in boldface in the index.
  ○ Adding some informations about authors for language specific titles.
  ○ Updated the bibliography.

• 2007/01/12
  ○ Added fake sections in the “Examples of documents” chapter.
  ○ Removed the preparation of the documentation in PostScript format.

• 2007/01/15
  ○ Added the `cmk` script to convert the documentation from PDF format into PostScript format.
  ○ Removed `duplex2v.pro`.

• 2007/01/17
  ○ Indexing the referenced commands.

• 2007/01/18
  ○ Corrected the name Thé Thành Hän (first name before last name, the english way).
  ○ Corrected some other names.

• 2007/01/19
  ○ Added mongolb.mld and mongolb.mlo.
  ○ Removed mongolb.mlo (new cyrillic encodings T2 and X2 in mongolb.mld).
  ○ Added the example file `mtc-3co.tex`.
  ○ Trying to use a recent version of the `cite` [16] package (2003/11/04, 4.01) to allow sorting, but still clashes with `hyperref`.

• 2007/01/26
  ○ Balancing the columns in the index.

• 2007/01/29
Changes history

- Correction of the indexing of the environments.

- 2007/01/31
  - Improving the index layout.
  - Updated galician.mld.

- 2007/02/05
  - Added mongolian.mld which loads mongolb.mld.

- 2007/02/09
  - Indexing the names of authors.
  - Updated the bibliography.

- 2007/02/12
  - Updated the acknowledgements.

* version 53

- 2007/02/13
  - Added the example file mtc-fko.tex.
  - Corrected \kernafterminitoc and siblings.

- 2007/02/19
  - Updated the bibliography.
  - Bibliographic references for packages and classes in the index.

- 2007/03/02
  - Added a header to the index, to explain notations.
  - Updated the bibliography.
  - Changed the style of page and line numbers in the index.
  - Updated kannada.mld.
  - Dangerous bend symbols are now in the right margin.

- 2007/03/06
  - Improved the presentation of example files.
  - Using the natbib package [145, 146] to sort the sequences of citations.

- 2007/03/09
  - Correction in table 7.5 on page 247.
  - Renamed minitoc-texmf.zip into minitoc-tds.zip.

- 2007/03/19
  - Use the sort&compress option of the natbib package [145, 146] to compress the sequences of citations; the hypernat package must also be loaded (after natbib and hyperref).
  - Updated the bibliography.

- 2007/03/22
  - Added changing the title of the parttoc for appendices in mtc-apx.tex.
Changes history

⋆ version 54

- 2007/03/27
  - Added the “open” and “close” features.
  - Indexing the features.
  - Added the mtc-ocf.tex example file.

- 2007/04/06
  - Added the \mtcfixnomenclature command.
  - Added the mtc-nom.tex example file.
  - Updated the bibliography.
  - Corrected the last argument of \mtcsetfeature and siblings, using \mtc@toks.
  - Some mini-tables are set on two columns in the manual.
  - Indexing the messages. Messages noted in the right margin.
  - Corrected a bug in mtcoff.
  - Added latin1.mld and latin2.mld for classical latin.

- 2007/04/12
  - Added internal hyperlinks for messages.
  - Load the hypcap package for hyperlinks in the documentation.

- 2007/06/06
  - Added \mtcoffset and co. for an horizontal offset of a mini-table.
  - Added \mtcsetoffset for an horizontal offset of a mini-table type.
  - Added the mtc-ofs.tex example file.
  - Added flagging of macros in example files.
  - The 2007 section in the “Postface” chapter was garbled.
  - More internal links in the documentation.
  - Updated the bibliography.
  - Added a clickable table of all messages.
  - Improved column breaks in the index.
  - Added a local minitoc in the “Jargon” chapter.
  - Added lithuanian2.mld.
  - Added latvian2.mld and letton2.mld.
  - Grouped .mld/.mlo pairs in tables 7.1 to 7.2 on pages 243–244.

⋆ version 55

- 2007/06/12
  - Added a hint (warning W0997) about the flowfram [433, 434] package (incompatible).

- 2007/06/22
  - Regrouping some marginal notes about messages; improving their positions.
  - Improve page breaks in the documentation.
Changes history

○ Updated the bibliography.
○ Corrected a bug about minitocs in appendices for the \memoir class.

• 2007/06/29
  ○ Changed the color of hyperlinks.
  ○ Revised the format of the headers.
  ○ Corrected some \mtcset... commands to use \edef to correctly evaluate \mtc@toks.

★ version 56

• 2007/07/02
  ○ Added \swahili.mld.

• 2007/08/03
  ○ Page headers modified in documentation.
  ○ Added stuff (files) for figures (maps) for many language areas.
  ○ Removed the .eps files.
  ○ Added the bengali language synonym of bangla.
  ○ Split the list of files into two tables (tables 7.1 to 7.2 on pages 243–244).

• 2007/12/04
  ○ Many minor typo fixes.
  ○ Darker colors for hyperlinks.
  ○ Updated and corrected the bibliography.
  ○ Corrected a typo in the \ptc@verse environment (thanks to François Pétiard).
  ○ Corrections of typs in the \mtchideinmainlof and \mtchideinmainlot environments (thanks to Andrew Bowden).
  ○ Replaced the .mtc1 extension by .mtc0 in the auto-configuration test (to avoid erasing the \jobname.mtc1 file).
  ○ Corrected a problem with \nofiles (Andreas Deininger).
  ○ The acknowledgements are moved to the “Complements” part.
  ○ Added a hint (warning W0099) about the \titlesec [46] package.
  ○ Complete indexing of the messages.
  ○ Updated \lithuanian2.mld.
  ○ Using the \chngpage package [467] to increase the width of the pages of the bibliography.
  ○ Renamed \if@longextensions@ as \if@mtc@longext@.
  ○ Updated \czech.mld.
  ○ Removed \@xsection.
  ○ Graphic files are indexed separately.
  ○ Updated \galician.mld.
  ○ Added a specific directory for image files in the TDS hierarchy.
  ○ Updated \lsorbian.mld, \ukraineb.mld, and \usorbian.mld.
  ○ Added \malayalam-b.mld, \malayalam-keli2.mld, \malayalam-mr.mld, and \malayalam-rachana3.mld.
  ○ Updated \malayalam-omega.mlo.
Changes history

- Suppressed parasite entries from the index.

★ version 57

- 2007/12/11
  - Updated the bibliography.

- 2007/12/18
  - Corrections in examples of documents.
  - Added occitan.mld.
  - Updated croatian.mld, danish.mld, dutch.mld, galician.mld, germanb2.mld, greek.mld, icelandic.mld, interlingua.mld, polish.mld, scottish.mld, and turkish.mld.

- 2008/01/15
  - Corrected polski.mld.
  - Updated the bibliography.
  - Added table 6.10 on page 231.
  - Added maps of Manchuria.

- 2008/04/03
  - Better captions for maps. Added maps for Italy, Karnataka, Germany, Mongols and China.
  - Added occitan2.mld and mexican.mld.
  - Added a map of czech dialects.
  - Added maps of danish dialects.
  - Added a map of the dutch language.
  - Added a map of french dialects.
  - Added a map of galician dialects.
  - Added maps of german dialects.
  - Added maps of hindi dialects.
  - Added maps of portugese dialects.
  - Added a map of the turkish language.
  - Added a map of the vietnamese language.
  - Added a map of the armenian diaspora.
  - Added a map of the sami dialects.
  - Added a map of the nationalities in ex-Yugoslavia.
  - Added a map of countries where spanish is an official language.
  - Added a map of the sorbian area.
  - Added an entry for the Wikipedia in the jargon.
  - Added a map of the minorities in Poland.
  - Added a map of the bengali diaspora.
  - Splited the TDS hierarchy into three tables 7.3 to 7.5 on pages 244–247.
  - Added maps of the basque dialects.
  - Added maps of the latvian dialects.
  - Added a map for the swahili language.
  - Added the turkish alphabet.
Changes history

- The page numbers in the index are now hyperlinks (thanks to François Pétiard).
- Colors added in figure 1.1 on page 31.
- Added maps of the languages in Europe.
- Added maps of Kosovo.
- Added a map of the languages in Africa.
- Corrected an error of message number.
- Added maps for Russia.
- Added a map of the districts of Slovakia.
- Added maps about Islam.
- Added a figure about hànzì characters.
- Added a figure about Chinese characters usage in the world.
- Added a figure about Chinese dialects.
- Added maps about writing systems.
- Added a map of the regions where Finnish is spoken.
- Shortened the “Installation” chapter.
- Updated from the babel package version v3.8j of 2008/03/16.
- Files lamed.pdf and lamed.tex replaced by lamed3.png.
- Added maps of the indigenous languages of México.

**version 58**

- 2008/06/26
  - Renamed minitoc-tds.zip into minitoc.tds.zip.
  - Added a simplified linguistic map of Europe.
  - Added a map of Polish dialects.
  - Added a figure about the Russian alphabet.
  - Added a map about the Russian alphabet.
  - Added a figure about the Serbian alphabets.
  - Added a map of the provinces of Vietnam.
  - Used \vrefrange to compress ranges of internal cross-references.
  - Added a map of Albanian dialects.
  - Added a map of Norway.
  - Added flags for many countries. Added a light gray frame around the flags.
  - Added a figure about Lusophony.
  - Added a figure about Germanophony.
  - Added a figure about Hispanophony.
  - Added a figure about Italophony.
  - Added a minitoc in the index to make it easier to consult.
  - Added figures about francophones countries.
  - Added a figure about Swahili-speaking countries.
  - Added a figure about Arabic-speaking countries.
  - Added a figure about Russian-speaking countries.
  - Added a figure about English-speaking countries.
Changes history

- Added flags \ifinparttoc, \ifinpartlof, \ifinpartlot, \ifinminitoc, \ifinminilof, \ifinminilot, \ifinscottoc, \ifinscloatf, \ifinscloat, \ifinsecttoc, \ifinsectlof, and \ifinsectlot.
- Added example document mtc-vti.tex, section 4.36 on page 148.
- Added a figure about dutch-speaking countries.
- Renamed \fminitoc.dtx and consorts as \minitoc-fr.dtx and consorts.

※ version 59: corrupted PDF files.

* version 60

- 2008/07/16
  - Minor correction in figure.
  - Updated the bibliography.
  - Added missing flag files (thanks to Morten Hødahl).
  - Replaced many .pdf image files (most of them are flag files) by the original .png file because they were corrupted during the conversion by ImageMagick (xdvi didn't see the problem but Acrobat Reader refuses to show the file); many thanks to Heiko Oberdiek and Staszek Wawrzykiewicz.
  - Back to standard colors and default hyperref color options.

* version 61

- 2015/07/13
  - Jean-Pierre F. Drucbert passed away in 2009. So this package is now looking for a maintainer.
  - Reduce size of documentation, by eliminating flags and other images, from 25+mb to less than 2mb.
  - Remove the CATALOG file, as it was redundant and stale.
  - Use mirror.ctan.org for CTAN references.
  - Done by Nils Ole Tippenhauer (nils_tippenhauer at sutd.edu.sg) and Karl Berry (karl at freefriends.org). They are not prospective future maintainers.
  - No functional changes.

* version 62

- 2018/07/12
  - sources moved to github https://github.com/minitoc/minitoc
  - Correction to \@ifundefined usage that generates errors in current latex (and didn’t work previously)
  - Repository set up by David Carlisle but permanent maintainers still required
Acknowledgments

I ought to thank the following peoples\textsuperscript{53}, for their help, their questions, their interventions in the news groups\textsuperscript{54}, and/or for their packages, classes, documents, and tools:


\textsuperscript{53} And I apologize to all whose I forgot.

\textsuperscript{54} Mainly, \texttt{fr.comp.text.tex} (in French) and \texttt{comp.text.tex} (in English), but also \texttt{de.comp.text.tex} (in German, but I do not read it well: send me also a mail in French or in English).
Acknowledgements

Acknowledgements