

The minitoc package^{*}

Jean-Pierre F. Drucbert
<mailto:jean-pierre.drucbert@onera.fr>

3rd August 2006

^{*}This document corresponds to minitoc v47, dated 2006/00/01.

Summary

Table of Contents	4
List of Figures	16
List of Tables	17
About this document	19
I User's Manual	21
1 The minitoc package	23
2 Frequently Asked Questions	53
3 Memento	75
4 Examples of documents	83
5 Messages	137
6 Jargon	184
7 Installation	214
8 Postface	218
Acknowledgments	226
II Implementation	228
9 Commented code of minitoc.sty	230
10 Commented code of mtcoff.sty	395
11 Commented code of mtcmess.sty	409
12 Patch for the memoir class	410

<i>SUMMARY</i>	3
13 The language definition (.mld) and object (.mlo) files	412
III Complements	502
Bibliography	504
Changes History	522
Index	549

Table of Contents

Table of Contents	4
List of Figures	16
List of Tables	17
About this document	19
 I User's Manual	 21
1 The <code>minitoc</code> package	23
1.1 Introduction	24
1.1.1 Important restrictions	25
1.1.2 Version	25
1.2 License	25
1.3 How to use the <code>minitoc</code> package	26
1.3.1 Loading the package and creating the mini-tables	26
1.3.2 Preparing the mini-tables	26
1.3.3 Placing the mini-tables	28
1.3.4 Starred chapters, parts and sections	32
1.4 Typesetting of the mini-tables	33
1.4.1 Chapter-level mini-tables	33
1.4.2 Titles for chapter-level mini-tables	35
1.4.3 Part-level mini-tables	37
1.4.4 Titles for part-level mini-tables	37
1.4.5 Section-level mini-tables	37
1.4.6 Titles for section-level mini-tables	38
1.4.7 Position of the titles	38
1.4.7.1 For mini-tables at the part level	38
1.4.7.2 For mini-tables at the chapter level	38
1.4.7.3 For mini-tables at the section level	38
1.4.7.4 Summary of the positioning of titles	39
1.4.8 Line spacing in the mini-tables	39
1.4.9 Simplified commands for fonts	40
1.4.10 Simplified command for mini-table titles	40
1.4.11 Simplified command for mini-table depths	41
1.4.12 Languages for the titles	41

1.4.13	Altering the layout of the mini-tables	42
1.5	Special Features	44
1.5.1	Horizontal Rules	44
1.5.2	Page Numbers, Leaders	44
1.5.3	Features for parttocs and other mini-tables	45
1.5.3.1	Remark about page styles	46
1.5.4	The “Chapter 0” Problem (solved)	46
1.5.4.1	Before version #23 (1994/11/08)	47
1.5.4.2	Since version #23 (1994/11/08)	47
1.5.5	Special Entries for TOC, LOF, LOT, Bibliography and Index	47
1.6	The notoccite option	49
1.7	The listfiles and nolistfiles options	49
1.8	The hints option	49
1.9	Usage with MS-DOS	51
1.10	Why several L ^A T _E X runs are required ?	51
1.11	The mtcoff package	52
2	Frequently Asked Questions	53
2.0	Introduction	54
2.1	How to avoid a page break near the rules before and after the mini-table ?	54
2.2	How about implementing others layouts for a mini-table ?	55
2.3	Two consecutive backslashes in a contents line make an error . . .	55
2.4	Reordering chapters makes havoc	55
2.5	Extensions for the names of auxiliary files	56
2.6	Playing with the chapter number	56
2.7	Supported document classes	56
2.8	Compatibility with L ^A T _E X versions	56
2.9	Other mini-tables	57
2.10	Why so many auxiliary files ?	57
2.11	How to do mini-tables at levels other than chapter ?	58
2.12	Incompatibility with L ^A T _E X2.09	58
2.13	Documents resetting the chapter number at each part	59
2.14	The mini-tables have too much spaced lines	59
2.15	The secttocs are wrong	59
2.16	How to remove the lines of dots ?	59
2.17	How to use the hyperref package with minitoc ?	59
2.18	Problem while upgrading minitoc	60
2.19	A local table of contents for the set of appendices	60
2.20	Use with the appendix package	60
2.21	Use with the tocloft package	61
2.22	Use with the memoir class	62
2.23	There are too many commands for fonts, titles, and depths . . .	63
2.24	Compatibility with the A _M S document classes	63
2.25	How to hide some entries from the main table of contents . . .	63
2.26	How to define your own .mld file ?	67
2.27	Use with the abstract package	67
2.28	Use with the sectsty package	67
2.29	Strange alignment in the minitocs	67

2.30	Some useful precautions with starred sectionning commands	68
2.31	Use with packages for captions	69
2.32	Bad interaction minitoc/hyperref/memoir	69
2.33	Use with the varsects package	69
2.34	Initial font settings	69
2.35	Use with the KOMA-Script classes	71
2.36	Use with the jura class or the alphanum package	71
2.37	The .mld files and the babel package	72
2.38	Use with the fncychap package	72
2.39	Use with the quotchap package	72
2.40	Use with the romannum package	72
2.41	Use with the sfheaders package	72
2.42	Use with the alnumsec package	73
2.43	Use with the captcont package	73
2.44	Vertical space (gap) before and after parttocs, partlofs, and partlots titles	73
2.45	Vertical spacing before the bottom rule of a minitable	74
3	Memento	75
4	Examples of documents	83
4.1	The minitoc-ex.tex document file	84
4.2	The mini-art.tex document file	90
4.3	The add.tex document file	96
4.4	The addsec.tex document file	101
4.5	The second.tex document file	106
4.6	The app-mem.tex document file	108
4.7	The mem.tex document file	108
4.8	The mem1.tex document file	109
4.9	The fo1.tex document file	110
4.10	The fo2.tex document file	111
4.11	The scr.tex document file	113
4.12	The subf.tex document file	114
4.13	The tsfc.tex document file	116
4.14	The tbi.tex document file	118
4.15	The 2c.tex document file	119
4.16	The bo.tex document file	120
4.17	The hop.tex document file	125
4.18	The cri.tex document file	126
4.19	The livre.tex document file	127
4.20	The ch0.tex document file	131
4.21	The tlc.tex document file	132
4.22	The mu.tex document file	133
4.23	The hir.tex document file	134
4.24	The hia.tex document file	135
5	Messages	137
5.1	Messages from the minitoc package	138
5.1.1	Informative messages	138
5.1.1.1	Informative messages for hints	146

5.1.2	Warning messages	149
5.1.2.1	Warning messages for hints	153
5.1.3	Error messages	169
5.2	Messages from the <code>mtcoff</code> package	180
5.2.1	Warning messages	180
5.3	Message from the <code>mtcpatchmem</code> package	183
6	Jargon	184
7	Installation	214
8	Postface	218
8.1	The origins	219
8.2	New design in 1993	219
8.3	Developments in 1994	219
8.4	No developments in 1995	220
8.5	Developments en 1996	220
8.6	Developments in 1997	220
8.7	Developments in 1998	220
8.8	Developments in 1999	221
8.9	Developments in 2000	221
8.10	Developments in 2001	221
8.11	Developments in 2002	221
8.12	Developments in 2003	222
8.13	Developments in 2004	222
8.14	Developments in 2005	222
8.15	Developments in 2006	224
	Acknowledgments	226
II	Implementation	228
9	Commented code of <code>minitoc.sty</code>	230
9.1	Introduction	233
9.2	Identification code	234
9.3	A file descriptor to write	234
9.4	Indentation and skip	234
9.5	Tests and flags	235
9.5.1	Flags for the <code>hints</code> option	236
9.5.2	Use of section-level mini-lists of floats	236
9.5.3	Presence of some packages and classes	237
9.5.4	Presence or absence of some sectionning commands	238
9.5.5	Flags to check to see if some commands are used	239
9.5.6	Check if the document has exactly 2 parts	240
9.6	Preparation for the <code>notoccite</code> option	240
9.7	Preparation for the <code>tight</code> and <code>k-tight</code> options	241
9.8	Preparation to work with <code>hyperref</code>	241
9.9	Checking the presence of some packages	242
9.9.1	Check if the <code>sectsty</code> package is loaded, and when	242

9.9.2	Check if the <code>varsects</code> package is loaded, and when	242
9.9.3	Check if the <code>fncychap</code> package is loaded, and when	242
9.9.4	Check if the <code>quotchap</code> package is loaded, and when	242
9.9.5	Check if the <code>romannum</code> package is loaded, and when	243
9.9.6	Check if the <code>sfheaders</code> package is loaded, and when	243
9.9.7	Check if the <code>alnumsec</code> package is loaded, and when	243
9.9.8	Check if the <code>captcont</code> package is loaded, and when	243
9.9.9	Check if the <code>caption</code> package is loaded, and when	244
9.9.10	Check if the <code>caption2</code> package is loaded, and when	244
9.9.11	Check if the <code>ccaption</code> package is loaded, and when	244
9.9.12	Check if the <code>mcaption</code> package is loaded, and when	244
9.10	Is the <code>memoir</code> class loaded?	245
9.11	Testing the emptiness of a file	245
9.12	Internal macros to decrement <code>minitoc</code> counters	247
9.13	Patching the <code>\part</code> command	247
9.14	Adding an entry in the TOC for a starred part	248
9.15	Section level macros	248
9.16	Corrections for numbering	249
9.17	Patching the <code>\section</code> command	249
9.18	Adding an entry in the TOC for a starred section	249
9.19	Chapter level macros	250
9.20	Patching the <code>\chapter</code> command	250
9.21	Adding an entry in the TOC for a starred chapter	250
9.22	Miscellaneous declarations	251
9.23	Autoconfiguration of extensions	252
9.24	Detecting obsolete versions of \LaTeX	253
9.25	A macro to make a TOC entry without leaders nor page numbers	253
9.26	Default values for the page-number customization macros	254
9.27	“Features” for the mini-tables	256
9.28	Fake tables of contents	259
9.29	Depth counters for <code>minilofs</code> and <code>minilots</code>	259
9.30	Chapter level commands	260
9.31	Starred parts, chapters or sections	260
9.32	Font commands for the mini-tables	261
9.33	Internal commands to position the mini-table titles	262
9.34	The <code>mtc@verse</code> environment	263
9.35	The <code>\minitoc</code> , <code>\minilof</code> , and <code>\minilot</code> commands	263
9.35.1	The <code>\minitoc</code> command	263
9.35.2	The <code>\minilof</code> command	266
9.35.3	The <code>\minilot</code> command	269
9.36	Patching the <code>\chapter</code> command, continued	272
9.37	The <code>\addstarred...</code> commands	273
9.38	TOC entries without leaders	274
9.39	Mini-tables with or without leaders	274
9.40	The <code>\dominitoc</code> command and its siblings	275
9.40.1	Analysis and splitting of the TOC file	278
9.41	Mini-lists of figures	282
9.41.1	Analysis and splitting of the list of figures file	282

9.42	Mini-lists of tables	284
9.42.1	Analysis and splitting of the list of tables file	285
9.43	Macro to write a contents line	287
9.44	Depth counters for partlofs and partlots	288
9.45	Part level commands	289
9.46	Fonts for the parttocs	289
9.47	Default titles for part-level mini-tables	290
9.48	The <code>ptc@verse</code> environment	292
9.49	The part level mini-tables : <code>\parttoc</code> , <code>\partlof</code> , and <code>\partlot</code>	293
9.49.1	The <code>\parttoc</code> command	293
9.49.2	The <code>\partlof</code> command	296
9.49.3	The <code>\partlot</code> command	299
9.50	Auxiliary commands for printing parttocs	301
9.51	Patching the <code>\part</code> command, continued	302
9.52	The <code>\doparttoc</code> command and its siblings	302
9.52.1	Processing macros for the parttocs	305
9.52.2	Processing macros for the partlofs	308
9.52.3	Processing macros for the partlots	310
9.53	Depth counters for sectlofs and sectlots	313
9.54	Section-level commands	313
9.55	Fonts commands for secttocs and co.	314
9.56	Internal macros for title positioning	314
9.57	The <code>stc@verse</code> environment	315
9.58	The <code>\secttoc</code> , <code>\sectlof</code> , and <code>\sectlot</code> commands	315
9.58.1	The <code>\secttoc</code> command	315
9.58.2	The <code>\sectlof</code> command	318
9.58.3	The <code>\sectlot</code> command	321
9.59	Auxiliary internal commands, section level	323
9.60	Patching the <code>\section</code> command (continued)	324
9.61	The <code>\dosecttoc</code> command and siblings	325
9.62	End of section-level commands	334
9.63	The <code>\mtcprepare</code> command	335
9.64	Necessary <code>\l@...</code> commands	335
9.65	The horizontal rules and their default values	336
9.66	The <code>\mtcset...</code> commands	336
9.66.1	Keywords for the <code>\mtcset...</code> commands	337
9.66.2	The <code>\mtcsetfont</code> command	338
9.66.3	The <code>\mtcsettitlefont</code> command	342
9.66.4	The <code>\mtcsettitle</code> command	344
9.66.5	The <code>\mtcsetformat</code> command	345
9.66.6	The <code>\mtcsetpagenumbers</code> command	349
9.66.7	The <code>\mtcsetrules</code> command	351
9.66.8	The <code>\mtcsetfeature</code> command	353
9.66.9	The <code>\mtcsetdepth</code> command	356
9.67	The <code>mtchideinmaintoc</code> environment and siblings	357
9.68	Fixing the “Glossary” entry in the TOC	359
9.69	Fixing the “Index” entry in the TOC	361
9.70	The <code>\mtcselectlanguage</code> command	362

9.71	The <code>\mtcloadmlo</code> internal command	362
9.72	The “coffee breaks”	363
9.73	Initialization of counters	363
9.74	Declarations for simple options	363
9.74.1	Options <code>tight</code> and <code>loose</code> , <code>k-tight</code> and <code>k-loose</code>	364
9.74.2	Options <code>checkfiles</code> and <code>nocheckfiles</code>	364
9.74.3	Options <code>dotted</code> and <code>undotted</code>	364
9.74.4	Option <code>notoccite</code>	364
9.74.5	Option <code>shorttext</code>	364
9.75	The <code>insection</code> option	365
9.76	The <code>listfiles</code> and <code>nolistfiles</code> options	365
9.77	Language options	365
9.78	The <code>hints</code> option	369
9.78.1	First part : <code>\mtc@hints@begindoc</code>	369
9.78.1.1	Hint about the <code>alphanum</code> package	370
9.78.1.2	Hint about the <code>appendix</code> package	370
9.78.1.3	Hint about the <code>tocbibind</code> package	370
9.78.1.4	Hint about the KOMA-Script classes	370
9.78.1.5	Hint about the <code>tocloft</code> package	371
9.78.1.6	Hint about the <code>titletoc</code> package	371
9.78.1.7	Hint about the <code>placeins</code> package	371
9.78.1.8	Hint about the <code>memoir</code> class	373
9.78.1.9	Hint about the <code>amsart</code> and <code>amsproc</code> classes	373
9.78.1.10	Hint about the <code>amsbook</code> class	373
9.78.1.11	Hint about the <code>abstract</code> package	374
9.78.1.12	Hint about the <code>jura</code> class	374
9.78.1.13	Hint about the alteration of the sectionning commands	374
9.78.1.13.1	Alteration of <code>\part</code>	374
9.78.1.13.2	Alteration of <code>\chapter</code>	375
9.78.1.13.3	Alteration of <code>\section</code>	375
9.78.1.14	Hint about the coherence of the calling sequences of some commands	376
9.78.2	Final part : <code>\mtc@hints@enddoc</code>	377
9.78.2.1	Hint about <code>\sect-lof lot</code> and the <code>insection</code> option	377
9.78.2.2	Final part of the coherence tests	378
9.78.2.3	Check if the main tables have been prepared (first part)	380
9.78.2.4	Check if the main tables have been prepared (second part)	381
9.78.2.5	Check the number of mini-tables, in case of short extensions	383
9.78.2.6	Final part of the hint about the <code>sectsty</code> package	384
9.78.2.7	Final part of the hint about the <code>varsects</code> package	385
9.78.2.8	Final part of the hint about the <code>fncychap</code> package	385
9.78.2.9	Final part of the hint about the <code>quotchap</code> package	385
9.78.2.10	Final part of the hint about the <code>romannum</code> package	386
9.78.2.11	Final part of the hint about the <code>sfheaders</code> package	386
9.78.2.12	Final part of the hint about the <code>alnumsec</code> package	386

9.78.2.13	Final part of the hint about the <code>captcont</code> package . . .	387
9.78.2.14	Final part of the hint about the <code>caption</code> package . . .	387
9.78.2.15	Final part of the hint about the <code>caption2</code> package . . .	387
9.78.2.16	Final part of the hint about the <code>ccaption</code> package . . .	388
9.78.2.17	Final part of the hint about the <code>mcaption</code> package . . .	388
9.78.2.18	Check if empty mini-tables have been detected . . .	388
9.78.2.19	Check if obsolete commands have been used . . .	390
9.78.2.20	Check if some hints have been written . . .	391
9.79	Processing of options . . .	391
9.79.1	Processing the <code>insection</code> option . . .	391
9.79.2	Processing the <code>notoccite</code> option . . .	392
9.79.3	Processing the <code>listfiles</code> option . . .	392
9.79.4	Processing the <code>hints</code> option . . .	393
9.79.5	Saving the sectionning commands . . .	393
9.80	The <code>fminitoc.dtx</code> file . . .	394
10	Commented code of <code>mtcoeff.sty</code>	395
10.1	Why <code>mtcoeff.sty</code> ? . . .	395
10.2	Identification of the package . . .	396
10.3	Faking counters and dimensions . . .	396
10.4	Faking simple commands . . .	398
10.5	Faking commands with one optional argument . . .	399
10.6	Disabling the internal commands . . .	400
10.7	Disabling the font commands . . .	400
10.8	Disabling the <code>\mtcset . . .</code> commands . . .	402
10.9	Disabling the new <code>\l@ . . .</code> commands . . .	402
10.10	Ignore the obsolete commands . . .	402
10.11	Disabling the <code>\mtcselectlanguage</code> command . . .	402
10.12	Disabling the <code>\mtcloadmlo</code> command . . .	403
10.13	Disabling the commands for the horizontal rules . . .	403
10.14	Disabling the commands for the page numbers . . .	403
10.15	Disabling the mini-table features commands . . .	404
10.16	Disabling miscellaneous flags and commands . . .	405
10.17	Caution for some commands . . .	405
10.18	Disabling commands for “coffee” . . .	406
10.19	Disabling the <code>mtchideinmain . . .</code> environments . . .	407
10.20	Inhibition of the <code>\mtc@saveXXXdepth</code> and <code>\mtc@restoreXXXdepth</code> commands . . .	407
10.21	Disabling the <code>\mtcfixglossary</code> command . . .	407
10.22	Disabling the <code>\mtcfixindex</code> command . . .	407
10.23	Disabling the <code>\addstarred . . .</code> commands . . .	408
11	Commented code of <code>mtcmess.sty</code>	409
12	Patch for the <code>memoir</code> class	410
13	The language definition (<code>.mld</code>) and object (<code>.mlo</code>) files	412
13.1	“Acadian” language : <code>acadian.mld</code> . . .	416
13.2	“Acadien” language : <code>acadien.mld</code> . . .	417
13.3	“Afrikaan” language : <code>afrikaan.mld</code> . . .	417

13.4	“Afrikaans” language : afrikaans.mld	417
13.5	“Albanian” language : albanian.mld	418
13.6	“American” language : american.mld	418
13.7	“Arab” language : arab.mld	418
13.8	“Arab2” language : arab2.mld	419
13.9	“Arabi” language : arabi.mld	419
13.10	“Arabic” language : arabic.mld	420
13.11	“Armenian” language : armenian.mld	420
13.12	“Australian” language : australian.mld	421
13.13	“Austrian” language : austrian.mld	421
13.14	“Bahasa” language : bahasa.mld	421
13.15	“Bahasai” language : bahasai.mld	422
13.16	“Bahasam” language : bahasam.mld	422
13.17	“Bangla” language : bangla.mld	423
13.18	“Basque” language : basque.mld	423
13.19	“Bicig” language : bicig.mld	424
13.20	“Bicig2” language : bicig2.mld	424
13.21	“Bicig3” language : bicig3.mld	425
13.22	“Bithe” language : bithe.mld	426
13.23	“Brazil” language : brazil.mld	426
13.24	“Brazilian” language : brazilian.mld	427
13.25	“Breton” language : breton.mld	427
13.26	“British” language : british.mld	428
13.27	“Bulgarian” language : bulgarian.mld	428
13.28	“Bulgarianb” language : bulgarianb.mld	429
13.29	“Buryat” language : buryat.mld	429
13.30	“Buryat2” language : buryat2.mld	430
13.31	“Canadian” language : canadian.mld	430
13.32	“Canadien” language : canadien.mld	431
13.33	“Castillan” language : castillan.mld	431
13.34	“Castillian” language : castillian.mld	431
13.35	“Catalan” language : catalan.mld	431
13.36	“Chinese1” language : chinese1.mld and chinese1.mlo	432
13.37	“Chinese2” language : chinese2.mld and chinese2.mlo	432
13.38	“Croatian” language : croatian.mld	433
13.39	“Czech” language : czech.mld	433
13.40	“Danish” language : danish.mld	434
13.41	“Dutch” language : dutch.mld	434
13.42	“English” language : english.mld	435
13.43	“English1” language : english1.mld	436
13.44	“English2” language : english2.mld	436
13.45	“Esperant” language : esperant.mld	437
13.46	“Esperanto” language : esperanto.mld	437
13.47	“Estonian” language : estonian.mld	438
13.48	“Ethiopia” language : ethiopia.mld	438
13.49	“Ethiopian” language : ethiopian.mld	439
13.50	“Ethiopian2” language : ethiopian2.mld	439
13.51	“Farsi1” language : farsi1.mld and farsi1.mlo	440
13.52	“Farsi2” language : farsi2.mld and farsi2.mlo	440

13.53	“Farsi3” language : farsi3.mld	441
13.54	“Finnish” language : finnish.mld	441
13.55	“Finnish2” language : finnish2.mld	442
13.56	“Francais” language : francais.mld	442
13.57	“French” language : french.mld	443
13.58	“French1” language : french1.mld	443
13.59	“French2” language : french2.mld	444
13.60	“Frenchb” language : frenchb.mld	445
13.61	“Frenchle” language : frenchle.mld	445
13.62	“Frenchpro” language : frenchpro.mld	445
13.63	“Galician” language : galician.mld	445
13.64	“German” language : german.mld	446
13.65	“Germanb” language : germanb.mld	446
13.66	“Germanb2” language : germanb2.mld	447
13.67	“Greek” language : greek.mld	447
13.68	“Greek-mono” language : greek-mono.mld	448
13.69	“Greek-polydemo” language : greek-polydemo.mld	449
13.70	“Greek-polykatha” language : greek-polykatha.mld	450
13.71	“Guarani” language : guarani.mld	451
13.72	“Hangul1” language : hangul1.mld and hangul1.mlo	451
13.73	“Hangul2” language : hangul2.mld and hangul2.mlo	452
13.74	“Hangul3” language : hangul3.mld and hangul3.mlo	453
13.75	“Hangul4” language : hangul4.mld and hangul4.mlo	453
13.76	“Hangul-u8” language : hangul-u8.mld and hangul-u8.mlo	454
13.77	“Hanja1” language : hanja1.mld and hanja1.mlo	454
13.78	“Hanja2” language : hanja2.mld and hanja2.mlo	455
13.79	“Hanja-u8” language : hanja-u8.mld and hanja-u8.mlo	455
13.80	“Hebrew” language : hebrew.mld	456
13.81	“Hebrew2” language : hebrew2.mld	457
13.82	“Hungarian” language : hungarian.mld	457
13.83	“Icelandic” language : icelandic.mld	458
13.84	“Indon” language : indon.mld	458
13.85	“Indonesian” language : indonesian.mld	458
13.86	“Interlingua” language : interlingua.mld	459
13.87	“Irish” language : irish.mld	459
13.88	“Italian” language : italian.mld	460
13.89	“Italian2” language : italian2.mld	460
13.90	“Japanese” language : japanese.mld and japanese.mlo	461
13.91	“Japanese2” language : japanese2.mld and japanese2.mlo	461
13.92	“Japanese3” language : japanese3.mld and japanese3.mlo	462
13.93	“Japanese4” language : japanese4.mld and japanese4.mlo	462
13.94	“Japanese5” language : japanese5.mld and japanese.mlo	463
13.95	“Kannada” language : kannada.mld	463
13.96	“Khalka” language : khalkha.mld	464
13.97	“Latin” language : latin.mld	464
13.98	“Latin2” language : latin2.mld	465
13.99	“Latvian” language : latvian.mld	465
13.100	“Letton” language : letton.mld	466
13.101	“Lithuanian” language : lithuanian.mld	466

13.102	“Lowersorbian” language : lowersorbian.mld	467
13.103	“Lsorbian” language : lsorbian.mld	467
13.104	“Magyar” language : magyar.mld	468
13.105	“Magyar2” language : magyar2.mld	468
13.106	“Magyar3” language : magyar.mld	469
13.107	“Malay” language : malay.mld	469
13.108	“Malayalam-keli” language : malayalam-keli.mld	469
13.109	“Malayalam-omega” language : malayalam-omega.mld and malayalam-omega.mlo	470
13.110	“Malayalam-rachana” language : malayalam-rachana.mld .	470
13.111	“Malayalam-rachana2” language : malayalam-rachana2.mld	471
13.112	“Manju” language : manju.mld	472
13.113	“Meyalu” language : meyalu.mld	472
13.114	“Mongol” language : mongol.mld	472
13.115	“Naustrian” language : naustrian.mld	473
13.116	“Newzealand” language : newzealand.mld	473
13.117	“Ngerman” language : ngerman.mld	473
13.118	“Ngermanb” language : ngermanb.mld	473
13.119	“Ngermanb2” language : ngermanb2.mld	474
13.120	“Norsk” language : norsk.mld	475
13.121	“Norsk2” language : norsk2.mld	475
13.122	“Nynorsk” language : nynorsk.mld	476
13.123	“Nynorsk2” language : nynorsk2.mld	476
13.124	“Polish” language : polish.mld	477
13.125	“Polish2” language : polish2.mld	477
13.126	“Polski” language : polski.mld	478
13.127	“Portuges” language : portuges.mld	478
13.128	“Portuguese” language : portuguese.mld	479
13.129	“Romanian” language : romanian.mld	479
13.130	“Romanian2” language : romanian2.mld	480
13.131	“Romanian3” language : romanian3.mld	480
13.132	“Russian” language : russian.mld	481
13.133	“Russian2m” language : russian2m.mld	481
13.134	“Russian2o” language : russian2o.mld	482
13.135	“Russianb” language : russianb.mld	483
13.136	“Russianc” language : russianc.mld	484
13.137	“Russian-cca” language : russian-cca.mld and russian-cca.mlo	485
13.138	“Russian-cca1” language : russian-cca1.mld and russian-cca1.mlo	485
13.139	“Russian-lh” language : russian-lh.mld and russian-lh.mlo	486
13.140	“Russian-lhcyralt” language : russian-lhcyralt.mld and russian-lhcyralt.mlo	486
13.141	“Russian-lhcyrkoi” language : russian-lhcyrkoi.mld and russian-lhcyrkoi.mlo	487
13.142	“Russian-lhcyrwin” language : russian-lhcyrwin.mld and russian-lhcyrwin.mlo	487
13.143	“Samin” language : samin.mld	488

13.144	“Scottish” language : <code>scottish.mld</code>	488
13.145	“Serbian” language : <code>serbian.mld</code>	489
13.146	“Serbianc” language : <code>serbianc.mld</code>	489
13.147	“Slovak” language : <code>slovak.mld</code>	490
13.148	“Slovene” language : <code>slovene.mld</code>	490
13.149	“Spanish” language : <code>spanish.mld</code>	491
13.150	“Spanish2” language : <code>spanish2.mld</code>	492
13.151	“Spanish3” language : <code>spanish3.mld</code>	492
13.152	“Spanish4” language : <code>spanish4.mld</code>	493
13.153	“Swedish” language : <code>swedish.mld</code>	493
13.154	“Swedish2” language : <code>swedish2.mld</code>	494
13.155	“Thai” language : <code>thai.mld</code> and <code>thai.mlo</code>	494
13.156	“Turkish” language : <code>turkish.mld</code>	495
13.157	“Uighur” language : <code>uighur.mld</code>	495
13.158	“Uighur2” language : <code>uighur2.mld</code>	495
13.159	“Uighur3” language : <code>uighur3.mld</code>	496
13.160	“UKenglish” language : <code>UKenglish.mld</code>	496
13.161	“Ukraineb” language : <code>ukraineb.mld</code>	496
13.162	“Ukrainian” language : <code>ukrainian.mld</code>	496
13.163	“Uppersorbian” language : <code>uppersorbian.mld</code>	497
13.164	“USenglish” language : <code>USenglish.mld</code>	498
13.165	“Usorbian” language : <code>usorbian.mld</code>	498
13.166	“Vietnam” language : <code>vietnam.mld</code>	498
13.167	“Vietnamese” language : <code>vietnamese.mld</code>	499
13.168	“Welsh” language : <code>welsh.mld</code>	499
13.169	“Xalx” language : <code>xalx.mld</code>	500
13.170	“Xalx2” language : <code>xalx2.mld</code>	500
13.171	“Xalx3” language : <code>xalx3.mld</code>	501

III	Complements	502
	Bibliography	504
	Changes History	522
	Index	549

List of Figures

1.1	Float barriers with the <code>section</code> option	30
1.2	Float barriers with the <code>above</code> option	30
1.3	Float barriers with the <code>below</code> option	30
1.4	Float barriers with the <code>above</code> and <code>below</code> options	30
1.5	Layout of a ToC (LoF, LoT) entry	42
2.1	Three compilations for <code>mini toc</code>	55
9.1	Hack to detect the limitation to short extensions.	252

List of Tables

1.1	Commands for a <code>minitoc</code>	27
1.2	Commands for a <code>secttoc</code>	28
1.3	Commands for a <code>parttoc</code>	31
1.4	Commands to add an entry in the table of contents for a starred part, chapter, or section.	33
1.5	Fonts and titles for the mini-table commands.	34
1.6	Fonts for the mini-table entries.	35
1.7	Available languages	36
1.8	Horizontal rules	43
1.9	Page numbers	43
1.10	Features for mini-tables	45
1.11	Extensions of the auxiliary files	52
2.1	Kernings before minitable bottom rules	74
3.1	Package options	75
3.2	General commands	76
3.3	Commands : part level	77
3.4	Commands : chapter level	78
3.5	Commands : section level	79
3.6	Commands for horizontal rules	80
3.7	Commands for page numbers	80
3.8	Commands for mini-tables features	81
3.9	Preparation and insertion commands	81
3.10	Adjustment commands	82
3.11	Obsolete commands	82
6.1	Category codes	190
6.2	Encoding schemes implemented in CJK	191
6.3	Document classes	192
6.4	Depths for sectionning commands	193
6.5	Most common font encodings	204
6.6	Most common font families	205
6.7	Most common font series	205
6.8	Most common font shapes	205
6.9	The five font parameters for some fonts	205
6.10	Author commands for fonts	206

<i>LIST OF TABLES</i>	18
7.1 List of files (<i>minitoc.1</i>)	215

About this document

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

- The most useful chapters are in the first part (“User’s Manual”, page 22):
 - the chapter “The minitoc package”, page 23, describes the essential commands to use the package;
 - the chapter “Frequently Asked Questions”, page 53, may be useful to solve some specific problems;
 - the “Memento” chapter, page 75, is a set of tables to be used as a remainder of the commands of this package;
 - the “Examples of documents” chapter, page 83, gives the code of some documents showing the basic usage of the minitoc package and some interesting situations;
 - the “Messages” chapter, page 137, 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;
 - the “Jargon” chapter, page 184, attempts to explain most of the technical terms used here;
 - the “Installation” chapter, page 214, describes all the files included in the distribution of the package;
 - the “Postface” chapter, page 218, gives an abbreviated history of the package.
- The second part, “Implementation”, page 229, is much more technical; you can read it if you are interested in the details of the coding of the package. The chapter “The language definition (.mld) and object (.mlo) files”, page 412, may be useful if you are interested by some language.
- The third part, “Complements”, page 503, contains a bibliography, a detailed history of the package, and an index.

For this document, I have used:

- a short table of contents (summary), with the \shorttoc command from my shorttoc package [92], displaying only parts and chapters;

- 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, these tables should help you to navigate in the document.

Part I

User's Manual

Contents of the First Part

1	The <code>minitoc</code> package	23
2	Frequently Asked Questions	53
3	Memento	75
4	Examples of documents	83
5	Messages	137
6	Jargon	184
7	Installation	214
8	Postface	218
	Acknowledgments	226

Chapter 1

The minitoc package

Contents

1.1	Introduction	24
1.1.1	Important restrictions	25
1.1.2	Version	25
1.2	License	25
1.3	How to use the minitoc package	26
1.3.1	Loading the package and creating the mini-tables	26
1.3.2	Preparing the mini-tables	26
1.3.3	Placing the mini-tables	28
1.3.4	Starred chapters, parts and sections	32
1.4	Typesetting of the mini-tables	33
1.4.1	Chapter-level mini-tables	33
1.4.2	Titles for chapter-level mini-tables	35
1.4.3	Part-level mini-tables	37
1.4.4	Titles for part-level mini-tables	37
1.4.5	Section-level mini-tables	37
1.4.6	Titles for section-level mini-tables	38
1.4.7	Position of the titles	38
1.4.7.1	For mini-tables at the part level	38
1.4.7.2	For mini-tables at the chapter level	38
1.4.7.3	For mini-tables at the section level	38
1.4.7.4	Summary of the positionning of titles	39
1.4.8	Line spacing in the mini-tables	39
1.4.9	Simplified commands for fonts	40
1.4.10	Simplified command for mini-table titles	40
1.4.11	Simplified command for mini-table depths	41
1.4.12	Languages for the titles	41
1.4.13	Altering the layout of the mini-tables	42
1.5	Special Features	44
1.5.1	Horizontal Rules	44
1.5.2	Page Numbers, Leaders	44
1.5.3	Features for parttocs and other mini-tables	45
1.5.3.1	Remark about page styles	46
1.5.4	The “Chapter 0” Problem (solved)	46

1.5.4.1	Before version #23 (1994/11/08)	47
1.5.4.2	Since version #23 (1994/11/08)	47
1.5.5	Special Entries for TOC, LOF, LOT, Bibliography and Index	47
1.6	The notoccite option	49
1.7	The listfiles and nolistfiles options	49
1.8	The hints option	49
1.9	Usage with MS-DOS	51
1.10	Why several L^AT_EX runs are required?	51
1.11	The mtcoff package	52

Figures

1.1	Float barriers with the section option	30
1.2	Float barriers with the above option	30
1.3	Float barriers with the below option	30
1.4	Float barriers with the above and below options	30
1.5	Layout of a ToC (LoF, LoT) entry	42

Tables

1.1	Commands for a minitoc	27
1.2	Commands for a secttoc	28
1.3	Commands for a parttoc	31
1.4	Commands to add an entry in the table of contents for a starred part, chapter, or section.	33
1.5	Fonts and titles for the mini-table commands.	34
1.6	Fonts for the mini-table entries.	35
1.7	Available languages	36
1.8	Horizontal rules	43
1.9	Page numbers	43
1.10	Features for mini-tables	45
1.11	Extensions of the auxiliary files	52

1.1 Introduction

The minitoc package, initially written by Nigel WARD and Dan JURAFSKY, has been almost completely redesigned by Jean-Pierre F. DRUCBERT (ONERA/Centre de Toulouse). A summary of the evolution of this package is given in the chapter 8 on page 218. This package creates a mini-table of contents (a “minitoc”¹) 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) or sections (classes like article²). Thus, this package should not be used with document classes without standard sectionning commands (like

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 loads the standard article class, this class will be just considered as a variant of the article class.

letter). When the document class defines a “part” sectionning 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 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 sectionning 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 #47. You will find a resumed history of the package in the “Postface” chapter (chapter 8 on page 218) and a more detailed history in “Changes History”, page 522.

1.2 License

This package must be distributed and/or may be modified under the conditions of the **L^AT_EX 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 L^AT_EX 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 improvements are welcome.

1.3 How to use the minitoc package

1.3.1 Loading the package and creating the mini-tables

`\usepackage` To use the minitoc package, you must introduce a command:
`\minitoc`
`\chapter` `\usepackage[...options...]{minitoc}`

in the preamble of the document³. 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*⁴ 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 following page.

For each mini-table of contents, an auxiliary file will be created with a name of the form `<document>.mtc<N>`, where `<N>` is the absolute chapter number. “Absolute” means that this number is unique, and always increasing from the first chapter⁵. The suffix is `.mlf<N>` for mini-lists of figures and is `.mlt<N>` for mini-lists of tables. (If under MS-DOS or any operating system with short extensions to filenames, see section 1.9 on page 51 and section 2.5 on page 56). There are similar commands for mini-tables at the part or section level, depending of the document class.

1.3.2 Preparing the mini-tables

`\dominitoc` The commands⁶ `\dominitoc`, `\dominilof`, and `\dominilot` (for mini-tables at
`\dominilof` the chapter level), take respectively the `<document>.toc`, `<document>.lof`, and
`\dominilot` `<document>.lot` files, and cut slices from them to create the `<document>.mtc<N>`,
`<document>.mlf<N>`, and `<document>.mlt<N>` files.

³ This command must be placed *after* any modification done on the sectionning commands; if you modify sectionning 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 ask for details with a high value for `minitocdepth`.

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 [79], by David CARLISLE, with his permission.

Table 1.1: Commands for a minitoc

\documentclass[...]{book}		
\usepackage[...options...]{minitoc}		
...		
\setlength{\mtcindent}{24pt}	default	
\setlength{\mtcskipamount}{\bigskipamount}	default	
...		
\setcounter{minitocdepth}{2}	default	
\renewcommand{\mtcfont}{\small\rmfamily\upshape\mdseries}		default
\renewcommand{\mtcSfont}{\small\rmfamily\upshape\bfseries}		default
or:		
\mtcsetdepth{minitoc}{2}		default
\mtcsetfont{minitoc}{*}{\small\rmfamily\upshape\mdseries}		default
\mtcsetfont{minitoc}{section}{\small\rmfamily\upshape\bfseries}		default
...		
\begin{document}		
...		
\dominitoc		
\dominilof		
\dominilot		
\tableofcontents	or \faketableofcontents	
\listoffigures	or \fakelistoffigures	
\listoftables	or \fakelistoftables	
...		
\chapter{...}		
\minitoc	if you want one	
\mtcskip		
\minilof	if you want one	
\mtcskip		
\minilot	if you want one	
...		

\dosecttoc	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.
\dosectlof	
\dosectlot	
\doparttoc	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.
\dopartlof	
\dopartlot	
\mtcprepare	This command accepts also an optional argument to set the default position of the title for all the mini-tables.
\tableofcontents	
\listoffigures	
\listoftables	To obtain a satisfactory result (i.e., non empty), please note that all these commands must <i>imperatively be put before</i> 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 figure (for the \do...lof commands) or in the list of tables (for the \do...lot) commands; else disorder in the mini-tables may result.



Table 1.2: Commands for a secttoc

\documentclass[...]{article}		
\usepackage[...options...]{minitoc}		
...		
\setlength{\stcindent}{24pt}	default	
...		
\setcounter{secttocdepth}{2}	default	
\renewcommand{\stcfont}{\small\rmfamily\upshape\mdseries}		default
\renewcommand{\stcSSfont}{\small\rmfamily\upshape\bfseries}		default
or:		
\mtcsetdepth{secttoc}{2}		default
\mtcsetfont{secttoc}{*}{\small\rmfamily\upshape\mdseries}		default
\mtcsetfont{secttoc}{subsection}{\small\rmfamily\upshape\bfseries}		default
...		
\begin{document}		
...		
\dosecttoc		
\dosectlof		
\dosectlot		
\tableofcontents	or \faketableofcontents	
\listoffigures	or \fakelistoffigures	
\listoftables	or \fakelistoftables	
...		
\section{...}		
\secttoc	if you want one	
\sectlof	if you want one	
\sectlot	if you want one	
...		

1.3.3 Placing the mini-tables

`\mtcskip` The `\mtcskip` command may be used to add a vertical skip between two mini-tables.

`\mtcskipamount` 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.

`\bigskipamount`

`\secttoc` 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. Of course, it is 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 the section-level table of contents for a section, you must use the sequence given in Table 1.2.

`\section`

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 `.slf<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 51 and section 2.5 on page 56).

`\usepackage`
`\FloatBarrier`



As floats (figures and tables) could drift⁷ 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 coherent sectlofs and sectlots. 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 `insection` option loads the `placeins` package [9] with its `verbose` and `section` options. 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 becomes “porous” upwards⁸ (\uparrow) or downwards (\downarrow), or both (\Updownarrow). The `section` option makes a more “watertight” barrier (\equiv). This is illustrated by the four figures 1.1 to 1.4 on the following page.

The `placeins` package, by Donald ARSENAU, is available on CTAN archives; 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 [138] and [159, 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 `.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 [147] or, better, of the `floatrow` package [135].



If you want to add the partial table of contents for a part, you must use the sequence given in Table 1.3 on page 31. 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 51 and section 2.5 on page 56).



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`).

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 ARSENAU

⁸ But a float can not drift upwards beyond the top of the current page.

Command	Action
\section	× × × × × × × × × × <i>Forbidden Area</i> × × × × × × × × × ×
	=====
	<i>Allowed Area</i>
	Float inserted here
\section	<i>Allowed Area</i>
	=====
	× × × × × × × × × × <i>Forbidden Area</i> × × × × × × × × × ×
	=====

Figure 1.1: Float barriers with the section option

Command	Action
\section	<i>Allowed Area</i>
	↑ ↑
	<i>Allowed Area</i>
	Float inserted here
\section	<i>Allowed Area</i>
	=====
	× × × × × × × × × × <i>Forbidden Area</i> × × × × × × × × × ×
	=====

Figure 1.2: Float barriers with the above option

Command	Action
\section	× × × × × × × × × × <i>Forbidden Area</i> × × × × × × × × × ×
	=====
	<i>Allowed Area</i>
	Float inserted here
\section	<i>Allowed Area</i>
	↓ ↓
	<i>Allowed Area</i>
	=====

Figure 1.3: Float barriers with the below option

Command	Action
\section	<i>Allowed Area</i>
	↕ ↕
	<i>Allowed Area</i>
	Float inserted here
\section	<i>Allowed Area</i>
	↕ ↕
	<i>Allowed Area</i>
	=====

Figure 1.4: Float barriers with the above and below options

Table 1.3: Commands for a parttoc

\documentclass[...]{book}		
\usepackage[...options...]{minitoc}		
...		
\setlength{\ptcindent}{0pt}	default	
...		
\setcounter{parttocdepth}{2}	default	
\renewcommand{\ptcfont}{\normalsize\rmfamily\upshape\mdseries}		default
\renewcommand{\ptcCfont}{\normalsize\rmfamily\upshape\bfseries}		default
\renewcommand{\ptcSfont}{\normalsize\rmfamily\upshape\mdseries}		default
or:		
\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	
...		

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 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 28) package option (default) skips empty mini-tables (with a note in the `.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 [84, 86] 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

`\addstarredpart` **NOTE:** if using `\chapter*` and a
`\addstarredchapter`
`\addstarredsection` `\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}{...}`

for example, 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}

```

`\adjustptc` There is sometimes a problem with mini-tables when you use `\chapter*` (or
`\adjustmtc` `\section*`): the minitocs appear in the wrong chapter. You can add a `\adjustmtc`
`\adjuststc` (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 52) knows what
to do about `\adjustmtc` (and others), but can do nothing about `\stepcounter`
or `\addtocounter`, as they are a standard basic commands of L^AT_EX, not minitoc
specific commands. Syntax:

```

\adjustptc[n]
\adjustmtc[n]
\adjuststc[n]

```

where n is the increment (default: 1).

Table 1.4: Commands to add an entry in the table of contents for a starred part, chapter, or section.

Level	With title
part	<code>\mtcaddpart[⟨title⟩]</code>
chapter	<code>\mtcaddchapter[⟨title⟩]</code>
section	<code>\mtcaddsection[⟨title⟩]</code>

`\decrementptc` There are similar commands to *decrement* or *increment* by 1 these counters: `\decrementptc`, `\decrementmtc`, `\decrementstc`, `\incrementptc`, `\incrementmtc`, and `\incrementstc`; the same remarks as above apply. These commands have no argument.

`\mtcaddpart` A more clever way to solve this problem is to use commands similar to:
`\mtcaddchapter`
`\mtcaddsection` `\mtcaddchapter[⟨title⟩]`



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 68 for the problems with `\mtcaddpart`.

1.4 Typesetting of the mini-tables

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

1.4.1 Chapter-level mini-tables

`\mtcfont` 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, `\mtcSfont` minilof or minilot. More selective choices are made with the following fonts. `\mtcSSfont` Section entries are typeset in the `\mtcSfont` font, which is `\small\bfseries` by default. For subsections, subsubsections, paragraphs and subparagraphs, the commands `\mtcSPfont` `\mtcSSfont`, `\mtcSSSfont`, `\mtcPfont` and `\mtcSPfont` are available (by default, `\mlffont` `\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. `\mlfSfont` There are also `\mlfSfont` and `\mltSfont` for sub-figures and sub-tables entries. `\mltSfont` Tables 1.5 on the following page and 1.6 on page 35 summarize these many commands⁹.

⁹ Thanks to Stefan ULRICH, who contributed these tables initially.

Table 1.5: Fonts and titles for the mini-table commands.

Command	Font default setting	Title string default setting	Title font default setting
For the <code>\part...</code> commands:			
<code>\parttoc</code>	<code>\ptcfont</code> <code>\normalsize\rmfamily*</code> <code>\small\rmfamily**</code>	<code>\ptctitle</code> Table of Contents [†]	<code>\ptifont</code> <code>\LARGE\bfseries*</code> <code>\Large\bfseries**</code>
<code>\partlof</code>	<code>\plffont</code> <code>\normalsize\rmfamily*</code> <code>\small\rmfamily**</code> <code>\plfSfont</code> <code>\normalsize\rmfamily*</code> <code>\small\rmfamily**</code>	<code>\plftitle</code> List of Figures [†]	<code>\ptifont</code> <code>\LARGE\bfseries*</code> <code>\Large\bfseries**</code>
<code>\partlot</code>	<code>\pltfont</code> <code>\normalsize\rmfamily*</code> <code>\small\rmfamily**</code> <code>\pltSfont</code> <code>\normalsize\rmfamily*</code> <code>\small\rmfamily**</code>	<code>\plttitle</code> List of Tables [†]	<code>\ptifont</code> <code>\LARGE\bfseries*</code> <code>\Large\bfseries**</code>
For the <code>\mini...</code> commands:			
<code>\minitoc</code>	<code>\mtcfont</code> <code>\small\rmfamily</code>	<code>\mtctitle</code> Contents [†]	<code>\mtifont</code> <code>\large\bfseries</code>
<code>\minilof</code>	<code>\mlffont</code> <code>\small\rmfamily</code> <code>\mlfSfont</code> <code>\small\rmfamily</code>	<code>\mlftitle</code> Figures [†]	<code>\mtifont</code> <code>\large\bfseries</code>
<code>\minilot</code>	<code>\mltfont</code> <code>\small\rmfamily</code> <code>\mltSfont</code> <code>\small\rmfamily</code>	<code>\mlttitle</code> Tables [†]	<code>\mtifont</code> <code>\large\bfseries</code>
For the <code>\sect...</code> commands:			
<code>\secttoc</code>	<code>\stcfont</code> <code>\small\rmfamily</code>	<code>\stctitle</code> Contents [†]	<code>\stifont</code> <code>\Large\bfseries</code>
<code>\sectlof</code>	<code>\slffont</code> <code>\small\rmfamily</code> <code>\slfSfont</code> <code>\small\rmfamily</code>	<code>\slftitle</code> Figures [†]	<code>\stifont</code> <code>\Large\bfseries</code>
<code>\sectlot</code>	<code>\sltfont</code> <code>\small\rmfamily</code> <code>\sltSfont</code> <code>\small\rmfamily</code>	<code>\slttitle</code> Tables [†]	<code>\stifont</code> <code>\Large\bfseries</code>

*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.



Note that the default choice of fonts is certainly not perfect and hence 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 mixed¹⁰ (the true length of 1em is not the same for the fonts). This can eventually be adjusted by changing some fonts.

¹⁰This appears, e.g., if you are using the Computer Modern Roman (CMR) fonts [120]. The symptom disappears if you do not use bold CMR fonts or if you use the TX fonts (txfonts package [184]), by example, like in this document. See also section 2.29 on page 67.

Table 1.6: Fonts for the mini-table entries.

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

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

1.4.2 Titles for chapter-level mini-tables

<code>\mtifont</code>	Titles are typeset in the <code>\mtifont</code> (<code>\large\bfseries</code> by default) font and the text
<code>\mtctitle</code>	strings of the titles are defined by <code>\mtctitle</code> , <code>\mlftitle</code> and <code>\mltttitle</code> , which are
<code>\mlftitle</code>	the strings “Contents”, “Figures” and “Tables” by default. These title commands should
<code>\mltttitle</code>	be redefined by <code>\renewcommand</code> or <code>\mtcsettitle</code> for languages other than english.
<code>\mtcsettitle</code>	
<code>\mtcselectlanguage</code>	The language option files like <code>french.mld</code> and <code>english.mld</code> (the suffix <code>.mld</code> means “minitoc language definition (file)”) (and many others, see the list in table 1.7 on the next page and section 1.4.12 on page 41) are available. You can easily prepare a similar file for a preferred language (see section 2.26 on page 67). You can change the language of these titles by using the <code>\mtcselectlanguage{language}</code> macro.

Table 1.7: Available languages

1. afrikaan (afrikaans)	28. english (american, australian, british, canadian, newzealand, UKenglish, USenglish)	49. greek- polydemo ^{c,e}	79. magyar2	106. russian-lh ^{c,g,h}
2. albanian		50. greek- polykatha ^{c,e}	80. magyar3	107. russian- lhcyralt ^{c,g,h}
3. arab (arabic) ^c		51. guarani ^h	81. malayalam- keli ^c	108. russian- lhcyrkoi ^{c,g,h}
4. arab2 ^{a,c}		52. hangul1 ^{c,d,g}	82. malayalam- omega ^{c,e,g,h}	109. russian- lhcyrwin ^{c,g,h}
5. arabi ^{c,j}		53. hangul2 ^{c,d,g}	83. malayalam- rachana ^c	110. samin
6. armenian ^c	29. english1	54. hangul3 ^{c,d,g}	84. malayalam- rachana2 ^c	111. scottish
7. bahasai (bahasa, indon, indonesian) ^c	30. english2	55. hangul4 ^{c,d,g}	85. mongol ^c	112. serbian
8. bahasam (malay, meyalu) ^c	31. esperant (esperanto)	56. hangul- u8 ^{c,e,f,g,h}	86. ngermanb (ngerman, naustrian)	113. serbianc ^c
9. bangla ^c	32. estonian	57. hanja1 ^{c,d,g}	87. ngermanb2	114. slovak
10. basque	33. ethiopia (ethiopian) ^c	58. hanja2 ^{c,d,g}	88. norsk	115. slovene
11. bicig (uighur) ^{c,i}	34. ethiopian2 ^{c,e,h}	59. hanja-u8 ^{c,e,f,g,h}	89. norsk2	116. spanish (castillan, castillian)
12. bicig2 (uighur2) ^{c,i}	35. farsi1 ^{c,f,g}	60. hebrew ^{c,h}	90. nynorsk	117. spanish2
13. bicig3 (uighur3) ^{c,i}	36. farsi2 ^{c,f,g}	61. hebrew2 ^{c,h}	91. nynorsk2	118. spanish3 ^{e,f}
14. bithe (manju) ^c	37. farsi3 ^{c,j}	62. icelandic ^f	92. polish	119. spanish4
15. brazil (brazilian)	38. finnish	63. interlingua	93. polish2 ^{c,e}	120. swedish
16. breton	39. finnish2	64. irish	94. polski ^c	121. swedish2
17. bulgarian ^c	40. french (frenchb, frenchle, frenchpro, francais, acadien, canadien)	65. italian	95. portuguese (portuges)	122. thai ^{c,d,f,g}
18. bulgarianb ^c		66. italian2	96. romanian	123. turkish
19. buryat ^c	41. french1	67. japanese ^{c,d,g}	97. romanian2	124. ukrainian (ukraineb) ^{b,c}
20. buryat2 ^c	42. french2	68. japanese2 ^{c,d,g}	98. romanian3	125. uppersorbian (usorbian)
21. catalan	43. galician	69. japanese3 ^{c,d,g}	99. russian ^{b,c}	126. vietnam (vietnamese) ^{c,d}
22. chinese1 ^{c,g}	44. german (austrian)	70. japanese4 ^{c,d,g}	100. russianb ^{b,c}	127. welsh
23. chinese2 ^{c,g}	45. germanb	71. japanese5 ^{c,d,g}	101. russianc ^{b,c}	128. xalx (khalkha) ^c
24. croatian	46. germanb2	72. kannada ^c	102. russian2m ^{c,e}	129. xalx2 ^c
25. czech	47. greek ^c	73. latin	103. russian2o ^{c,e}	130. xalx3 ^c
26. danish	48. greek-mono ^{c,e}	74. latin2	104. russian-cca ^{c,g,h}	
27. dutch		75. latvian (letton) ^e	105. russian- cca1 ^{c,g,h}	
		76. lithuanian		
		77. lowersorbian (lsorbian)		
		78. magyar (hungarian)		

The languages in parentheses are aliases of a main language and their .mld files will load the .mld file of that main language.

^a The arab(ic) and arab2 languages require the use of the ArabTeX package [130, 131].

^b The russian language is not yet supported by the babel system [30, 31], but russianb [136] is supported if you use babel-3.6 or a higher version; russianc is an extra. Look also at other .mld files for russian.

^c Some languages may require specific fonts.

^d Requires the CJK package [143, 144].

^e Requires Lambda (Λ), the version of L^AT_EX for Omega (Ω).

^f Requires a 8-bits input encoding.

^g Uses also a .mlo file.

^h Requires a specific input encoding.

ⁱ The bicig language is also known as uighur.

^j The arabi and farsi3 languages require the use of the Arabi package [116].

1.4.3 Part-level mini-tables

`\ptcfont` 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 `\normalsize\bfseries` by default. Section entries are typeset in the `\ptcSfont` font, which is `\normalsize\rmfamily` by default. For subsections, subsubsections, paragraphs and subparagraphs, the commands `\ptcSSfont`, `\ptcSSSfont`, `\ptcPfont`, and `\ptcSPfont` are available (by default, `\normalsize\rmfamily`) if you want to use various fonts. Partial lists of figures and tables are typeset in the fonts `\plfSfont` and `\pltSfont` for sub-figures and sub-tables entries.

1.4.4 Titles for part-level mini-tables

`\ptifont` Titles are typeset in the `\ptifont` (`\LARGE\bfseries` by default) font and the text strings of the titles are defined by `\ptctitle`, `\plftitle` and `\plttitle`, which are the strings “Table of Contents”, “List of Figures” and “List of Tables” by default. `\plftitle` 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 preceding page) are available. Read also section 1.4.12 on page 41. You can easily prepare a similar file for a preferred language (see section 2.26 on page 67). You can change the language of these titles by using the `\mtcselectlanguage{language}` macro.

1.4.5 Section-level mini-tables

`\stcfont` The section-level table of contents is typeset in the `\stcfont` font, which is defined as `\normalsize\rmfamily` by default. In fact, the font `\stcfont` 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 `\stcSSfont` font, which is `\normalsize\bfseries` by default. Subsubsection entries are typeset in the `\stcSSSfont` font, which is `\normalsize\rmfamily` by default. For paragraphs and subparagraphs, the commands `\stcPfont` and `\stcSPfont` 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 `\slfSfont` and `\sltSfont` for sub-figures and sub-tables entries.

1.4.6 Titles for section-level mini-tables

`\stifont` Titles are typeset in the `\stifont` (`\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 option files like `french.mld` and `english.mld` (and also many others, as listed in table 1.7 on page 36 and explained in section 1.4.12 on page 41) are available. You can easily prepare a similar file for your preferred language (see section 2.26 on page 67). 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

`\doparttoc` 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: `[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 `parttoc` (or `partlof` or `partlot`), just use such an optional argument with the command `\parttoc` (or `\partlof` or `\partlot`).

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
`\dopartlof` preparation commands:
`\dopartlot`
`\dominitoc` `\doparttoc`, `\dopartlof`, `\dopartlot`,
`\dominilof` `\dominitoc`, `\dominilof`, `\dominilot`,
`\dominilot` `\dosecttoc`, `\dosectlof`, `\dosectlot`,
`\dosecttoc` `\mtcprepare`
`\dosectlof`
`\dosectlot` accepts an optional argument to change the positioning of the title: [l] for left (default),
`\mtcprepare` [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:
`\partlof`
`\partlot` `\parttoc`, `\partlof`, `\partlot`,
`\minitoc` `\minitoc`, `\minilof`, `\minilot`,
`\minilof` `\secttoc`, `\sectlof`, `\sectlot`
`\minilot`
`\secttoc` accept the same optional arguments, but these options change the positioning only for
`\sectlof` the title of the current mini-table.
`\sectlot`

1.4.8 Line spacing in the mini-tables

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

But with the KOMA-Script classes [164] (`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 of the options given to these KOMA-Script classes (`parindent` option, `parskip` option and variants).

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

1.4.9 Simplified commands for fonts

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

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

By example,

```
\mtcsetfont{minitoc}{subsection}%
    {\small\rmfamily\upshape\bfseries}

\mtcsetfont{minilof}{subfigure}%
    {\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 `\pti font` (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}
```

By example,

```
\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}
```

By example,

```
\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 Languages for the titles

Most of the strings defined in the language option files (`.mld`) were taken from the superb `babel` package [30, 31] of Johannes BRAAMS, some were adapted, others were made available by gentle users or taken from specific packages, like `ArabTeX` [130, 131], `AraBi` [116], `ArmTeX` (armenian) [90], `BangTeX` (bangla) [169], `ethiop` [23], `guarani` [26], `Malayalam` [3, 4], `MonTeX` (mongol) [87, 89], `CJK` (chinese, corean-hangûl/hanja, japanese, thai) [143, 144], `PLATeX` [168, 204] (polish), `SIATeX` [152] (swedish), `FarsiTeX` (farsi or iranian), or `vietnam.sty` — latvian (letton), greek-mono, greek-polydemo, greek-polykatha, `polish2`, `russian2m`, `russian2o` and `spanish3` need `Lambda (Λ)`, i.e., the `Omega (Ω)` version of `LATeX`, (see [127]), 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. See table 1.7 on page 36.



But for some oriental languages¹¹, 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` file¹², 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.

¹¹Mainly for chinese, farsi, hangûl (korean), hanja (korean), japanese, malayalam-omega, thai and some variants of russian.

¹²The extension `.mlo` means *minitoc language object*.

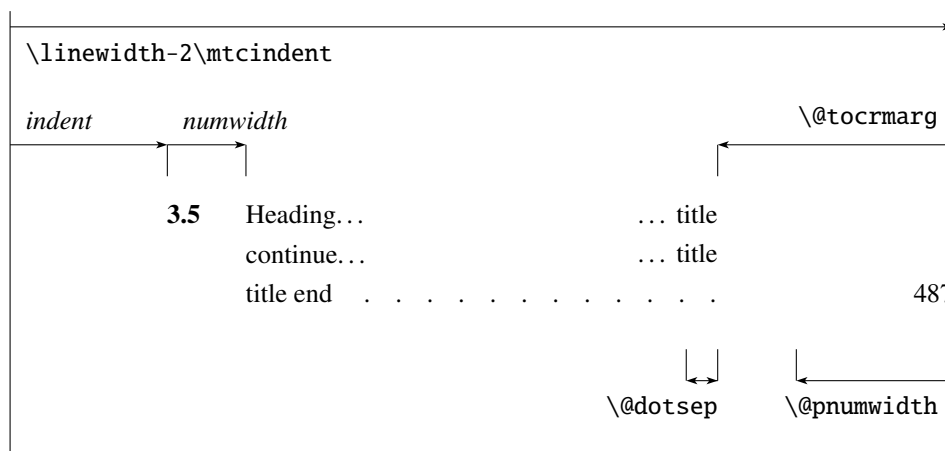


Figure 1.5: Layout of a ToC (LoF, LoT) entry

1.4.13 Altering the layout of the mini-tables

The layout of a mini-table is described in the figure 1.5 (this figure is adapted from [207]), 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 “... 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.

`\mtcsetformat` As these commands are internal (their names contain the “@” character) and must have a local effect only on specific 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

Table 1.8: Horizontal rules

				defaults for		
	rules in		no rules in	book	report	article
<code>\ptcrule</code>	parttocs	<code>\noptcrule</code>	parttocs	N	N	Y
<code>\plfrule</code>	partlofs	<code>\noplfrule</code>	partlofs	N	N	Y
<code>\pltrule</code>	partlots	<code>\nopltrule</code>	partlots	N	N	Y
<code>\mtcrule</code>	minitocs	<code>\nomtcrule</code>	minitocs	Y	Y	(NA)
<code>\mlfrule</code>	minilofs	<code>\nomlfrule</code>	minilofs	Y	Y	(NA)
<code>\mltrule</code>	minilots	<code>\nomltrule</code>	minilots	Y	Y	(NA)
<code>\stcrule</code>	secttocs	<code>\nostcrule</code>	secttocs	(NA)	(NA)	Y
<code>\slfrule</code>	sectlofs	<code>\noslfrule</code>	sectlofs	(NA)	(NA)	Y
<code>\sltrule</code>	sectlots	<code>\nosltrule</code>	sectlots	(NA)	(NA)	Y

(NA) = not available.

Table 1.9: Page numbers

Type	Page numbers (Default)	No page numbers
parttoc	<code>\ptcpagenumbers</code>	<code>\noptcpagenumbers</code>
minitoc	<code>\mtcpagenumbers</code>	<code>\nomtcpagenumbers</code>
secttoc	<code>\stcpagenumbers</code>	<code>\nostcpagenumbers</code>
partlof	<code>\plfpagenumbers</code>	<code>\noplfpagenumbers</code>
minilof	<code>\mlfpagenumbers</code>	<code>\nomlfpagenumbers</code>
sectlof	<code>\slfpagenumbers</code>	<code>\noslfpagenumbers</code>
partlot	<code>\pltpagenumbers</code>	<code>\nopltpagenumbers</code>
minilot	<code>\mltpagenumbers</code>	<code>\nomltpagenumbers</code>
sectlot	<code>\sltpagenumbers</code>	<code>\nosltpagenumbers</code>

one of the `dotinterval` (for `\@dotsep`), `pagenumwidth` (for `\@pnumwidth`), or `tocrightmargin` (for `\@tocrmarg`) keywords; so:

```
\mtcsetformat{partlof}{tocrightmargin}{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.

Note that the `tocrightmargin` (for `\@tocrmarg`) parameter should obviously be greater than the `pagenumwidth` parameter (this appears in the figure 1.5 on the preceding page).

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.

1.5 Special Features

1.5.1 Horizontal Rules

`\mtcsetrules` 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 on the page before are available. But you can also use the following command, which is simpler:

```
\mtcsetrules{mini-table|*}{on|off}
```

where *mini-table* is one of the `parttoc`, `partlof`, `partlot`, `minitoc`, `minilof`, `minilot`, `secttoc`, `sectlof`, or `sectlot` keywords; if the first argument is a star (*), all mini-tables are affected; the keywords `on` and `off` have the following synonyms¹³:

- `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`, `N`, `false`, `FALSE`, `faux`, `FAUX`, `f`, `F`, `non`, `NON`, and `0`.

1.5.2 Page Numbers, Leaders

`\mtcsetpagenumbers` By default, the page numbers are listed in each `minitoc`, `minilof`, etc. Some authors want only the section titles (with the section numbers), but without page numbers. Hence the obvious declarations of table 1.9 on the preceding page are available. But you can also use the following command:

```
\mtcsetpagenumbers{mini-table|*}{on|off}
```

where *mini-table* is one of the `parttoc`, `partlof`, `partlot`, `minitoc`, `minilof`, `minilot`, `secttoc`, `sectlof`, or `sectlot` keywords; the keywords `on` and `off` have the following synonyms¹³:

- `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`, `N`, `false`, `FALSE`, `faux`, `FAUX`, `f`, `F`, `non`, `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 `undotted` package option removes these dots. The `dotted` package option is the default. See also section 1.4.13 on page 42.

¹³ `0` and `o` are the letter `O`, `0` is the zero digit.

Table 1.10: Features for mini-tables

Type	Command	Default
parttoc	<code>\beforeparttoc</code>	<code>\cleardoublepage</code>
parttoc	<code>\afterparttoc</code>	<code>\cleardoublepage</code>
parttoc	<code>\thispageparttocstyle</code>	<code>\thispagestyle{empty}</code>
partlof	<code>\beforepartlof</code>	<code>\cleardoublepage</code>
partlof	<code>\afterpartlof</code>	<code>\cleardoublepage</code>
partlof	<code>\thispagepartlofstyle</code>	<code>\thispagestyle{empty}</code>
partlot	<code>\beforepartlot</code>	<code>\cleardoublepage</code>
partlot	<code>\afterpartlot</code>	<code>\cleardoublepage</code>
partlot	<code>\thispagepartlotstyle</code>	<code>\thispagestyle{empty}</code>
minitoc	<code>\beforeminitoc</code>	<code>\empty</code>
minitoc	<code>\afterminitoc</code>	<code>\empty</code>
minitoc	<code>\thispageminitocstyle</code>	<code>\empty</code>
minilof	<code>\beforeminilof</code>	<code>\empty</code>
minilof	<code>\afterminilof</code>	<code>\empty</code>
minilof	<code>\thispageminilofstyle</code>	<code>\empty</code>
minilot	<code>\beforeminilot</code>	<code>\empty</code>
minilot	<code>\afterminilot</code>	<code>\empty</code>
minilot	<code>\thispageminilotstyle</code>	<code>\empty</code>
secttoc	<code>\beforesecttoc</code>	<code>\empty</code>
secttoc	<code>\aftersecttoc</code>	<code>\empty</code>
secttoc	<code>\thispagesecttocstyle</code>	<code>\empty</code>
sectlof	<code>\beforesectlof</code>	<code>\empty</code>
sectlof	<code>\aftersectlof</code>	<code>\empty</code>
sectlof	<code>\thispagesectlofstyle</code>	<code>\empty</code>
sectlot	<code>\beforesectlot</code>	<code>\empty</code>
sectlot	<code>\aftersectlot</code>	<code>\empty</code>
sectlot	<code>\thispagesectlotstyle</code>	<code>\empty</code>
<code>\mtcsetfeature{mini-table}{before after pagestyle}{command}</code> Modifies the features for a mini-table.		

1.5.3 Features for parttocs and other mini-tables

By default, a parttoc (or a partlof or a partlot), in a book- or report-class document, is preceded and followed by a `\cleardoublepage` (which acts like `\clearpage` in a one-side document), and has a page style of `empty`. Since version #32, you can modify this behaviour by redefining the commands of table 1.10, whose meaning is obvious. A feature defined as `\empty` does nothing.

`\mtcsetfeature` 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` 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 styles¹⁴



The default commands for part-level mini-tables page styles are defined as being `\thispagestyle{empty}`, because in document classes defining the `\chapter` command, 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 comes 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 example¹⁵:

```
\mtcsetfeature{parttoc}{before}%
  {\cleardoublepage}
\mtcsetfeature{parttoc}{thispagestyle}%
  {\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 `parttocs`, the `myheadings` page style for the following pages of the `parttocs`, 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).

¹⁴This remark is taken and adapted from a draft of the second edition of the JMPL [13], by Benjamin BAYART, where he comments the `minitoc` package.

¹⁵This 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.

1.5.4.1 Before version #23 (1994/11/08)

`\firstpartis` To make the minitoc package work with such documents, you must insert the command:
`\firstchapteris`
`\firstsectionis` `\firstchapteris{⟨N⟩}`

before the `\dominitoc` and analogous commands. $\langle N \rangle$ is the number of the first chapter. This command *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.

1.5.4.2 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 for TOC, LOF, LOT, Bibliography and Index

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 [210] by Peter R. WILSON (this package is 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 `scrbook`, `scrreprt` or `scrartcl.cls` KOMA-Script classes [164] with some options (`liststotoc`, `liststotocnumbered`, `bibtotoc`, `bibtotocnumbered`, and `idxstotoc`). The solutions are the same.

`\mtcaddchapter` So you must add an `\mtcaddchapter` command, *without argument*, after each of the
`\tableofcontents` involved commands `\tableofcontents`, `\listoffigures`, and `\listoftables`.
`\listoffigures`
`\listoftables`

`\adjustmtc` For the bibliography, you should add a `\adjustmtc` command after the `\bibliography`
`\bibliography` command.

`\printindex` For the glossary, it is a bit more complicated, you should add the following commands just after the `\printglossary` command:

```

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

```

But this can be done by:

```
\mtcfixglossary[chapter|section|part]
```

where the optional argument is the level for the glossary entry in the TOC. By default, if `\chapter` is defined, the `chapter` level is used, else the `section` level. If neither `\chapter` or `\section` are defined, the `part` level will be used if `\part` is defined; else an error is reported. It is *recommended* that you check the result and, if necessary, you adjust the optional argument.



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

```

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

```

But this can be done by:

```
\mtcfixindex[chapter|section|part]
```

where the optional argument is the level for the index entry in the TOC. By default, if `\chapter` is defined, the `chapter` level is used, else the `section` level. If neither `\chapter` or `\section` are defined, the `part` level will be used if `\part` is defined; else an error is reported. It is *recommended* that you check the result and, if necessary, you adjust the optional argument.



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

And proceed with extreme care, tracking in the `.log` file the insertion of `.mtc<N>` files (and siblings). They are some examples in the `add.tex` (see section 4.3 on page 96) and `addsec.tex` (see section 4.4 on page 101) files distributed with `minitoc`. The `addsec.tex` example shows how much that problem is difficult.

1.6 The notoccite option

`\cite` This option loads the `notoccite` package [8] (by Donald ARSENEAU). It avoids problems with `\cite` commands in sectioning commands or captions: if you then run `BIBTEX` using the `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 use something like:

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

The `nolistfiles` package option is the default (no list).

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 `.log` file and emits a warning. The hints written in the `.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 problems and potential problems are currently detected:



- `\part` • Alteration of some of the following commands¹⁶: `\part`, `\@part`, `\@spart`, `\@part`, `\chapter`, `\@chapter`, `\@schapter`, `\section`, `\@sect`, and `\@ssect`. Note that the `hyperref` (see section 2.17 on page 59) package alters these commands at `\begin{document}`, hence this problem might be reported if you use this package, but these alterations seem harmless. Note that the `hyperref` must be loaded *before* `minitoc`.
 - `\@chapter`
 - `\chapter`
 - `\@schapter`
 - `\schapter`
 - `\@schapter` • Presence of the following packages or classes, which need some precautions: `amsbook` (class), `memoir` (class), `appendix`, `placeins` (beware to its options and its release date (2005/04/18 at least)), `scrbook` (class), `scrreprt` (class), `scrartcl` (class), `tocbibind`, and `tocloft`.
 - `\section`
 - `\@sect`
 - `\@ssect`
- Presence of the following packages or classes, which are unfortunately incompatible with the `minitoc` package: `amsart` (class), `amsproc` (class), `alphanum`, `jura` (class), `titlesec`, and `titletoc`¹⁷.
- `\parttoc` • Usage of `\parttoc` without calling `\doparttoc, ...`, usage of `\sectlot` without calling `\dosectlot`. Or the reverse.
 - `\doparttoc`
 - `\sectlot`
 - `\dosectlot` Usage of `\parttoc` without calling `\[fake]tableofcontents, ...`, usage of `\sectlot` without calling `\[fake]listoftables`.
- `\tableofcontents` Usage of `\sectlof` and/or `\sectlot` without using the `insection` package option of `minitoc` (or the `placeins` package without its `section` option).
 - `\listoftables`
 - `\sectlof`
 - `\sectlot` • If you are using short extensions (because of your operating system or the `shorttext` package option, see section 1.9 on the next page) and go beyond the limit of 99 parts, chapters or sections, the `hints` package option displays a warning.
- `abstract` • If the `abstract` package [208] (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 is detected and you should add a `\mtcaddchapter[]` or a `\mtcaddsection[]` command after your `abstract` environment.
 - `\mtcaddchapter`
 - `\mtcaddsection`
- If the `sectsty` package [153] (by Rowland McDONNELL) is used, it must be loaded *before* the `minitoc` package.
 - If the `varsects` package [189] (by Daniel TAUPIN[†]) is used, it must be loaded *before* the `minitoc` package.
 - If the `fncychap` package [146] (by Ulf A. LINDGREN) is used, it must be loaded *before* the `minitoc` package.
 - If the `quotchap` package [193] (by Karsten TINNEFELD) is used, it must be loaded *before* the `minitoc` package.
 - If the `romannum` package [215] (by Peter R. WILSON) is used, it must be loaded *before* the `minitoc` package.

¹⁶The 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.

¹⁷The `titlesec` package redefines the sectionning commands in a way completely alien to the standard \LaTeX way; hence `minitoc` and `titlesec-titletoc` are fundamentally incompatible, and it is very sad.

- If the `sheaders` package [148] (by Maurizio LORETI) is used, it must be loaded *before* the `minitoc` package.
 - If the `alnumsec` package [128] (by Frank KÜSTER) is used, it must be loaded *before* the `minitoc` package.
 - If the `captcont` package [85] (by Steven Douglas COCHRAN) is used, it must be loaded *before* the `minitoc` package.
 - If one of the `caption` [186], `caption2`¹⁸ [185], (both written by Axel SOMMERFELDT), `ccaption` [212] (written by Peter R. WILSON), or `mcaption` [113] (written by Stephan HENNIG), packages is used, it must be loaded *before* the `minitoc` package.
 - If you try to insert empty mini-tables, the `hints` option gives a global warning (except if you used also the `nocheckfiles` option, see section 1.3.3 on page 28).
- `\firstpartis` • If you use one of the obsolete commands (`\firstpartis`, `\firstchapteris`, `\firstchapteris` or `\firstsectionis`), a warning is issued for each use, of course, but also a `\firstsectionis` 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 `minitoc` package determines dynamically the type of extensions available and will use it. All other modifications will be done automatically. The `.mtc⟨N⟩` extensions will become `.M⟨N⟩`, where `⟨N⟩` is the absolute chapter number. The extensions `.mlf⟨N⟩` and `.mlt⟨N⟩` become `.F⟨N⟩` and `.T⟨N⟩`. The `.ptc⟨N⟩` extensions become `.P⟨N⟩`, where `⟨N⟩` is the absolute part number. The extensions `.plf⟨N⟩` and `.plt⟨N⟩` become `.G⟨N⟩` and `.U⟨N⟩`. The `.stc⟨N⟩` extensions become `.S⟨N⟩`, where `⟨N⟩` is the absolute section number. The extensions `.slf⟨N⟩` and `.slt⟨N⟩` become `.H⟨N⟩` and `.V⟨N⟩`. All these extensions are listed in table 1.11 on the next 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 tragical. The `hints` option (section 1.8 on page 49) will report such situations. See also section 2.5 on page 56.

1.10 Why several L^AT_EX 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 L^AT_EX

¹⁸This package is obsolete; now use the `caption` package.

Table 1.11: Extensions of the auxiliary files

mini-table	long extensions (UNIX, etc.)	short extensions (MS-DOS, etc.)
parttoc	.ptc $\langle N \rangle$.P $\langle N \rangle$
partlof	.plf $\langle N \rangle$.G $\langle N \rangle$
partlot	.plt $\langle N \rangle$.U $\langle N \rangle$
minitoc	.mtc $\langle N \rangle$.M $\langle N \rangle$
minilof	.mlf $\langle N \rangle$.F $\langle N \rangle$
minilot	.mlt $\langle N \rangle$.T $\langle N \rangle$
secttoc	.stc $\langle N \rangle$.S $\langle N \rangle$
sectlof	.slf $\langle N \rangle$.H $\langle N \rangle$
sectlot	.slt $\langle N \rangle$.V $\langle N \rangle$

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 55).

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 .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	54
2.1	How to avoid a page break near the rules before and after the mini-table?	54
2.2	How about implementing others layouts for a mini-table?	55
2.3	Two consecutive backslashes in a contents line make an error	55
2.4	Reordering chapters makes havoc	55
2.5	Extensions for the names of auxiliary files	56
2.6	Playing with the chapter number	56
2.7	Supported document classes	56
2.8	Compatibility with L ^A T _E X versions	56
2.9	Other mini-tables	57
2.10	Why so many auxiliary files?	57
2.11	How to do mini-tables at levels other than chapter?	58
2.12	Incompatibility with L ^A T _E X2.09	58
2.13	Documents resetting the chapter number at each part	59
2.14	The mini-tables have too much spaced lines	59
2.15	The secttocs are wrong	59
2.16	How to remove the lines of dots?	59
2.17	How to use the hyperref package with minitoc?	59
2.18	Problem while upgrading minitoc	60
2.19	A local table of contents for the set of appendices	60
2.20	Use with the appendix package	60
2.21	Use with the tocloft package	61
2.22	Use with the memoir class	62
2.23	There are too many commands for fonts, titles, and depths	63
2.24	Compatibility with the A _M S document classes	63
2.25	How to hide some entries from the main table of contents	63
2.26	How to define your own .mld file?	67
2.27	Use with the abstract package	67
2.28	Use with the sectsty package	67
2.29	Strange alignment in the minitocs	67
2.30	Some useful precautions with starred sectionning commands	68
2.31	Use with packages for captions	69

2.32	Bad interaction minitoc/hyperref/memoir	69
2.33	Use with the varsects package	69
2.34	Initial font settings	69
2.35	Use with the KOMA-Script classes	71
2.36	Use with the jura class or the alphanum package	71
2.37	The .mld files and the babel package	72
2.38	Use with the fncychap package	72
2.39	Use with the quotchap package	72
2.40	Use with the romannum package	72
2.41	Use with the sfheaders package	72
2.42	Use with the alnumsec package	73
2.43	Use with the captcont package	73
2.44	Vertical space (gap) before and after parttocs, partlofs, and partlots titles	73
2.45	Vertical spacing before the bottom rule of a minitable	74

Figures

2.1	Three compilations for minitoc	55
-----	--	----

Tables

2.1	Kernings before minitable bottom rules	74
-----	--	----

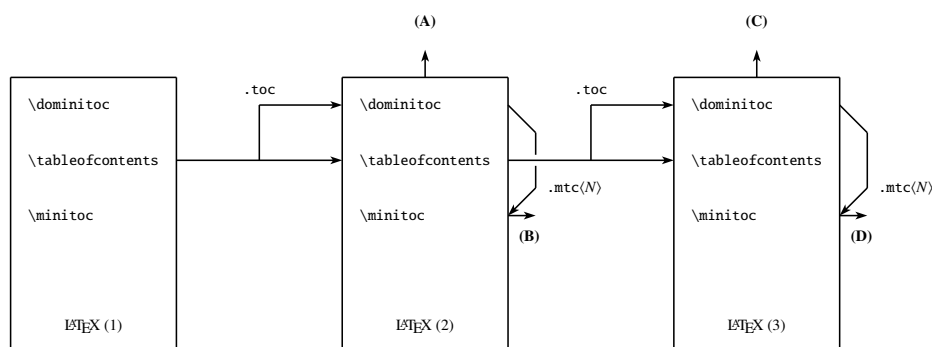
2.0 Introduction

Here is a list of problems and frequently asked questions about `minitoc.sty`. If the version has a number less than 47, please upgrade to version #47. 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 49), 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 resort (please join a minimal but complete example reproducing the problem).

2.1 How to avoid a page break near the rules before and after the mini-table?

`\enlargethispage` This problem seemed solved since version #8, but version #12 adds better fixes. You may have to make some final tuning with `\enlargethispage`. See the L^AT_EX manual [132]. The `needspace` package [206] may also be useful.



(A) `\tableofcontents` produces a table of contents, which is likely inaccurate.

(B) `\minitoc` produces minitocs, which are likely inaccurate.

(C) `\tableofcontents` produces a table of contents, which is accurate.

(D) `\minitoc` produces minitocs, which are accurate.

Figure 2.1: Three compilations for minitoc

2.2 How about implementing others layouts for a mini-table?

Suggestions are welcome, but look at the section 1.4.13 on page 42.

2.3 Two consecutive backslashes in a contents line make an error

Use `\protect\linebreak`.

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. 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 `shorttext`. 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

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

The mechanism would break. It is better to add `\chapter` commands, to create empty (but numbered in a legal way) chapters. Since version #10, the `minitoc` package works with appendices. Version #19 allows to begin with a chapter other than number 1. And look at “Special Entries for TOC, LOF, LOT, Bibliography and Index”, section 1.5.5 on page 47.

The same remark applies to the `part` and `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”. There are “parttocs” if the document class defines the `\part` command. Note that classes like “letter”, which have not the classical sectionning structure, cannot be supported. Classes using sectionning commands with other names are not supported¹. See also section 2.24 on page 63.

2.8 Compatibility with L^AT_EX versions

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

¹ This would be very difficult: any user can create new sectionning 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 sectionning commands and on the syntax of these commands.


```
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`. First, “`minibbl`” is another problem, strongly related to the `BibTeX`’s dealing with `.aux` files. Look at the `chapterbib` [7], `bibunits` [104], `multibib` [105], and `bibtopic` [12] packages. Version #13 has implemented basic `minilofs` and `minilots`. `Minibbbs` are not the aim of this package².

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 28) 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 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 49.

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

² See <http://www.tex.ac.uk/cgi-bin/textfaq2html?label=multibib>

be useful to add to it a cleaning feature (which should be optional, to allow debugging). The table 1.11 on page 52 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 script [16] (written in Python) provided by Emmanuel BEFFARA:

<http://rubber.sourceforge.net/>
<http://www.pps.jussieu.fr/~beffara/soft/rubber/>

2.11 How to do 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 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!).

	part	chapter	section
book	*	*	
report	*	*	
article	*		*

2.12 Incompatibility with L^AT_EX2.09

`\protect` The more recent version of L^AT_EX2_ε adds `\protect` before `\contentsline` in the
`\contentsline` `.toc`, `.lof` and `.lot` files. The version #17 of `minitoc` attempts to be compatible with L^AT_EX2_ε and L^AT_EX2.09. This will be the *last* version usable with L^AT_EX2.09. Versions #18 and later are L^AT_EX2_ε specific, and no more compatible with L^AT_EX2.09, 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 *absolute* number.

2.14 The mini-tables have too much spaced lines

From version #29, you can have tight mini-tables with the `tight` option, and with the `k-tight` option for the KOMA-Script classes [164] (since version #43).

2.15 The secttocs are wrong

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

2.16 How to remove 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 1.4.13 on page 42.

2.17 How to use the hyperref package with minitoc?



Since version #31, `minitoc` works correctly with the powerful `hyperref` package [179], thanks to Heiko OBERDIEK, who used the work of Bernd JAEHNE and Didier VERNA. Hence the `minitoc-hyper.sty` package 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 `fminitoc.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 L^AT_EX 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 table of contents for the set of appendices

<code>\doparttoc</code> <code>\tableofcontents</code> <code>\appendix</code> <code>\part</code> <code>\parttoc</code> <code>\addtocontents</code> <code>\protect</code> <code>\setcounter</code> <code>\chapter</code> <code>\partbegin</code>	<p>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 <code>\part</code> subdivision of the document and ask for a table of contents at the <code>\part</code> level:</p> <pre> \doparttoc % after \begin{document} . . . \tableofcontents . . . \appendix \part{Appendices} % create a part level subdivision \parttoc % create a local table of contents % To suppress the appendix part in the main toc \addtocontents{toc}{\protect\setcounter{tocdepth}{-1}} \chapter{First appendix} . . . % 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} </pre>
---	---

See also section 2.25 on page 63.

2.20 Use with the appendix package

<code>appendices</code> <code>\addcontentsline</code> <code>\adjustmtc</code> <code>\adjuststc</code>	<p>If you use the <code>appendix</code> package [209] (by Peter R. WILSON), you will observe a serious problem with minitocs in the <code>appendices</code> environment (and after it): they do not match with their respective appendices. In fact, the environment opening <code>\begin{appendices}</code> hides a <code>\addcontentsline</code> 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 <code>\adjustmtc</code> or <code>\adjuststc</code> command (depending if the appendices are at the chapter or section level) after <i>each</i> <code>\begin{appendices}</code> command. An other solution is to add the following commands in the preamble <i>after</i> the loading of the <code>appendix</code> package:</p>
--	---

```
\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 preceeding chapter or section.

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

2.21 Use with the `tocloft` package

`\mtcsetfont` (This answer is given in the documentation of the `tocloft` package [207].) The `tocloft` (by Peter R. WILSON) and `minitoc` packages have an unfortunate interaction³, which fortunately 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}
```

³ Discovered by Lyndon DUDGING.

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 [213, 214] offers basically the fonctionnalités 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 page 60, 1.5.5 on page 47, and 2.21 on the page before 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` 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 (`\mtcsetttitlefont` 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 `memoir` class (or of the `tocloft` package), you must disable the `memoir/tocloft` font commands. This is done by the following commands:

```
\let\cftpfont\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
```

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

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

`\mtcsettitle` Since version #42, the `\mtcsettitle` command is available. You do not need anymore to know `\mtctitle`, `\slttitle`, etc.

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

2.24 Compatibility with the \mathcal{AMS} document classes

`\mtcaddchapter` The `amsart.cls` and `amsproc.cls` document classes are unfortunately incompatible with `minitoc`. The `amsbook.cls` document class requires the insertion of commands if you want a list of figures and/or a list of tables:

```
\listoffigures
\mtcaddchapter % added
\listoftables
\mtcaddchapter % added
```

2.25 How to hide some entries from the main table of contents

`mtchideinmaintoc` It is a problem similar to that of section 2.19 on page 60. An example is having a local
`mtchideinmainlof` table of contents for a chapter (`\minitoc`) whose entries should not appear in the main
`mtchideinmainlot` 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 `apx.tex` example file:

```

1 (*apx)
2 \documentclass[oneside]{book}
3 \usepackage{lipsum} % provides filling text
4 \usepackage{tocbibind} % adds some entries in the main TOC.
5 \usepackage[tight,listfiles]{minitoc}
6 \setcounter{minitocdepth}{3}
7 \setcounter{parttocdepth}{3}
8 \begin{document}
9 \doparttoc
10 \dominitoc
11 \tableofcontents
12 \mtcaddchapter % because tocbibind adds a chapter entry in the TOC
13 \chapter{First}
14 \minitoc
15 First chapter
16 \section{First section}
17 \lipsum[1]
18 \section{Second section}
19 \lipsum[2]
20 \chapter{Second}
21 \minitoc
22 Second
23 \section{First section of second chapter}
24 \lipsum[3]
25 \section{Second section of second chapter}
26 \lipsum[4]
27 \appendix % begins the appendices
28 \addcontentsline{toc}{part}{Appendices} % adds a part entry in the TOC
29 \adjustptc % fixes the parttoc counter ptc
30 \parttoc % adds a partial toc for the appendices
31 \begin{mtchideinmaintoc}[-1] % hides the details of the
32 % % appendices in the main TOC, but chapter-level
33 % % entries would be still visible if you use 0
34 % % in place of -1 as optional argument.
35 \chapter{First appendix}
36 \minitoc
37 First appendix
38 \section{First section}
39 \lipsum[5]
40 \section{Second section}
41 \lipsum[6]
42 \chapter{Second appendix}
43 \minitoc
44 Second appendix
45 \section{First section of second appendix}
46 \lipsum[7]

```



```

47 \section{Second section of second appendix}
48 \lipsum[8]
49 \end{mtchideinmaintoc} % end of hiding
50 \end{document}
51 </apx>

```

Of course, the environments `mtchideinmainlof` and `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 `\chapter` command), else the `restore` command might be inserted to early into the `.toc`, `.lof` or `.lot` file. There is an example file (`hide1.tex`):

```

52 < *hide1>
53 \documentclass{report}
54 \usepackage%
55   [tight,listfiles]{minitoc}
56 \begin{document}
57 \dominilof
58 \dominilot
59 \listoffigures
60 \listoftables
61 \chapter{First}
62 \minilof
63 \minilot
64 \begin{figure}
65 \caption{AAAA1}
66 \end{figure}
67 \begin{figure}
68 \caption{AAAA2}
69 \end{figure}
70 \begin{table}
71 \caption{TAAAA1}
72 \end{table}
73 \begin{table}
74 \caption{TAAAA2}
75 \end{table}
76 \chapter{Second}
77 \minilof
78 \minilot
79 %%-----

```

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.

```

80 \begin{mtchideinmainlof}
81 \begin{mtchideinmainlot}
82 \begin{figure}
83 \caption{BBBB1}

```

```

84 \end{figure}
85 \begin{figure}
86 \caption{BBBB2}
87 \end{figure}
88 \begin{table}
89 \caption{TBBBB1}
90 \end{table}
91 \begin{table}
92 \caption{TBBBB2}
93 \end{table}
94 \chapter{Third}

```

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.

```

95 \end{mtchideinmainlot}
96 \end{mtchideinmainlof}
97 %%-----
98 \minilof
99 \minilot
100 \begin{figure}
101 \caption{CCCC1}
102 \end{figure}
103 \begin{figure}
104 \caption{CCCC2}
105 \end{figure}
106 \begin{table}
107 \caption{TCCCC1}
108 \end{table}
109 \begin{table}
110 \caption{TCCCC2}
111 \end{table}
112 \end{document}
113 </hide1>

```

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 (`hide2.tex`):

```

114 <{*hide2}
115 \documentclass{report}
116 \usepackage%
117   [tight,listfiles]{minitoc}
118 \begin{document}
119 \dominilof
120 \dominilot
121 \listoffigures
122 \listoftables
123 \chapter{First}
124 \minilof
125 \minilot
126 \begin{figure}
127 \caption{AAAA1}
128 \end{figure}
129 \begin{figure}
130 \caption{AAAA2}
131 \end{figure}
132 \begin{table}
133 \caption{TAAAA1}
134 \end{table}
135 \begin{table}
136 \caption{TAAAA2}
137 \end{table}
138 \chapter{Second}
139 \minilof
140 \minilot

```

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 command forms: a command is inserted *before* the caption of the first “hidden” entry.

```

141 \begin{figure}
142 \mtchideinmainlof
143 \caption{BBBB1}
144 \end{figure}
145 \begin{figure}
146 \caption{BBBB2}
147 \endmtchideinmainlof

```

```

148 \end{figure}
149 \begin{table}
150 \mtchideinmainlot
151 \caption{TBBBB1}
152 \end{table}

```

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 command forms: a command is inserted *after* the caption of the *last* “hidden” entry.

```

153 \begin{table}
154 \caption{TBBBB2}
155 \endmtchideinmainlot
156 \end{table}
157 \chapter{Third}
158 \minilof
159 \minilot
160 \begin{figure}
161 \caption{CCCC1}
162 \end{figure}
163 \begin{figure}
164 \caption{CCCC2}
165 \end{figure}
166 \begin{table}
167 \caption{TCCCC1}
168 \end{table}
169 \begin{table}
170 \caption{TCCCC2}
171 \end{table}
172 \end{document}
173 </hide2>

```

This method, recommended while more delicate to apply, is more reliable in delimiting the hiding domain: it solves the problem of the asynchronicity between the writing of floats and the writing of the normal text.

2.26 How to define your own .mld file?

`\mtcsettitle` First, you should not directly modify one of the distributed .mld and .mlo files.
`\mtcselectlanguage` 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 $\langle language \rangle$ [.mld-.mlo] pairs of language definition files.

2.27 Use with the abstract package

`abstract` If the abstract package [208] (by Peter R. WILSON), is used with its `addtotoc` option,
`\mtcaddchapter` a “Abstract” entry is added to the table of contents, as a starred chapter if the document
`\mtcaddsection` class defines `\chapter`, else as a starred section. This problem is detected by the `hints`
`\chapter` 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 [153] (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, subsections 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 `\l@part`, `\l@chapter`, `\l@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:

```
\newcommand*\l@section{\@dottedtocline{1}{1.5em}{2.3em}}
\newcommand*\l@subsection{\@dottedtocline{2}{3.8em}{3.2em}}
```

```

\newcommand*\l@subsubsection{\@dottedtocline{3}{7.0em}{4.1em}}
\newcommand*\l@paragraph{\@dottedtocline{4}{10em}{5em}}
\newcommand*\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, the dimensions (second and third arguments) are given in “em” units, and this unit depends of 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 35), 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 [207] to change the indentation, with font independent units. But then see section 2.21 on page 61.
- Use the same font for the section and subsection entries in the minitocs, using the `\mtcsetfont` command (see section 1.4.9 on page 40) or redefining the `\mtcSfont`, `\mtcSSfont`, `\mtcSSSfont`, `\mtcPfont` and `\mtcSPfont` commands (see table 1.6 on page 35), or similar.

2.30 Some 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` [186], `caption2` [185]⁴, (both written by Axel SOMMERFELDT), `ccaption` [212] (by Peter R. WILSON), or `mcaption` [113] (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` [179] packages are used in a document of class `memoir` [213, 214], the chapter header “Chapter” does not appear on the first page of the chapter.

This problem is fixed in version #44 of `minitoc`.

2.33 Use with the `varsects` package

If the `varsects` package [189] (by Daniel TAUPIN⁺) is used, it must be loaded *before* the `minitoc` package, because it alters (redefines) the sectionning 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 `parttoc`s as an example, there is a `\ptcfont` font-command which is used for two purposes⁵:

⁴ This package is obsolete; now use a recent version of the `caption` package.

⁵ The same remarks apply to the other mini-tables.

- 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\plfSfont\ptcfont      % (subfigures)
\let\pltfont\ptcfont       % (tables)
\let\pltSfont\ptcfont      % (subtables)

```

to define some default fonts. But this sequence is executed only once. If you alter `\ptcfont`, the modification is not applied to these font commands. The command `\ptcfont` is invoked at the beginning of each `parttoc`. `\ptcCfont` is invoked for each chapter entry in a `parttoc` (`\ptcSfont` for each section entry, etc.). So `\ptcfont` can be used to define some global characteristics for the fonts in the `parttocs`, while `\ptcCfont` (etc.) can be used to customize the fonts for each level of entries.

Note that if you say:

```

\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)

```

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 are 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\ptcSSfont{\lowlevelsfont}
\def\ptcSSSfont{\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 parttocs are used in the examples above; but, of course, the situation is the same for minitocs and secttocs. `\highlevelsfont` and `\lowlevelsfont` are macro names that you can choose, they are not part of the minitoc package.

2.35 Use with the KOMA-Script classes

If a KOMA-Script class [164], 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 47. 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 sectionning 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 [30, 31], you can automatize the loading of the .mld file by adding some code in the preamble of your document, like this:

```
\AtBeginDocument{%  
  \addto\captionslanguageI{\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, by example).

2.38 Use with the fncychap package

If the fncychap package [146] (by Ulf A. LINDGREN) is used, it must be loaded *before* the minitoc package, because it alters (redefines) the sectionning commands. Of course, the hints option detects this problem.

2.39 Use with the quotchap package

If the quotchap package [193] (by Karsten TINNEFELD) is used, it must be loaded *before* the minitoc package, because it alters (redefines) the sectionning commands. Of course, the hints option detects this problem.

2.40 Use with the romannum package

If the romannum package [215] (by Peter R. WILSON) is used, it must be loaded *before* the minitoc package, because it alters (redefines) the numbering of the sectionning commands. Of course, the hints option detects this problem.

2.41 Use with the sfheaders package

If the sfheaders package [148] (by Maurizio LORETI) is used, it must be loaded *before* the minitoc package, because it alters (redefines) the sectionning commands. Of course, the hints option detects this problem.

2.42 Use with the `alnumsec` package

If the `alnumsec` package [128] (by Frank KÜSTER) is used, it must be loaded *before* the `minitoc` package, because it alters (redefines) the numbering of the sectioning commands. Of course, the `hints` option detects this problem.

2.43 Use with the `captcont` package

If the `captcont` package [85] (by Steven Douglas COCHRAN) is used, it must be loaded *before* the `minitoc` package, because it alters (redefines) the caption commands. Of course, the `hints` option detects this problem.

2.44 Vertical space (gap) before and after `parttocs`, `partlofs`, and `partlots` titles



These vertical gaps were hard-coded like for the chapter heads in the `book` and `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 `\mtcgapbeforeheads` and `\mtcgapafterheads`, with these default values. These commands apply globally to `parttocs`, `partlofs` and `partlots`. These are *commands*, *not* dimensions, so they must be modified via `\renewcommand` (*not* via `\setlength`). An example of use is given in the `gaps.tex` document file:

```
174 (*gaps)
175 \documentclass[a4paper,oneside,12pt]{book}
176 \usepackage{vruler} % to have a vertical graduation to note positions
177 \usepackage{txfonts}
178 \usepackage[english2,tight,listfiles]{minitoc}
179 \begin{document}
180 \setvruler[1cm][0][10][3][0][0pt][0pt][0pt][] % with vruler package
181 \doparttoc
182 \faketableofcontents
183 \part{First part}
```

A normal `parttoc`, with the normal gaps before and after it.

```
184 \parttoc
185 \chapter{First chapter of first part} \chapter{Second chapter of first part}
186 \part{Second part}
```

We set large gaps. Note the new position of the `parttoc`.

```
187 \renewcommand{\mtcgapbeforeheads}{100pt}
```

```

188 \renewcommand{\mtcgapafterheads}{80pt}
189 \parttoc
190 \chapter{First chapter of second part} \chapter{Second chapter of second part}
191 \part{Third part}

```

We set small gaps. Note the new position of the parttoc.

```

192 \renewcommand{\mtcgapbeforeheads}{20pt}
193 \renewcommand{\mtcgapafterheads}{10pt}
194 \parttoc
195 \chapter{First chapter of third part} \chapter{Second chapter of third part}
196 \end{document}
197 </gaps>

```

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 mini-table. The values should depend of the line spacing and of the font size. They are defined by macros that you can adjust by redefining them via `\renewcommand`. The (empirical) default values are given in table 2.1.

Table 2.1: Kernings before minitable bottom rules

Command	Default value
<code>\kernafterparttoc</code>	<code>\kern-1.\baselineskip\kern.5ex</code>
<code>\kernafterpartlof</code>	<code>\kern-1.\baselineskip\kern.5ex</code>
<code>\kernafterpartlot</code>	<code>\kern-1.\baselineskip\kern.5ex</code>
<code>\kernafterminitoc</code>	<code>\kern-.5\baselineskip\kern.5ex</code>
<code>\kernafterminilof</code>	<code>\kern-1.\baselineskip\kern0.ex</code>
<code>\kernafterminilot</code>	<code>\kern-1.\baselineskip\kern0.ex</code>
<code>\kernaftersecttoc</code>	<code>\kern-1.\baselineskip\kern.5ex</code>
<code>\kernaftersectlof</code>	<code>\kern-1.\baselineskip\kern.5ex</code>
<code>\kernaftersectlot</code>	<code>\kern-1.\baselineskip\kern.5ex</code>

Chapter 3

Memento

Tables	
3.1	Package options 75
3.2	General commands 76
3.3	Commands: part level 77
3.4	Commands: chapter level 78
3.5	Commands: section level 79
3.6	Commands for horizontal rules 80
3.7	Commands for page numbers 80
3.8	Commands for mini-tables features 81
3.9	Preparation and insertion commands 81
3.10	Adjustment commands 82
3.11	Obsolete commands 82

Table 3.1: Package options

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

Language options are listed in table 1.7 on page 36. Default: english.

Table 3.2: General commands

Command	Meaning
<code>\faketableofcontents</code>	Replaces <code>\tableofcontents</code> if you want mini-tables of contents but no main table of contents.
<code>\fakelistoffigures</code>	Replaces <code>\listoffigures</code> if you want mini-lists of figures but no main list of figures.
<code>\fakelistoftables</code>	Replaces <code>\listoftables</code> if you want mini-lists of tables but no main list of tables.
<code>\mtcselectlanguage{language}</code>	Loads <code>language.mld</code> to select a language for mini-tables titles.
<code>\mtcsetdepth{mini-table}{depth}</code>	Changes the depth for some mini-tables.
<code>\mtcsetfeature{mini-table}{before after pagestyle}{commands}</code>	Modifies the features for a mini-table.
<code>\mtcsetfont{mini-table}{sectioning-level}{font commands}</code>	Redefines a minitoc font command.
<code>\mtcsetformat{mini-table}{dotinterval pagenumwidth tocrightmargin}{value}</code>	Changes the layout of some mini-tables.
<code>\mtcsetpagenumbers{mini-table *}{on off}</code>	Activates/inhibits page numbers in some or all mini-tables.
<code>\mtcsetrules{mini-table *}{on off}</code>	Activates/inhibits horizontal rules in some or all mini-tables.
<code>\mtcsettitle{mini-table}{title string}</code>	Changes the title for some mini-tables.
<code>\mtcsettitlefont{mini-table}{font commands}</code>	Changes the font of the title for some mini-tables.
<code>\mtcskip</code>	To add a vertical skip between the mini-tables.
<code>\mtcskipamount</code>	Length of <code>\mtcskip</code> . Default: <code>\bigskipamount</code> .
<code>\tightmtcfalse</code>	Loose mini-tables. Default.
<code>\tightmtctrue</code>	Tight mini-tables.
<code>\ktightmtcfalse</code>	Loose mini-tables. Default. (KOMA-Script classes).
<code>\ktightmtctrue</code>	Tight mini-tables. (KOMA-Script classes).
<code>\undottedmtcfalse</code>	Dotted lines in mini-tables (from entry to page number). Default.
<code>\undottedmtctrue</code>	No dotted lines in mini-tables (from entry to page number).

Table 3.3: Commands: part level

Command	Meaning
<code>\doparttoc[x]</code>	Before <code>\[fake]tableofcontents</code> if you use <code>\parttoc*</code> .
<code>\dopartlof[x]</code>	Before <code>\[fake]listoffigures</code> if you use <code>\partlof*</code> .
<code>\dopartlot[x]</code>	Before <code>\[fake]listoftables</code> if you use <code>\partlot*</code> .
<code>\parttoc[x]</code>	After each <code>\part</code> command for which a parttoc is needed*.
<code>\partlof[x]</code>	After each <code>\part</code> command for which a partlof is needed*.
<code>\partlot[x]</code>	After each <code>\part</code> command for which a partlot is needed*.
<code>\setcounter{parttocdepth}{depth}</code>	Depth of the following parttocs. Analog to <code>tocdepth</code> . Default: 2. Has no action on partlofs and partlots.
<i>or:</i>	
<code>\mtcsetdepth{parttoc partlof partlot}{depth}</code>	Idem, but can also act on partlofs and partlots.
<code>\ptcindent</code>	Left/right indentation of a partial table. Default: 24pt.
<code>\ptcfont</code>	Font command for parttoc. Default: <code>\small\rmfamily\upshape\mdseries</code> (article) or: <code>\normalsize\rmfamily\upshape\mdseries</code> (book, report).
<code>\ptcCfont</code>	Font command for parttoc, chapter entries. Default: <code>\normalsize\rmfamily\upshape\bfseries</code> .
<code>\ptcSfont</code>	Font command for parttoc, section entries. Default: <code>\small\rmfamily\upshape\bfseries</code> (article) or: <code>\small\rmfamily\upshape\bfseries</code> (book, report).
<code>\ptcSSfont</code>	Font command for parttoc, subsection entries**.
<code>\ptcSSSfont</code>	Font command for parttoc, subsubsection entries**.
<code>\ptcPfont</code>	Font command for parttoc, paragraph entries**.
<code>\ptcSPfont</code>	Font command for parttoc, subparagraph entries**.
<code>\plffont</code>	Font for partlof. Default: <code>\small\rmfamily\upshape\mdseries</code> .
<code>\plfSfont</code>	Font for partlof (subfigures). Default: <code>\small\rmfamily\upshape\mdseries</code> .
<code>\pltfont</code>	Font for partlot. Default: <code>\small\rmfamily\upshape\mdseries</code> .
<code>\pltSfont</code>	Font for partlot (subtables). Default: <code>\small\rmfamily\upshape\mdseries</code> .
<code>\ptctitle</code>	Title of parttocs. Default: Table of Contents.
<code>\plftitle</code>	Title of partlofs. Default: List of Figures.
<code>\plttitle</code>	Title of partlots. Default: List of Tables.
<code>\ptifont</code>	Font for partXXX titles. Default: <code>\Large\rmfamily\upshape\bfseries</code> (article) or: <code>\LARGE\rmfamily\upshape\bfseries</code> (book, report).
<code>\mtcgapbeforeheads</code>	Vertical gap before part-level mini-tables titles. Default: 50pt
<code>\mtcgapafterheads</code>	Vertical gap after part-level mini-tables titles. Default: 40pt

*: `[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: 1 for left (default), c for centered, r for right, n or e for no title.

** : defaults like `\ptcfont`.

Table 3.4: Commands: chapter level

Command	Meaning
<code>\dominitoc[x]</code>	Before <code>\[fake]tableofcontents</code> if you use <code>\minitoc*</code> .
<code>\dominilof[x]</code>	Before <code>\[fake]listoffigures</code> if you use <code>\minilof*</code> .
<code>\dominilot[x]</code>	Before <code>\[fake]listoftables</code> if you use <code>\minilot*</code> .
<code>\minitoc[x]</code>	After each <code>\chapter</code> command for which a minitoc is needed*.
<code>\minilof[x]</code>	After each <code>\chapter</code> command for which a minilof is needed*.
<code>\minilot[x]</code>	After each <code>\chapter</code> command for which a minilot is needed*.
<code>\setcounter{minitocdepth}{depth}</code>	Depth of the following minitocs. Analog to <code>tocdepth</code> . Default: 2. Has no action on minilofs and minilots.
<i>or:</i>	
<code>\mtcsetdepth{minitoc minilof minilot}{depth}</code>	Idem, but can also act on minilofs and minilots.
<code>\mtcindent</code>	Left/right indentation of a mini-table. Default: 24pt.
<code>\mtcfont</code>	Font command for minitoc. Default: <code>\small\rmfamily\upshape\mdseries</code> .
<code>\mtcSfont</code>	Font command for minitoc, section entries. Default: <code>\small\rmfamily\upshape\bfseries</code> .
<code>\mtcSSfont</code>	Font command for minitoc, subsection entries**.
<code>\mtcSSSfont</code>	Font command for minitoc, subsubsection entries**.
<code>\mtcPfont</code>	Font command for minitoc, paragraph entries**.
<code>\mtcSPfont</code>	Font command for minitoc, subparagraph entries**.
<code>\mlffont</code>	Font for minilof. Default: <code>\small\rmfamily\upshape\mdseries</code> .
<code>\mlfSfont</code>	Font for minilof (subfigures). Default: <code>\small\rmfamily\upshape\mdseries</code> .
<code>\mltfont</code>	Font for minilot. Default: <code>\small\rmfamily\upshape\mdseries</code> .
<code>\mltSfont</code>	Font for minilot (subtables). Default: <code>\small\rmfamily\upshape\mdseries</code> .
<code>\mtctitle</code>	Title of minitocs. Default: Contents.
<code>\mlftitle</code>	Title of minilofs. Default: Figures.
<code>\mlttitle</code>	Title of minilots. Default: Tables.
<code>\mtifont</code>	Font for miniXXX titles. Default: <code>\large\rmfamily\upshape\bfseries</code> .

*: `[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: 1 for left (default), c for centered, r for right, n or e for no title.

** : defaults like `\mtcfont`.

Table 3.5: Commands: section level

Command	Meaning
<code>\dosecttoc[x]</code>	Before <code>\[fake]tableofcontents</code> if you use <code>\secttoc*</code> .
<code>\dosectlof[x]</code>	Before <code>\[fake]listoffigures</code> if you use <code>\sectlof*</code> .
<code>\dosectlot[x]</code>	Before <code>\[fake]listoftables</code> if you use <code>\sectlot*</code> .
<code>\secttoc[x]</code>	After each <code>\section</code> command for which a <code>secttoc</code> is needed*.
<code>\sectlof[x]</code>	After each <code>\section</code> command for which a <code>sectlof</code> is needed*.
<code>\sectlot[x]</code>	After each <code>\section</code> command for which a <code>sectlot</code> is needed*.
<code>\setcounter{secttocdepth}{depth}</code>	Depth of the following <code>secttocs</code> . Analog to <code>tocdepth</code> . Default: 2. Has no action on <code>sectlofs</code> and <code>sectlots</code> .
<i>or:</i>	
<code>\mtcsetdepth{secttoc sectlof sectlot}{depth}</code>	Idem, but can also act on <code>sectlofs</code> and <code>sectlots</code> .
<code>\stcindent</code>	Left/right indentation of a mini-table. Default: 24pt.
<code>\stcfont</code>	Font command for <code>secttoc</code> . Default: <code>\small\rmfamily\upshape\mdseries</code> .
<code>\stcSSfont</code>	Font command for <code>secttoc</code> , subsection entries**.
<code>\stcSSSfont</code>	Font command for <code>secttoc</code> , subsubsection entries**.
<code>\stcPfont</code>	Font command for <code>secttoc</code> , paragraph entries**.
<code>\mtcSPfont</code>	Font command for <code>secttoc</code> , subparagraph entries**.
<code>\slffont</code>	Font for <code>sectlof</code> . Default: <code>\small\rmfamily\upshape\mdseries</code> .
<code>\slfSfont</code>	Font for <code>sectlof</code> (subfigures). Default: <code>\small\rmfamily\upshape\mdseries</code> .
<code>\sltfont</code>	Font for <code>sectlot</code> . Default: <code>\small\rmfamily\upshape\mdseries</code> .
<code>\sltSfont</code>	Font for <code>sectlot</code> (subtables). Default: <code>\small\rmfamily\upshape\mdseries</code> .
<code>\stctitle</code>	Title of <code>secttocs</code> . Default: Contents.
<code>\slftitle</code>	Title of <code>sectlofs</code> . Default: Figures.
<code>\slttitle</code>	Title of <code>sectlots</code> . Default: Tables.
<code>\stifont</code>	Font for <code>sectXXX</code> titles. Default: <code>\large\rmfamily\upshape\bfseries</code> .

*: `[x]` is an optional argument to set the position of the title; the setting is local for the `\sectXXX` commands, global for the `\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 `\stcfont`.

Table 3.6: Commands for horizontal rules

Command	Meaning
<code>\[no]ptcrule</code>	Activates or inhibits rules in parttocs.
<code>\[no]mtcrule</code>	Activates or inhibits rules in minitocs.
<code>\[no]stcrule</code>	Activates or inhibits rules in secttocs.
<code>\[no]plfrule</code>	Activates or inhibits rules in partlofs.
<code>\[no]mlfrule</code>	Activates or inhibits rules in minilofs.
<code>\[no]slfrule</code>	Activates or inhibits rules in sectlofs.
<code>\[no]pltrule</code>	Activates or inhibits rules in partlots.
<code>\[no]mltrule</code>	Activates or inhibits rules in minilots.
<code>\[no]sltrule</code>	Activates or inhibits rules in sectlots.
<code>\mtcsetrules{mini-table *}{on off}</code>	Activates/inhibits horizontal rules in some or all mini-tables.
<code>\kernafterparttoc</code>	Vertical kerning between a parttoc and its bottom rule.
<code>\kernafterpartlof</code>	Vertical kerning between a partlof and its bottom rule.
<code>\kernafterpartlot</code>	Vertical kerning between a partlot and its bottom rule.
<code>\kernafterminitoc</code>	Vertical kerning between a minitoc and its bottom rule.
<code>\kernafterminilof</code>	Vertical kerning between a minilof and its bottom rule.
<code>\kernafterminilot</code>	Vertical kerning between a minilot and its bottom rule.
<code>\kernaftersecttoc</code>	Vertical kerning between a secttoc and its bottom rule.
<code>\kernaftersectlof</code>	Vertical kerning between a sectlof and its bottom rule.
<code>\kernaftersectlot</code>	Vertical kerning between a sectlot and its bottom rule.

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

Table 3.7: Commands for page numbers

Command	Meaning
<code>\[no]ptcpagenumbers</code>	Activates or inhibits page numbers in parttocs.
<code>\[no]plfpagenumbers</code>	Activates or inhibits page numbers in partlofs.
<code>\[no]pltpagenumbers</code>	Activates or inhibits page numbers in partlots.
<code>\[no]mtcpagenumbers</code>	Activates or inhibits page numbers in minitocs.
<code>\[no]mlfpagenumbers</code>	Activates or inhibits page numbers in minilofs.
<code>\[no]mltpagenumbers</code>	Activates or inhibits page numbers in minilots.
<code>\[no]stcpagenumbers</code>	Activates or inhibits page numbers in secttocs.
<code>\[no]slfpagenumbers</code>	Activates or inhibits page numbers in sectlofs.
<code>\[no]sltpagenumbers</code>	Activates or inhibits page numbers in sectlots.
<code>\mtcsetpagenumbers{mini-table *}{on off}</code>	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

Command	Default	Meaning
<code>\beforeparttoc</code>	<code>\cleardoublepage</code>	Action before a parttoc.
<code>\beforepartlof</code>	<code>\cleardoublepage</code>	Action before a partlof.
<code>\beforepartlot</code>	<code>\cleardoublepage</code>	Action before a partlot.
<code>\afterparttoc</code>	<code>\cleardoublepage</code>	Action after a parttoc.
<code>\afterpartlof</code>	<code>\cleardoublepage</code>	Action after a partlof.
<code>\afterpartlot</code>	<code>\cleardoublepage</code>	Action after a partlot.
<code>\thispageparttocstyle</code>	<code>\thispagestyle{empty}</code>	Page style for a parttoc.
<code>\thispagepartlofstyle</code>	<code>\thispagestyle{empty}</code>	Page style for a partlof.
<code>\thispagepartlotstyle</code>	<code>\thispagestyle{empty}</code>	Page style for a partlot.
<code>\beforeminitoc</code>	<code>\empty</code>	Action before a minitoc.
<code>\beforeminilof</code>	<code>\empty</code>	Action before a minilof.
<code>\beforeminilot</code>	<code>\empty</code>	Action before a minilot.
<code>\afterminitoc</code>	<code>\empty</code>	Action after a minitoc.
<code>\afterminilof</code>	<code>\empty</code>	Action after a minilof.
<code>\afterminilot</code>	<code>\empty</code>	Action after a minilot.
<code>\thispageminitocstyle</code>	<code>\empty</code>	Page style for a minitoc.
<code>\thispageminilofstyle</code>	<code>\empty</code>	Page style for a minilof.
<code>\thispageminilotstyle</code>	<code>\empty</code>	Page style for a minilot.
<code>\beforesecttoc</code>	<code>\empty</code>	Action before a secttoc.
<code>\beforesectlof</code>	<code>\empty</code>	Action before a sectlof.
<code>\beforesectlot</code>	<code>\empty</code>	Action before a sectlot.
<code>\aftersecttoc</code>	<code>\empty</code>	Action after a secttoc.
<code>\aftersectlof</code>	<code>\empty</code>	Action after a sectlof.
<code>\aftersectlot</code>	<code>\empty</code>	Action after a sectlot.
<code>\thispagesecttocstyle</code>	<code>\empty</code>	Page style for a secttoc.
<code>\thispagesectlofstyle</code>	<code>\empty</code>	Page style for a sectlof.
<code>\thispagesectlotstyle</code>	<code>\empty</code>	Page style for a sectlot.
<code>\mtcsetfeature{mini-table}{before after pagestyle}{commands}</code> Modifies the features for a mini-table.		

Table 3.9: Preparation and insertion commands

Type	Phase	Level		
		part	chapter	section
table of contents	preparation	<code>\doparttoc[p]</code>	<code>\dominitoc[p]</code>	<code>\dosecttoc[p]</code>
	insertion	<code>\parttoc[p]</code>	<code>\minitoc[p]</code>	<code>\secttoc[p]</code>
list of figures	preparation	<code>\dopartlof[p]</code>	<code>\dominilof[p]</code>	<code>\dosectlof[p]</code>
	insertion	<code>\partlof[p]</code>	<code>\minilof[p]</code>	<code>\sectlof[p]</code>
list of tables	preparation	<code>\dopartlot[p]</code>	<code>\dominilot[p]</code>	<code>\dosectlot[p]</code>
	insertion	<code>\partlot[p]</code>	<code>\minilot[p]</code>	<code>\sectlot[p]</code>
all	preparation	<code>\mtcprepare[p]</code>		

Each of these commands accepts one optional argument p , which specifies the position of the title of the mini-table. This argument 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

Command	Meaning
<code>\adjustptc[n]</code>	Adjusts (increments) the parttoc counter <code>ptc</code> by n .
<code>\adjustmtc[n]</code>	Adjusts (increments) the minitoc counter <code>mtc</code> by n .
<code>\adjuststc[n]</code>	Adjusts (increments) the secttoc counter <code>stc</code> by n .
<code>\decrementptc</code>	Adjusts (decrements by 1) the parttoc counter <code>ptc</code> .
<code>\decrementmtc</code>	Adjusts (decrements by 1) the minitoc counter <code>mtc</code> .
<code>\decrementstc</code>	Adjusts (decrements by 1) the secttoc counter <code>stc</code> .
<code>\incrementptc</code>	Adjusts (increments by 1) the parttoc counter <code>ptc</code> .
<code>\incrementmtc</code>	Adjusts (increments by 1) the minitoc counter <code>mtc</code> .
<code>\incrementstc</code>	Adjusts (increments by 1) the secttoc counter <code>stc</code> .
<code>\mtcaddpart[title]</code>	Adds the title of a <code>\part*</code> in the ToC.
<code>\mtcaddchapter[title]</code>	Adds the title of a <code>\chapter*</code> in the ToC.
<code>\mtcaddsection[title]</code>	Adds the title of a <code>\section*</code> in the ToC.
<code>\mtcfixglossary[chapter section part]</code>	Adjusts the entry for glossary in the ToC.
<code>\mtcfixindex[chapter section part]</code>	Adjusts the entry for index in the ToC.
<code>\begin{mtchideinmaintoc}[depth] ... \end{mtchideinmaintoc}</code>	Environment to hide entries in the main table of contents.
<code>\begin{mtchideinmainlof}[depth] ... \end{mtchideinmainlof}</code>	Environment to hide entries in the main list of figures.
<code>\mtchideinmainlof[depth] ... \endmtchideinmainlof</code>	Pair of commands to hide entries in the main list of figures.
<code>\begin{mtchideinmainlot}[depth] ... \end{mtchideinmainlot}</code>	Environment to hide entries in the main list of tables.
<code>\mtchideinmainlot[depth] ... \endmtchideinmainlot</code>	Pair of commands to hide entries in the main list of tables.

Table 3.11: Obsolete commands

Command	Meaning
<code>\firstpartis{N}</code>	N is the number of the first part.
<code>\firstchapteris{N}</code>	N is the number of the first chapter.
<code>\firstsectionis{N}</code>	N is the number of the first section.

Chapter 4

Examples of documents

Contents

4.1	The minitoc-ex.tex document file	84
4.2	The mini-art.tex document file	90
4.3	The add.tex document file	96
4.4	The addsec.tex document file	101
4.5	The second.tex document file	106
4.6	The app-mem.tex document file	108
4.7	The mem.tex document file	108
4.8	The mem1.tex document file	109
4.9	The fo1.tex document file	110
4.10	The fo2.tex document file	111
4.11	The scr.tex document file	113
4.12	The subf.tex document file	114
4.13	The tsfc.tex document file	116
4.14	The tbi.tex document file	118
4.15	The 2c.tex document file	119
4.16	The bo.tex document file	120
4.17	The hop.tex document file	125
4.18	The cri.tex document file	126
4.19	The livre.tex document file	127
4.20	The ch0.tex document file	131
4.21	The tlc.tex document file	132
4.22	The mu.tex document file	133
4.23	The hir.tex document file	134
4.24	The hia.tex document file	135

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 specific example files has been shown earlier: `hide1.tex` on page 65, `hide2.tex` on page 66, and `gaps.tex` on page 73.

4.1 The minitoc-ex.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:

```

198 <*minitoc - ex>
199 %%%%%%%%%%%%%%% A example file (differs from previous versions)
200 %% minitoc-ex.tex
201 %% This file contains a set of tests for the minitoc.sty version #47
202 %% package file. You can alter most of parameters to test.
203 %% book/report (\chapter must be defined).
204 %% You can use a copy of this file to play with minitoc
205 %% commands and parameters.
206 \documentclass[12pt,a4paper]{report} % the report class uses less pages
207 %%\documentclass[12pt,a4paper]{book}
208 \usepackage{lipsum} % provides filling text
209 %%\usepackage{hyperref} % if used, load it BEFORE minitoc
210 %%\usepackage{mtcoff}
211 \usepackage[tight,listfiles]{minitoc} % tight option make shorter mini-tables
212 \setcounter{secnumdepth}{5} % depth of numbering of sectionning commands
213 \setcounter{tocdepth}{3} % depth of table of contents
214 \setlength{\mtcindent}{24pt} % indentation of minitocs, default
215 \renewcommand{\mtcfont}{\small\rm} % font for minitocs, default
216 \renewcommand{\mtcSfont}{\small\bf} % font for minitocs, sections, default
217 %%\renewcommand{\mtcSSfont}{\small\sf} % font for minitocs, subsections
218 %% you can make experiments with \mtcSSfont, \mtcPfont and \mtcSPfont
219 %% but it is "fontomania"...
220 \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 of the numbering of the chapters (it is absolute).

```

221 %% TEST: uncomment the 3 next lines to test
222 %% resetting chapter number in each part
223 %%\makeatletter
224 %%\@addtoreset{chapter}{part}
225 %%\makeatother
226 %% 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):

```

227 \begin{document}
228 \mtcpagenumbers
229 \noptcrule
230 %% \nomtcrule % suppresses minitoc rules
231 %% \nomtcpagenumbers % suppresses minitoc page numbers

```

```

232 %% \nomlfpagenumbers           % ----- minilof ----
233 %% \nomltpagenumbers          % ----- minilot ----

```

The preparation commands, with their optional argument if necessary:

```

234 \dominitoc
235 \dominilof[c]                % centers title of minilof's
236 \dominilot
237 \doparttoc                    % test of parttoc/partlof stuff
238 \dopartlof                   % added in version #15
239 \dopartlot                    % added in version #15

```

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

```

240 \tableofcontents             % or \faketableofcontents
241 \listoffigures               % or \fakelistoffigures
242 \fakelistoftables            % or \listoftables

```

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

```

243 %%\addtocounter{chapter}{-1} % to begin with Chapter 0

```

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

```

244 \part{First Part}
245 \parttoc
246 \partlof[r]
247 \partlot[r]

```

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

```

248 \twocolumn\sloppy           % the minitoc in twocolumn layout is ugly,
249 \chapter{AAAAA}              % a chapter with a lot of sections
250 \minitoc[r]                  % minitoc title on the right
251 \lipsum[1]
252 \section{S1}
253 \lipsum[2]
254 \section{S2}
255 \lipsum[3]
256 \section{S3}
257 \lipsum[4]

```

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

```

258 \section*{S4}
259 \addcontentsline{toc}{section}{\protect\numberline{}{S4}}
260 \lipsum[5]
261 \section{S5}
262 \lipsum[6]
263 \section{S6}

```

```
264 \lipsum[6]
265 \section{S7}
266 \lipsum[7]
267 \section{S8}
268 \lipsum[9]
269 \section{S9}
270 \lipsum[10]
271 \section{S10}
272 \lipsum[11]
273 \section{S11}
274 \lipsum[12]
275 \section{S12}
276 \lipsum[13]
277 \section{S13}
278 \lipsum[14]
279 \section{S14}
280 \lipsum[15]
281 \section{S15}
282 \lipsum[16]
283 \section{S16}
284 \lipsum[17]
285 \section{S17}
286 \lipsum[18]
287 \section{S18}
288 \lipsum[19]
289 \section{S19}
290 \lipsum[20]
291 \section{S20}
292 \lipsum[21]
293 \section{S21}
294 \lipsum[22]
295 \section{S22}
296 \lipsum[23]
297 \section{S23}
298 \lipsum[24]
299 \section{S24}
300 \lipsum[25]
301 \section{S25}
302 \lipsum[26]
303 \section{S26}
304 \lipsum[27]
305 \section{S27}
306 \lipsum[28]
307 \section{S28}
308 \lipsum[29]
309 \section{S29}
310 \lipsum[30]
311 \section{S30}
312 \lipsum[31]
313 \subsection{SS1}
314 \lipsum[32]
315 \section{S31}
316 \lipsum[33]
```

We return to one column mode. A new chapter, with a minitoc, a minilof and a minilot:

```

317 \onecolumn\fussy           % back to one column
318 \chapter{BBBBB}
319 \minitoc
320 \mtcskip                   % put some skip here
321 \minilof                   % a minilof
322 \mtcskip                   % put some skip here
323 \minilot                   % a minilot
324 \lipsum[34]
325 \section{T1}
326 \lipsum[35]
327 \begin{figure}[t]          % tests compatibility with floating bodies
328 \setlength{\unitlength}{1mm}
329 \begin{picture}(100,50)
330 \end{picture}
331 \caption{F1}               % (tables are similar)
332 \end{figure}
333 \begin{table}[b]           % tests compatibility with floating bodies
334 \setlength{\unitlength}{1mm}
335 \begin{picture}(100,50)
336 \end{picture}
337 \caption{T1}               % (tables are similar)
338 \end{table}
339 \clearpage
340 \subsection{tt1}{TT1}      % tests optional arg. of a sectionning command
341 \lipsum[36]
342 \subsubsection{TTT1}
343 \lipsum[37]
344 \paragraph{TTTT1}
345 \lipsum[38]
346 \begin{figure}
347 \setlength{\unitlength}{1mm}
348 \begin{picture}(100,50)
349 \end{picture}
350 \caption[f2]{F2}           % tests optional arg. of a caption
351 \end{figure}
352 \section{T2}
353 \lipsum[39]

```

A starred chapter with an entry added in the TOC; all subaltern sectionning commands must also be starred.

```

354 \chapter*{CCCCC}          % tests a pseudo-chapter; could have a minitoc.
355 \addstarredchapter{CCCCC}
356 \lipsum[40]
357 \section*{U1}
358 \addcontentsline{toc}{section}{U1}
359 \lipsum[41]
360 \subsection*{UU1}
361 \addcontentsline{toc}{subsection}{UU1}
362 \lipsum[42]
363 \subsubsection*{UUU1}

```

```

364 \addcontentsline{toc}{subsubsection}{UUU1}
365 \lipsum[43]
366 \paragraph*{UUUU1}
367 \addcontentsline{toc}{paragraph}{UUUU1}
368 \lipsum[44]
369 \section*{U2}
370 \addcontentsline{toc}{section}{U2}
371 \lipsum[45]
372 \part{Second Part}
373 \parttoc
374 \partlof[c]

```

This chapter has no minitoc, but if you uncomment `\minitoc`, the minitoc will appear.

```

375 %%                                % the following chapter should have no minitoc,
376 \chapter{DDDDD}                  % but if you uncomment \minitoc,
377 %%\minitoc                        % the minitoc appears
378 \lipsum[46]
379 \section{V1}
380 \lipsum[47]
381 \subsection{VV1}
382 \lipsum[48]
383 \subsubsection{VVV1}
384 \lipsum[48]
385 \paragraph{VVVV1}
386 \lipsum[49]
387 \begin{figure}[t]                % tests compatibility with floating bodies
388 \setlength{\unitlength}{1mm}
389 \begin{picture}(100,50)
390 \end{picture}
391 \caption{F3}                      % (I have not tested tables, but it is similar)
392 \end{figure}
393 \lipsum[50]
394 \section{V2}
395 \lipsum[51]

```

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

```

396 \chapter{EEEE}                  % this chapter should have a minitoc
397 {%                               % left brace, see below
398 \setcounter{minitocdepth}{3}    % depth of mini table of contents;
399                                % try with different values.
400 \minitoc
401 }                               % right brace
402 %% this pair of braces is used to keep local the change
403 %% on minitocdepth.
404 \lipsum[52]
405 \section{W1}                     % with the given depth
406 \lipsum[53]
407 \subsection{WW1}
408 \lipsum[54]
409 \subsubsection{WWW1}
410 \lipsum[55]

```



```

411 \begin{figure}[t]           % tests compatibility with floating bodies
412 \setlength{\unitlength}{1mm}
413 \begin{picture}(100,50)
414 \end{picture}
415 \caption{F4}               % (I have not tested tables here, but it is similar)
416 \end{figure}
417 \lipsum[56]
418 \paragraph{WWW1}
419 \lipsum[57]
420 \subparagraph{WWWWW1}
421 \lipsum[58]
422 \section{W2}
423 \lipsum[59]

```

Here, we encounter a classical problem: to make a local table of contents for a set of appendices, and to hide these entries in the main table of contents. First, we create a part, with its parttoc:

```

424 \appendix
425 \part{Appendices}
426 \parttoc

```

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

```

427 \begin{mtchideinmaintoc}[-1]
428 \chapter{Comments}
429 \minitoc
430 \section{C1}
431 \lipsum[60]
432 \section{C2}
433 \lipsum[61]
434 \section{C3}
435 \lipsum[62]
436 \begin{figure}[t]           % tests compatibility with floating bodies
437 \setlength{\unitlength}{1mm}
438 \begin{picture}(100,50)
439 \end{picture}
440 \caption{F5}               % (I have not tested tables, but it is similar)
441 \end{figure}
442 \section{C4}
443 \chapter{Evolution}
444 \minitoc
445 \minilof %Empty => invisible
446 \minilot %Empty => invisible
447 \section{D1}
448 \lipsum[62]
449 \section{D2}
450 \lipsum[63]
451 \section{D3}
452 \lipsum[64]
453 \section{D4}

```

We terminate the part by adding a marker in the TOC file, then we terminate the `mtchideinmaintoc` environment:

```

454 %% this line closes the omitted part
455 \addtocontents{toc}{\protect\partbegin}
456 %% this line restore the depth in the main TOC
457 \end{mtchideinmaintoc}
458 \lipsum[65]
459 \end{document}
460 </minitoc - ex>

```

4.2 The mini-art.tex document file

This is a basic document using the `minitoc` package. It contains sections but no chapters, so it must use an `article`-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:

```

461 (*mini - art)
462 %%%%%%%%%%%%%%% A example file (may differ from previous versions)
463 %% mini-art.tex
464 %% This file contains a set of tests for the minitoc.sty version #47
465 %% style option file. You can alter most of parameters to test.
466 %% article (\section must be defined)
467 \documentclass[12pt,a4paper]{article}
468 \usepackage{lipsum} % provides filling text
469 %%\usepackage{hyperref} % If used, load it BEFORE minitoc
470 \usepackage[tight,insection,listfiles]{minitoc}
471 \setcounter{secnumdepth}{5} % depth of numbering of sectioning commands
472 \setcounter{tocdepth}{3} % depth of table of contents
473 \setlength{\stcindent}{24pt} % indentation of secttocs, default
474 %% % font for secttocs, default
475 \renewcommand{\stcfont}{\small\rmfamily\upshape\mdseries}%
476 %% % font for secttocs, subsections
477 %%\renewcommand{\stcSSfont}{\small\sffamily}%
478 %% you can make experiments with \stcSSfont, \stcPfont and \stcSPfont
479 %% but it is "fontomania"...
480 \raggedbottom % 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 of the numbering of the sections (it is absolute).

```

481 %% TEST: uncomment the 3 next lines to test
482 %% resetting section number in each part
483 %%\makeatletter
484 %%\@addtoreset{section}{part}
485 %%\makeatother

```

```
486 %%% 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):

```
487 \begin{document}
```

The preparation commands, with their optional argument if necessary:

```
488 \dosecttoc
489 \dosectlof[c]           % center title of sectlof's
490 \dosectlot
491 \doparttoc              % test of parttoc/partlof stuff
492 \dopartlof              % added in version #15
493 \dopartlot              % added in version #15
```

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

```
494 \faketableofcontents    % or \faketableofcontents
495 \fakelistoffigures      % to check compatibility
496 \fakelistoftables       % to check compatibility
```

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

```
497 \part{First Part}
498 \parttoc
499 \partlof[r]
500 \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).

```
501 \twocolumn\sloppy       % the secttoc in twocolumn layout is ugly,
502                          % but works. Ideas to make it better?
503 \section{AAAAA}          % a section with a lot of sections
504 \secttoc[r]              % secttoc title on the right
505 \mtcskip \sectlof %ADDED
506 \lipsum[1]
507 \subsection{S1}
508 \lipsum[2]
509 \subsection{S2}
510 \lipsum[3]
511 \subsection{S3}
512 \lipsum[4]
513 \subsection*{S4}
514 %%\addcontentsline{toc}{starsubsection}{*S4*}
515 \lipsum[5]
516 \subsection{S5}
```

```
517 \lipsum[6]
518 \subsection{S6}
519 \lipsum[7]
520 \subsection{S7}
521 \lipsum[8]
522 \subsection{S8}
523 \lipsum[9]
524 \subsection{S9}
525 \lipsum[10]
526 \subsection{S10}
527 \lipsum[11]
528 \subsection{S11}
529 \lipsum[12]
530 \subsection{S12}
531 \lipsum[13]
532 \subsection{S13}
533 \lipsum[14]
534 \subsection{S14}
535 \lipsum[15]
536 \subsection{S15}
537 \lipsum[16]
538 \subsection{S16}
539 \lipsum[17]
540 \subsection{S17}
541 \lipsum[18]
542 \subsection{S18}
543 \lipsum[19]
544 \subsection{S19}
545 \lipsum[20]
546 \subsection{S20}
547 \lipsum[21]
548 \subsection{S21}
549 \lipsum[22]
550 \subsection{S22}
551 \lipsum[23]
552 \subsection{S23}
553 \lipsum[24]
554 \subsection{S24}
555 \lipsum[25]
556 \subsection{S25}
557 \lipsum[26]
558 \subsection{S26}
559 \lipsum[27]
560 \subsection{S27}
561 \lipsum[28]
562 \subsection{S28}
563 \lipsum[29]
564 \subsection{S29}
565 \lipsum[30]
566 \subsection{S30}
567 \lipsum[31]
```

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.

```

568 \onecolumn\fussy                                % back to one column
569 \section{BBBBB}
570 \secttoc
571 \mtcskip                                          % put some skip here
572 \sectlof                                         % a sectlof
573 \lipsum[32]
574 \subsection{T1}
575 \lipsum[33]
576 \begin{figure}[t]                                % tests compatibility with floating bodies
577 \setlength{\unitlength}{1mm}
578 \begin{picture}(100,50)
579 \end{picture}
580 \caption{F1}                                     % (I have not tested tables, but it is similar)
581 \end{figure}
582 \FloatBarrier
583 \subsubsection[tt1]{TT1}                         % tests optional arg. of a sectioning command
584 \lipsum[34]
585 \paragraph{TTT1}
586 \lipsum[35]
587 \subparagraph{TTTT1}
588 \lipsum[36]
589 \begin{figure}[t]
590 \setlength{\unitlength}{1mm}
591 \begin{picture}(100,50)
592 \end{picture}
593 \caption[f2]{F2}                                 % tests optional arg. of a caption
594 \end{figure}
595 \FloatBarrier
596 \subsection{T2}
597 \lipsum[37]
598 \section*{CCCCC}                                % tests a pseudo-section. should have no secttoc
599 %%\addstarredsection{CCCCC}
600 \mtcaddsection[CCCCC]
601 \secttoc \mtcskip \sectlof %ADDED
602 \lipsum[38]
603 \subsection{U1}
604 \lipsum[39]
605 \subsubsection{UU1}
606 \lipsum[40]
607 \paragraph{UUU1}
608 \lipsum[41]
609 \subparagraph{UUUU1}
610 \lipsum[42]
611 \subsection{U2}
612 \lipsum[43]
613 \part{Second Part}
614 \parttoc
615 \partlof[c]
616 \partlot
617 %%                                              % the following section should have no secttoc,

```

```

618 \section{DDDDD}           % but if you uncomment \secttoc,
619 %%\secttoc
620 \mtcskip \sectlof %ADDED
621 %                         % the secttoc appears
622 \lipsum[44]
623 \subsection{V1}
624 \lipsum[45]
625 \subsubsection{VV1}
626 \lipsum[46]
627 \paragraph{VVV1}
628 \lipsum[47]
629 \subparagraph{VVVV1}
630 \lipsum[48]
631 \begin{figure}[t]         % tests compatibility with floating bodies
632 \setlength{\unitlength}{1mm}
633 \begin{picture}(100,50)
634 \end{picture}
635 \caption{F3}             % (I have not tested tables, but it is similar)
636 \end{figure}
637 \FloatBarrier
638 \lipsum[49]
639 \subsection{V2}
640 \lipsum[50]

```

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

```

641 \section{EEEEEE}         % this section should have a secttoc
642 {%                       % left brace, see below
643 \setcounter{secttocdepth}{3} % depth of sect table of contents;
644                             % try with different values.
645 \secttoc
646 \mtcskip \sectlof %ADDED
647 }                         % right brace
648 %% this pair of braces is used to keep local the change
649 %% on secttocdepth.
650 \lipsum[51]
651 \subsection{W1}          % with the given depth
652 \lipsum[52]
653 \subsubsection{WW1}
654 \lipsum[53]
655 \paragraph{WWW1}
656 \lipsum[54]
657 \begin{figure}[t]         % tests compatibility with floating bodies
658 \setlength{\unitlength}{1mm}
659 \begin{picture}(100,50)
660 \end{picture}
661 \caption{F4}             % (I have not tested tables, but it is similar)
662 \end{figure}
663 \FloatBarrier
664 bla bla bla bla bla bla bla bla bla bla
665 \subparagraph{WWWW1}
666 \lipsum[55]
667 \subsection{W2}

```

```

668 \lipsum[56]
669 \chapter*{}
670 \part{Appendices}
671 \parttoc
672 \mtcskip
673 \partlof
674 \mtcskip
675 \partlot
676 \FloatBarrier
677 \appendix
678 \section{Comments}
679 \lipsum[57]
680 \secttoc
681 \mtcskip \sectlof %ADDED
682 \subsection{C1}
683 \lipsum[58]
684 \subsection{C2}
685 \lipsum[59]
686 \subsection{C3}
687 \lipsum[60]
688 \begin{figure}[hb] % tests compatibility with floating bodies
689 \setlength{\unitlength}{1mm}
690 \begin{picture}(100,50)
691 \end{picture}
692 \caption{F5} % (I have not tested tables, but it is similar)
693 \end{figure}
694 \FloatBarrier
695 \subsection{C4}
696 \lipsum[61]
697 \FloatBarrier
698 \section{Evolution}
699 \secttoc
700 \sectlof % empty
701 \sectlot % empty
702 \lipsum[62]
703 \subsection{D1}
704 \lipsum[63]
705 \subsection{D2}
706 \lipsum[64]
707 \subsection{D3}
708 \lipsum[65]
709 \subsection{D4}
710 \lipsum[66]
711 \end{document}
712 </mini-art>

```

4.3 The add.tex document file

This document shows how to add special entries in the table of contents, and the interaction with the tocbibind package [210].

```

713 < *add>
714 \documentclass[12pt,a4paper]{report}
715 %%\documentclass[12pt,a4paper]{book}
716 \usepackage{url}
717 \usepackage{tocbibind}
718 \usepackage{makeidx}
719 \makeatletter
720 \newif\ifscan@allowed
721 \scan@allowedtrue
722 \makeatother
723 \def\dotfil{\leaders\hbox to.6em{\hss .\hss}\hfil}%
724 \def\pfill{\unskip~\dotfill\penalty500\strut\nobreak
725           \dotfil~\ignorespaces}%

```

Load the minitoc package, or mtcoff.

```

726 \usepackage[tight,hints,listfiles]{minitoc}
727 %%\usepackage{mtcoff}

728 \makeindex
729 \begin{document}

```

We call the mini-table preparation commands:

```

730 \dominitoc
731 \dominilof
732 \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 47:

```

733 \tableofcontents
734 \mtcaddchapter
735 \listoffigures
736 \mtcaddchapter
737 \listoftables
738 \mtcaddchapter

```

For a chapter, we want a minitoc, a minilof and a minitoc:

```

739 \chapter{First chapter}\index{chapter!normal}
740 \minitoc
741 \mtcskip
742 \minilof

```



```
743 \mtcskip
744 \minilot
```

Then the text of the chapter, with sections, figures and tables:

```
745 \section{First section}
746
747 \begin{figure}[tp]
748 \caption{First figure}
749 \end{figure}
750 \begin{table}[tp]
751 \caption{First table}
752 \end{table}
753
754 \section{Second section}
755 A small nice citation:\
756 \index{small}\index{citation}\index{nice}\index{A}\index{a}%
757 \index{and}\index{bird}\index{But}\index{cannot}%
758 \index{claim}\index{great}\index{he}\index{I}%
759 \index{imagine}\index{it}\index{know}\index{land}%
760 \index{on}\index{once}\index{that}\index{to}\index{tree}%
761 \index{would}\index{yes}%
762 \textsf{A bird cannot land once on a great tree and claim to know it.}
763 But I imagine that he would, yes.}\
764 \hbox{}\hfill
765 Iain~M.~\textsc{Banks}~(1993),~\textsl{Against~a~dark~background.}%
766 \index{Iain}\index{Banks}\index{Against}\index{dark}\index{background}
767
768 \begin{figure}[tp]
769 \caption{Second figure}
770 \end{figure}
771 \begin{table}
772 \caption{Second table}
773 \end{table}
```

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 `add.tex`. For each case, look at the Table of Contents and the involved chapter.

```
774 \chapter*{Second chapter, starred}
775 \index{chapter!starred}
776 %% UNCOMMENT ONE AND ONLY ONE OF THE 3 FOLLOWING LINES
777 \mtcaddchapter[Second chapter, starred] % OK
778 %%\mtcaddchapter[~] % produce a (strange) correct result. OK
779 %%\addcontentsline{toc}{xchapter}{}
780 %%%
781 %%\mtcaddchapter[] % BAD SOLUTION
782 %%\mtcaddchapter % BAD SOLUTION
783 %%%
784 \index{tests}
785
```

```

786 This is a starred chapter; you can test here variations on
787 the \verb|\mtcaddchapter| command. Just uncomment one (and
788 only one) of the \verb|\mtcaddchapter| commands after
789 \verb|\chapter*| in the source code of \texttt{add.tex}.
790 For each case, look at the \index{Table of Contents}Table of Contents
791 and at this chapter.
792 \index{a}\index{added}\index{after}\index{also}\index{and}%
793 \index{at}\index{can}\index{case}\index{chapter}\index{code}%
794 \index{command}\index{commands}\index{Contents}\index{each}%
795 \index{entries}\index{For}\index{here}\index{I}\index{in}%
796 \index{index}\index{is}\index{Just}\index{just}\index{look}%
797 \index{lot}\index{of}\index{on}\index{one}\index{only}%
798 \index{source}\index{starred}\index{Table}\index{test}%
799 \index{the}\index{This}\index{this}\index{to}\index{uncomment}%
800 \index{variations}\index{you}%
801 I also added a lot of index entries, just to test.
802
803 \chapter{Third chapter}
804 \index{chapter!normal}
805 \minitoc
806 \mtcskip
807 \minilof
808 \mtcskip
809 \minilot
810 \section{Third section}
811
812 \begin{figure}
813 \caption{Third figure}
814 \end{figure}
815 \begin{table}
816 \caption{Third table}
817 \end{table}
818
819 \section{Fourth section}
820
821 \begin{figure}
822 \caption{Fourth figure}
823 \end{figure}
824 \begin{table}
825 \caption{Fourth table}
826 \end{table}

```

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

```

827 \nocite*
828 \def\noopsort#1{\relax}
829 \bibliographystyle{plain}
830 \bibliography{add}
831 \adjustmtc

```

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

```

832 \printindex
833 %%\mtcfixindex % use this OR the 3 following lines
834 \addcontentsline{lof}{xchapter}{}
835 \addcontentsline{lot}{xchapter}{}
836 \mtcaddchapter
837 %%
838
839 \appendix
840 \chapter{App.~1}
841 \index{chapter!appendix}
842 \minitoc
843 \mtcskip
844 \minilof
845 \mtcskip
846 \minilot
847 \section{Fifth section}
848
849 \begin{figure}
850 \caption{Fifth figure}
851 \end{figure}
852 \begin{table}
853 \caption{Fifth table}
854 \end{table}
855
856 \section{Sixth section}
857
858 \begin{figure}
859 \caption{Sixth figure}
860 \end{figure}
861 \begin{table}
862 \caption{Sixth table}
863 \end{table}

```

The next chapter asks for a minitoc, a minilof and a minilot, but contains no tables; hence the `minitoc` package will give some warnings.

```

864 \chapter{App.~2}
865 \index{chapter!appendix}
866 %% contains no tables but asks for a minilot! No minilot printed.
867 \minitoc
868 \mtcskip
869 \minilof
870 \mtcskip
871 \minilot
872 \section{Seventh section}
873 \begin{figure}
874 \caption{Seventh figure}
875 \end{figure}
876 \begin{figure}

```

```

877 \caption{Eighth figure}
878 \end{figure}
879
880 \section{Eighth section}
881
882 \begin{figure}
883 \caption{Ninth figure}
884 \end{figure}
885 \begin{figure}
886 \caption{Eleventh figure}
887 \end{figure}
888
889 \end{document}
890 \</add>

```

And we need also its small bibliographic data base:

```

891 \<{*addbib}
892 @MISC{minitoc,
893     TITLE="The {\texttt{minitoc}} package",
894     MONTH=aug,
895     AUTHOR="Drucbert, Jean-Pierre F.",
896     ADDRESS="{\texttt{drucbert@onera.fr}}",
897     NOTE="{\url{CTAN:macros/latex/contrib/minitoc/minitoc.pdf}}",
898     YEAR=2006}
899
900 @MISC{fminitoc,
901     TITLE="Le paquetage {\texttt{minitoc}}",
902     MONTH=aug,
903     AUTHOR="Drucbert, Jean-Pierre F.",
904     ADDRESS="{\texttt{drucbert@onera.fr}}",
905     NOTE="{\url{CTAN:macros/latex/contrib/minitoc/fminitoc.pdf}}",
906     YEAR=2006}
907
908 @MISC{shorttoc,
909     TITLE="The {\texttt{shorttoc}} package",
910     MONTH=aug,
911     AUTHOR="Drucbert, Jean-Pierre F.",
912     ADDRESS="{\texttt{drucbert@onera.fr}}",
913     NOTE="{\url{CTAN:macros/latex/contrib/shorttoc/shorttoc.dtx}}",
914     YEAR=2002}
915 \</addbib>

```

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

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

4.4 The addsec.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 [210] 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:

```

916 \documentclass[oneside,12pt,a4paper]{article}
917 \usepackage{url,tocbibind,makeidx}
918 \makeatletter
919 \newif\ifscan@allowed
920 \scan@allowedtrue
921 \makeatother
922 \def\dotfil{\leaders\hbox to.6em{\hss.\hss}\hfil}%
923 \def\pfill{\unskip~\dotfill\penalty500\strut\nobreak
924           \dotfil~\ignorespaces}%
925 \usepackage[tight,hints,insection,listfiles]{minitoc}
926 %%\usepackage{mtcoff}
927 \makeindex

```

The preparation commands:

```

929 \begin{document}
930 \doparttoc \dopartlof \dopartlot
931 \dosecttoc \dosectlof \dosectlot

```

The commands to prepare the table of contents, the list of figures and the list of table. As we use the `tocbibind` package, we must add some `\mtcaddsection` commands:

```

932 \setcounter{tocdepth}{6}
933 \setcounter{parttocdepth}{6}
934 \setcounter{secttocdepth}{6}
935 \tableofcontents
936 \mtcaddsection
937 \listoffigures
938 \mtcaddsection
939 \listoftables
940 \mtcaddsection

```

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.

```

941 \part{Part~1}
942 \parttoc \mtcskip \partlof \mtcskip \partlot
943
944 \section{First section}
945 \index{section!normal}
946 \secttoc \mtcskip \sectlof \mtcskip \sectlot

```

```

947 \subsection{First subsection}
948
949 \begin{figure}[tp]
950 \caption{First figure}
951 \end{figure}
952 \begin{table}[tp]
953 \caption{First table}
954 \end{table}
955
956 \subsection{Second subsection}
957 A small nice citation:\
958 \index{small}\index{citation}\index{nice}\index{A}%
959 \index{a}\index{and}\index{bird}\index{But}%
960 \index{cannot}\index{claim}\index{great}\index{he}%
961 \index{I}\index{imagine}\index{it}\index{know}%
962 \index{land}\index{on}\index{once}\index{that}%
963 \index{to}\index{tree}\index{would}\index{yes}%
964 A bird cannot land once on a great tree and claim to know it.
965 But I imagine that he would, yes.\
966 \hbox{}\hfill
967 Iain M. Banks (1993), \textsl{Against a dark background.}%
968 \index{Iain}\index{Banks}\index{Against}\index{dark}%
969 \index{background}
970 \begin{figure}[tp]
971 \caption{Second figure}
972 \end{figure}
973 \begin{table}
974 \caption{Second table}
975 \end{table}

```

Here, we try a starred section, with its entry in the table of contents. You can try several solutions (good or bad).

```

976 \section*{Second section, starred}
977 \index{section!starred}
978 %% UNCOMMENT ONE AND ONLY ONE OF THE 4 FOLLOWING LINES
979 \mtcaddsection[Second section, starred] % OK
980 %%\mtcaddsection[] % BAD
981 %%\mtcaddsection[~] % produce a (strange) correct result.
982 %%\mtcaddsection % BAD
983 %%%\mtcaddsection[~] % produce a (strange) correct result.
984 \index{tests}
985
986 This is a starred section; you can test here variations on
987 the \verb|\mtcaddsection| command. Just uncomment one (and
988 only one) of the \verb|\mtcaddsection| commands after
989 \verb|\section*| in the source code of \texttt{add.tex}.
990 For each case, look at the \index{Table of Contents}Table of Contents
991 and at this section.
992 \index{a}\index{added}\index{after}\index{also}\index{and}%
993 \index{at}\index{can}\index{case}\index{section}%
994 \index{code}\index{command}\index{commands}%
995 \index{Contents}\index{each}\index{entries}\index{For}%

```

```

996 \index{here}\index{I}\index{in}\index{index}\index{is}%
997 \index{Just}\index{just}\index{look}\index{lot}%
998 \index{of}\index{on}\index{one}\index{only}\index{source}%
999 \index{starred}\index{Table}\index{test}\index{the}%
1000 \index{This}\index{this}\index{to}\index{uncomment}%
1001 \index{variations}\index{you}%
1002 I also added a lot of index entries, just to test.
1003
1004 \section{Third section}
1005 \index{section!normal}
1006 \secttoc \mtcskip \sectlof \mtcskip \sectlot
1007 \subsection{Third subsection}
1008
1009 \begin{figure}
1010 \caption{Third figure}
1011 \end{figure}
1012 \begin{table}
1013 \caption{Third table}
1014 \end{table}
1015
1016 \subsection{Fourth subsection}
1017
1018 \begin{figure}
1019 \caption{Fourth figure}
1020 \end{figure}
1021 \begin{table}
1022 \caption{Fourth table}
1023 \end{table}
1024
1025 \subsubsection{Even a sub-sub-section!}
1026 \subsubsection{And yet another one}
1027
1028 \part{Part~2}
1029 \parttoc \mtcskip \partlof \mtcskip \partlot
1030
1031 \section{Fourth section}
1032 \index{section!normal}
1033 \secttoc \mtcskip \sectlof \mtcskip \sectlot
1034 \subsection{Fifth subsection}
1035
1036 \begin{figure}[tp]
1037 \caption{Fifth figure}
1038 \end{figure}
1039 \begin{table}[tp]
1040 \caption{Fifth table}
1041 \end{table}
1042
1043 \subsection{Sixth subsection}
1044 A small nice citation:\\
1045 \index{small}\index{citation}\index{nice}\index{A}%
1046 \index{a}\index{and}\index{bird}\index{But}%
1047 \index{cannot}\index{claim}\index{great}\index{he}%
1048 \index{I}\index{imagine}\index{it}\index{know}%
1049 \index{land}\index{on}\index{once}\index{that}%

```

```

1050 \index{to}\index{tree}\index{would}\index{yes}%
1051 A bird cannot land once on a great tree and claim to know it.
1052 But I imagine that he would, yes.\\
1053 \hbox{}\hfill
1054 Iain M. Banks (1993), \textsl{Against a dark background.}%
1055 \index{Iain}\index{Banks}\index{Against}%
1056 \index{dark}\index{background}
1057
1058 \begin{figure}[tp]
1059 \caption{Sixth figure}
1060 \end{figure}
1061 \begin{table}
1062 \caption{Sixth table}
1063 \end{table}
1064
1065 \section*{Fifth section, starred}
1066 \index{section!starred}
1067 %% UNCOMMENT ONE AND ONLY ONE OF THE 4 FOLLOWING LINES
1068 \mtcaddsection[Fifth section, starred] % OK
1069 %%\mtcaddsection[] % OK
1070 %%\mtcaddsection[~] % produce a (strange) correct result.
1071 %%\mtcaddsection % OK
1072 %%%%%%%%%%%
1073 \index{tests}
1074
1075 This is a starred section; you can test here variations on
1076 the \verb|\mtcaddsection| command. Just uncomment one (and
1077 only one) of the \verb|\mtcaddsection| commands after
1078 \verb|\section*| in the source code of \texttt{add.tex}.
1079 For each case, look at the \index{Table of Contents}Table of Contents
1080 and at this section.\index{a}%
1081 \index{added}\index{after}\index{also}\index{and}%
1082 \index{at}\index{can}\index{case}\index{section}%
1083 \index{code}\index{command}\index{commands}\index{Contents}%
1084 \index{each}\index{entries}\index{For}\index{here}%
1085 \index{I}\index{in}\index{index}\index{is}%
1086 \index{Just}\index{just}\index{look}\index{lot}%
1087 \index{of}\index{on}\index{one}\index{only}%
1088 \index{source}\index{starred}\index{Table}\index{test}%
1089 \index{the}\index{This}\index{this}\index{to}%
1090 \index{uncomment}\index{variations}\index{you}%
1091 I also added a lot of index entries, just to test.
1092
1093 \section{Sixth section}
1094 \index{section!normal}
1095 \secttoc \mtcskip \sectlof \mtcskip \sectlot
1096 \subsection{Seventh subsection}
1097
1098 \begin{figure}
1099 \caption{Seventh figure}
1100 \end{figure}
1101 \begin{table}
1102 \caption{Seventh table}
1103 \end{table}

```



```

1104
1105 \subsection{Eighth subsection}
1106
1107 \begin{figure}
1108 \caption{Eighth figure}
1109 \end{figure}
1110 \begin{table}
1111 \caption{Eighth table}
1112 \end{table}

```

The bibliography: as we want an entry for it in the table of contents, we use the `tocbibind` package [210] and a correction with `\adjuststc`:

```

1113 \nocite*
1114 \def\noopsort#1{\relax}
1115 \bibliographystyle{plain}
1116 \bibliography{add}
1117 \adjuststc

```

The index: as we want an entry for it in the table of contents, we use the `tocbibind` package [210] and a correction with `\mtcfixindex`:

```

1118 \printindex
1119 \mtcfixindex % use this OR the 2 following lines
1120 %%\addcontentsline{lof}{xsect}{}
1121 %%\addcontentsline{lot}{xsect}{}
1122 %%\mtcaddsection
1123
1124 \appendix
1125 \section{App.~1}
1126 \index{section!appendix}
1127 \secttoc \mtcskip \sectlof \mtcskip \sectlot
1128 \subsection{Ninth subsection}
1129
1130 \begin{figure}
1131 \caption{Ninth figure}
1132 \end{figure}
1133 \begin{table}
1134 \caption{Ninth table}
1135 \end{table}
1136
1137 \subsection{Tenth subsection}
1138
1139 \begin{figure}
1140 \caption{Tenth figure}
1141 \end{figure}
1142 \begin{table}
1143 \caption{Tenth table}
1144 \end{table}
1145
1146 \section{App.~2}
1147 \index{section!appendix}
1148 %% contains no tables but asks for a sectlot! No sectlot printed.

```

```

1149 \secttoc \mtcskip \sectlof \mtcskip \sectlot
1150 \subsection{Eleventh subsection}
1151
1152 \begin{figure}
1153 \caption{Eleventh figure}
1154 \end{figure}
1155 \begin{figure}
1156 \caption{Twelfth figure}
1157 \end{figure}
1158
1159 \subsection{Twelfth subsection}
1160
1161 \begin{figure}
1162 \caption{Thirteenth figure}
1163 \end{figure}
1164 \begin{figure}
1165 \caption{Fourteenth figure}
1166 \end{figure}
1167
1168 \end{document}
1169 \end{addsec}

```

4.5 The second.tex document file

This document tests the french2.mld minitoc language definition file (section 13.59 on page 444) and its supporting code in the minitoc package. First, the preamble of the document uses the french2 minitoc package language option¹:

```

1170 (*second)
1171 %% Test de french2.mld:
1172 %% « seconde » ou « deuxième » partie?
1173 %% compilez 3 fois.
1174 \documentclass{report}
1175 \usepackage[french2,tight,listfiles]{minitoc}
1176 \usepackage[french]{babel}
1177 \usepackage{franc,frnew}
1178 \usepackage[OT1,TS1,T1]{fontenc}
1179 \usepackage[isolatin]{inputenc}
1180 \usepackage{mypatches}
1181 \begin{document}

```

We test if there is only two parts: if yes, we will use “seconde”. Else (zero, one, three or more parts), we will use “deuxième”.

```

1182 \ifmtcsecondpart
1183 Il n'y a que 2 parties: seconde.
1184 \else
1185 Il y a une partie ou plus de deux parties: deuxième.

```

¹ The franc, frnew and mypatches packages are local additions, provided with the minitoc package.

The body of the document, with two parts:

If you want also a third part, comment out this line and recompile 3 times:

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. By example, try to add the following lines after `\faketableofcontents`:

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

4.6 The app-mem.tex document file

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

```
1206 \documentclass[oneside]{memoir}
1207 \documentclass[oneside]{memoir}
1208 \usepackage{lipsum} % filling text
1209 \usepackage%
1210 [tight,listfiles]{minitoc}
1211 \begin{document}
1212 \dominitoc
1213 \tableofcontents
1214 \adjustmtc
1215 \chapter{First chapter}
1216 \minitoc
1217 \lipsum[1]
1218 \section{First section}
1219 \lipsum[2]
```

The `appendices` are set in an `appendices` environment; we can

add an entry in the TOC with `\addappheadtotoc` (a command from the `memoir` class):

```
1220 \begin{appendices}
1221 \addappheadtotoc
1222 \chapter{Afterthought}
1223 \minitoc
1224 \lipsum[3]
1225 Afterthought appendix
1226 \section{Further remarks}
1227 \lipsum[4]
1228 \end{appendices}
1229 \chapter{Conclusion}
1230 \minitoc
1231 \section{Bye}
1232 \lipsum[5]
1233 \end{document}
1234 \end{app-mem}
```

4.7 The mem.tex document file

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

```
1235 \documentclass%
1236 [oneside]{memoir}
1237 \usepackage{lipsum}
1238 \usepackage{hyperref}
1239 \usepackage{memhfixc}
1240 \usepackage%
1241 [tight,listfiles]{minitoc}
1242 \usepackage{mtcoff}
1243 \begin{document}
```

We use the starred form `\tableofcontents*` specific of the `memoir` class. Note that the the

`\chapter` command has *two* optional arguments in the `memoir` class.

```
1245 \dominitoc
1246 \tableofcontents*
1247
1248 \chapter[oneA][oneB]{oneC}
1249 \minitoc
1250 \lipsum[1]
1251 \newpage
1252 \lipsum[2]
1253 \section{S-1-one}
1254 \lipsum[3]
1255 \section{S-1-two}
1256 \lipsum[4]
1257
1258 \chapter[twoA][twoB]{twoC}
1259 \minitoc
1260 \lipsum[4]
1261 \newpage
```

<pre> 1262 \lipsum[5] 1263 \section{S-2-one} 1264 \lipsum[6] 1265 \section{S-2-two} </pre>	<pre> 1266 \lipsum[7] 1267 \end{document} 1268 \mem </pre>
--	--

4.8 The `mem1.tex` document file

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

```

1269 \mem1
1270 \documentclass[oneside]{memoir}
1271 \usepackage{lipsum} % filling text

```

We inhibit some font commands of the `memoir` class:

```

1272 \providecommand{\cftsecfont}{\empty}
1273 \providecommand{\cftsubsecfont}{\empty}

```

Then we redefine them:

```

1274 \renewcommand{\cftsubsecfont}{\normalsize\scshape}
1275 \renewcommand{\cftsubsecfont}{\normalsize\scshape}

```

We load the `minitoc` package and try to use some `minitoc` font commands, without success:

```

1276 %% hyperref before minitoc, optional
1277 %% \usepackage[linktocpage=true]{hyperref} \usepackage{memhfixc}
1278 \usepackage[tight,listfiles]{minitoc}
1279 \mtcsetfont{minitoc}{section}{\normalsize\scshape} % <- no scshape
1280 \mtcsetfont{minitoc}{subsection}{\normalsize\scshape}% <- no scshape

```

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

```

1281 \providecommand{\cftsecfont}{\empty}
1282 \providecommand{\cftsubsecfont}{\empty}
1283 \renewcommand{\cftsubsecfont}{\normalsize\rmfamily\scshape}
1284 \renewcommand{\cftsubsecfont}{\normalsize\rmfamily\scshape}

```

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

```

1285 \mtcsettitlefont{minitoc}{\Large\scshape}
1286 %% this is working beautifully ->
1287 \mtcsettitlefont{minitoc}{Chapter Contents}

```

The document body:

```

1288 \begin{document}
1289 \dominitoc
1290 \tableofcontents*
1291
1292 \chapter[OneA][OneB]{OneC}
1293 \minitoc
1294 \section{This section}
1295 \lipsum[1]
1296 \section{Second section}
1297 \lipsum[2]
1298 \section{Third section}
1299 \lipsum[3]
1300 \end{document}
1301 </mem1>

```

4.9 The 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.

```

1302 <{*fo1>
1303 \documentclass{report}
1304 \usepackage{lipsum}
1305 \usepackage{txfonts}
1306 \usepackage[tight,listfiles]{minitoc}
1307 \begin{document}
1308 \doparttoc
1309 \dominitoc
1310 \tableofcontents
1311 \part{Introduction}
1312 \clearpage

```

A first copy, with default fonts:

```

1313 \parttoc

```

A second copy, roman bold font for chapter entries:

```

1314 \clearpage
1315 \mtcsetfont{parttoc}{chapter}{\normalsize\rmfamily\upshape\bfseries}
1316 \parttoc

```

A third copy, typewriter bold font for chapter entries:

```

1317 \clearpage
1318 \mtcsetfont{parttoc}{chapter}{\normalsize\ttfamily\upshape\bfseries}

```

```
1319 \parttoc
```

A fourth copy, not bold typewriter font for chapter entries:

```
1320 \clearpage
1321 \mtcsetfont{parttoc}{*}{\normalsize\ttfamily\upshape\mdseries}
1322 \parttoc
1323 \chapter{A very short chapter}
1324 \minitoc
1325 \lipsum[1]
1326 \section{First section}
1327 \lipsum[2]
1328 \subsection{Alpha}
1329 \lipsum[3]
1330 \subsection{Beta}
1331 \lipsum[4]
1332 \section{Second section}
1333 \lipsum[5]
1334 \subsection{Gamma}
1335 \lipsum[6]
1336 \subsection{Delta}
1337 \lipsum[7]
1338 \end{document}
1339 </fo1>
```

4.10 The 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 fo1.tex (see section 4.9 on the preceding page).

```
1340 <fo2>
1341 \documentclass{report}
1342 \usepackage{lipsum}
1343 \usepackage{txfonts}
1344 \usepackage[tight,listfiles]{minitoc}
```

We declare the fonts with the old method:

```
1345 \def\ptcSSfont{\ptcfont} % (subsections)
1346 \def\ptcSSSfont{\ptcfont} % (subsubsections)
1347 \def\ptcPfont{\ptcfont} % (paragraphs)
1348 \def\ptcSPfont{\ptcfont} % (subparagraphs)
1349 \def\plffont{\ptcfont} % (figures)
1350 \def\plfSfont{\ptcfont} % (subfigures)
1351 \def\pltfont{\ptcfont} % (tables)
1352 \def\pltSfont{\ptcfont} % (subtables)
1353 \begin{document}
1354 \doparttoc
```

```

1355 \dominitoc
1356 \tableofcontents
1357 \part{Introduction}
1358 \clearpage

```

A first version of the parttoc, with the fonts defined above:

```

1359 \parttoc
1360 \clearpage

```

A second version of the parttoc, with chapter entries in a roman bold font:

```

1361 \mtcsetfont{parttoc}{chapter}{\normalsize\rmfamily\upshape\bfseries}
1362 \parttoc
1363 \clearpage

```

A third version of the parttoc, with chapter entries in a typewriter bold font:

```

1364 \mtcsetfont{parttoc}{chapter}{\normalsize\ttfamily\upshape\bfseries}
1365 \parttoc
1366 \clearpage

```

A fourth version of the parttoc, with chapter entries in a non bold typewriter font:

```

1367 \mtcsetfont{parttoc}{*}{\normalsize\ttfamily\upshape\mdseries}
1368 \parttoc
1369 \chapter{A very short chapter}
1370 \minitoc
1371 \lipsum[1]
1372 \section{First section}
1373 \lipsum[2]
1374 \subsection{Alpha}
1375 \lipsum[3]
1376 \subsection{Beta}
1377 \lipsum[4]
1378 \section{Second section}
1379 \lipsum[5]
1380 \subsection{Gamma}
1381 \lipsum[6]
1382 \subsection{Delta}
1383 \lipsum[7]
1384 \end{document}
1385 </fo2>

```


4.11 The `scr.tex` document file

This document shows the use of the `minitoc` package with a KOMA-Script document class, `scrreprt`. Some precautions are needed, because these classes have specific interface with the TOC (package options and commands).

```

1386 \*Scr
1387 \documentclass[12pt,halfparskip,liststotoc,bibtotoc]{scrreprt}
1388 \setcounter{secnumdepth}{4}
1389 \setcounter{tocdepth}{4}
1390 \usepackage[latin1]{inputenc}
1391 \usepackage{longtable}

```

The `hyperref` package, if used, must be loaded *before* `minitoc`:

```

1392 \usepackage{hyperref}

```

With a KOMA-Script class, use the `k-tight` package option in place of `tight`; as it is a document in german, use also a language package option:

```

1393 \usepackage[k-tight,germanb,listfiles]{minitoc}
1394 \usepackage[germanb]{babel}
1395 \begin{document}
1396 \pagenumbering{Roman} % page number in Roman, reset to 1 (I)
1397 \dominitoc
1398 \tableofcontents

```

In this document class, with the `liststotoc` class option, `\listoftables` adds a chapter entry in the TOC, so we must add `\mtcaddchapter` after `\listoftables`:

```

1399 \listoftables
1400 \mtcaddchapter

```

With the `bibtotoc` class option, you should add a `\adjustmtc` command after `\bibliography`. This problem is similar to the one of compatibility with the `tocbibind` package.

```

1401 \clearpage
1402 \pagenumbering{arabic} % page number in arabic digits
1403 \setcounter{page}{1} % forced to 1
1404
1405 \chapter[Test 1]\label{cha:test-1}
1406
1407 Text.
1408
1409 \section{Tabelle}\label{sec:tabelle}
1410
1411 \begin{table}
1412   \centering

```

```

1413 \begin{longtable}[l]{ll}
1414 \underline{Ausbildungsbetrieb} \hspace{10mm} & Kommanditgesellschaft
1415 \\
1416 \underline{Ausbildender} & Hammer \\
1417 Ausbildungsstätte & XXXXX Ort
1418 \end{longtable}
1419 \caption{Ausbildungsbetrieb}
1420 \end{table}
1421
1422 Text.
1423
1424 \appendix
1425 \chapter{Dokumente}
1426 \minitoc % Aufruf Minitoc
1427
1428 \section{Anhang}\label{sec:anhang}
1429 Text.\\
1430 Hier sollte nun der Anhang sein, davor das Verzeichnis dafür.
1431 \end{document}
1432 </scr>

```

4.12 The subf.tex document file

This document shows the use of the minitoc package with a document containing subfigures (here with the subfigure package [84]). 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:

```

1433 <{*subf}
1434 \documentclass[12pt]{report}
1435 \usepackage{subfigure}
1436 \makeatletter
1437 \renewcommand{\l@subfigure}{%
1438 \@dottedxxxline{\ext@subfigure}{2}{3.9em}{3.3em}}
1439 \makeatother

```

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

```

1440 \usepackage{varioref}
1441 \usepackage%
1442 [tight,listfiles]{minitoc}

```

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

```

1443 \newcommand{\goodap}{%

```

```

1444 \hspace{\subfigtopskip}%
1445 \hspace{\subfigbottomskip}}
1446 \setcounter{lofdepth}{2}
1447 \begin{document}

```

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

```

1448 \mtcsetdepth{minilof}{2}
1449 \mtcsetfont{minitoc}{section}{\small\rmfamily\upshape\bfseries}
1450 \mtcsetfont{partlof}{subfigure}{\small\rmfamily\slshape\bfseries}
1451 \mtcsetfont{partlof}{figure}{\small\rmfamily\upshape\bfseries}
1452 \mtcsetfont{minilof}{subfigure}{\small\rmfamily\slshape\bfseries}
1453 \mtcsetfont{minilof}{figure}{\small\rmfamily\upshape\bfseries}
1454 %% no tables in this document
1455 %% \mtcsetfont{partlot}{subtable}{\small\rmfamily\slshape\bfseries}
1456 %% \mtcsetfont{partlot}{table}{\small\rmfamily\upshape\bfseries}
1457 %% \mtcsetfont{minilot}{subtable}{\small\rmfamily\slshape\bfseries}
1458 %% \mtcsetfont{minilot}{table}{\small\rmfamily\upshape\bfseries}

```

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

```

1459 \dominilof
1460 \tableofcontents
1461 \listoffigures

```

A chapter, with is minilof, twice but with different depths:

```

1462 \chapter{First Chapter}
1463 \minilof
1464 \mtcskip
1465 \mtcsetdepth{minilof}{1}
1466 \minilof

```

A figure containing three subfigures and their captions:

```

1467 \begin{figure}
1468 \centering
1469 \fbox{%
1470 \begin{minipage}{3.5in}%
1471 \raggedright
1472 \begin{center}
1473 \subfigure[First]{%
1474 \fbox{\hbox to 20mm{\vbox to 15mm{\vfil\null}\hfil}}}%
1475 \hspace{\subfigtopskip}\hspace{\subfigbottomskip}%
1476 \subfigure[Second Figure]{%
1477 \fbox{\hbox to 20mm{\vbox to 10mm{\vfil\null}\hfil}}}%
1478 \subfigure[Third]{\label{3figs-c}%
1479 \fbox{\hbox to 20mm{\vbox to 10mm{\vfil\null}\hfil}}}%
1480 \caption{Three subfigures.}%
1481 \label{3figs}%
1482 \end{center}

```

```

1483      \vspace{4pt}%
1484      This figure contains two top ‘subfigures’ and
1485      Figure~\ref{3figs-c}.
1486      \end{minipage}}
1487 \end{figure}
1488 Figure~\vref{3figs} contains two top ‘‘subfigures’’ and
1489 Figure~\vref{3figs-c}.
1490 \end{document}
1491 </subf>

```

4.13 The tsfc.tex document file

This document² shows the use of the minitoc package with a document containing subfigures (here with the subfig package [86]). We show how to use minilofs and to adjust their depth.

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

```

1492 < *tsfc>
1493 \documentclass{report}
1494 \usepackage{fullpage}
1495 \usepackage[config=altsf]{subfig}
1496 \usepackage[tight,listfiles]{minitoc}

```

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

```

1497 \newdimen\testtemp
1498 \newcommand{\ru}[1]{%
1499   \testtemp #1%
1500   \advance\testtemp .5pt%
1501   \divide\testtemp 2%
1502   \hbox to \testtemp{\leaders\hbox to 1mm{%
1503     \vrule height1mm depth0pt width.25pt\hfil}\hfil}%
1504   \hbox to 0pt{\hss\vrule height3mm depth0pt width.25pt\hss}%
1505   \hbox to \testtemp{\leaders\hbox to 1mm{%
1506     \hfil\vrule height1mm depth0pt width.25pt\hfil}}
1507 %%
1508 \fboxsep=-\fboxrule
1509 \newcommand{\figbox}[1]{%
1510   \fbox{%
1511     \vbox to 1in{%
1512       \vfil
1513       \hbox to 2in{%
1514         \parbox{2in}{%
1515           \centering
1516           #1}}}%

```

² It is derived from one of the examples distributed with the subfig package [86].

```

1517     \vfil
1518     \vbox to 0pt{%
1519         \vss
1520         \hbox to 2in{%
1521             \hfil
1522             \ru{1.1in}%
1523             \hfil}}}}

```

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

```

1524 \begin{document}
1525 \setcounter{lofdepth}{2}
1526 \dominilof
1527 \listoffigures
1528 \newpage

```

A chapter containing a figure with subfigures. We print a minilof twice, with different depths:

```

1529 \chapter{Reference Test}
1530 \minilof
1531 \mtcskip
1532 \mtcsetdepth{minilof}{1}
1533 \minilof
1534
1535 \begin{figure}[ht]%
1536     \centering
1537     \subfigure{%
1538         \label{fig+A}%
1539         \figbox{SUBFIGURE ONE:\\
1540             (no opt)}}
1541     \quad
1542     \subfigure[]{%
1543         \label{fig+B}%
1544         \figbox{SUBFIGURE TWO:\\
1545             (empty opt)}}\\
1546
1547     \subfigure[Subfigure Three.]{%
1548         \label{fig+C}%
1549         \figbox{SUBFIGURE THREE:\\
1550             (opt)}}
1551     \quad
1552     \subfigure[][Subfigure Four.]{%
1553         \label{fig+D}%
1554         \figbox{SUBFIGURE FOUR:\\
1555             (empty opt and opt)}}
1556     \quad
1557     \subfigure[][]{%
1558         \label{fig+E}%
1559         \figbox{SUBFIGURE FIVE:\\
1560             (both empty opt)}}\\
1561

```

```

1562 \subfigure[The Sixth Subfigure.][Subfigure Six.]{%
1563   \label{fig+F}%
1564   \figbox{SUBFIGURE SIX:\\
1565     (both opt)}}
1566 \quad
1567 \subfigure[The Seventh Subfigure][]{%
1568   \label{fig+G}%
1569   \figbox{SUBFIGURE SEVEN:\\
1570     (opt and empty opt)}}
1571
1572 \caption{Optional argument test.}%
1573 \label{fig+main}%
1574 \end{figure}
1575
1576 The figure~\ref{fig+main} on page~\pageref{fig+main} is composed
1577 of the seven subfigures~\subref{fig+A} (aka: \ref{fig+A}),
1578 \subref{fig+B} (aka: \ref{fig+B}), \subref{fig+C} (aka: \ref{fig+C}),
1579 \subref{fig+D} (aka: \ref{fig+D}), \subref{fig+E} (aka: \ref{fig+E}),
1580 \subref{fig+F} (aka: \ref{fig+F}), and \subref{fig+G} (aka: \ref{fig+G}).
1581
1582 \section{Centering Test}
1583 Note that figures~\ref{fig+B}, \ref{fig+E} and \ref{fig+G} are centered.
1584 This means that \verb|\subfigcapskip| has been set to zero and is not
1585 offsetting the simple label to the left. Also the remaining captioned
1586 subfigures (figures~\ref{fig+C}, \ref{fig+D}, and \ref{fig+F}) should
1587 have centered labels.
1588 \end{document}
1589 </tsfc>

```

4.14 The tbi.tex document file

This document shows the use of the minitoc package with a document using the tocbibind package [210].

```

1590 <{*tbi}
1591 \documentclass[a4paper]{report}

```

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

```

1592 \usepackage[nottoc]{tocbibind}
1593 \usepackage%
1594   [tight,listfiles]{minitoc}
1595 \begin{document}
1596 \dominitoc
1597 \tableofcontents

```

As there is no entry for the TOC in the TOC, no correction is necessary; we comment out the usual correction³:

```

1598 %% tocbibind compatibility
1599 %% not used if nottoc option :
1600 %% \mtcaddchapter[]

```

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

```

1601 \chapter*{Introduction}
1602 \mtcaddchapter[Introduction]

```

³ It is recommended to keep this comment; you could change your mind.

Some normal chapters:

```
1603 \chapter{Title of chapter~1}
1604 \minitoc
1605 \section{as1}
1606 \section{as2}
1607 \chapter{Title of chapter~2}
1608 \minitoc
1609 \section{bs1}
1610 \section{bs2}
1611 \chapter{Title of chapter~3}
1612 \minitoc
1613 \section{cs1}
1614 \section{cs2}
1615 \chapter{Title of chapter~4}
1616 \minitoc
1617 \section{ds1}
1618 \section{ds2}
```

Another starred chapter, with an entry in the TOC:

```
1619 \chapter*{Conclusion}
1620 \mtcaddchapter[Conclusion]
```

Yet another starred chapter, with an entry in the TOC, but with starred sections, also listed in the TOC:

```
1621 \chapter*{Appendices}
1622 \mtcaddchapter[Appendices]
1623 \minitoc
1624 \section*{first appendix}
1625 \addcontentsline{toc}{section}%
1626   {first appendix}
1627 \section*{second appendix}
1628 \addcontentsline{toc}{section}%
1629   {second appendix}
```

The list of figures has an entry in the TOC (via the `tocbibind` package), so a correction must be applied:

```
1630 \listoffigures
1631 %% tocbibind compatibility
1632 \mtcaddchapter
```

The `tocbibind` package adds an entry in the TOC for the bibliography, so we must add the recommended correction:

```
1633 \begin{thebibliography}{3}
1634   \bibitem {s1}{title ...}
1635 \end{thebibliography}
1636 %% tocbibind compatibility
1637 \adjustmtc
1638 \end{document}
1639 </tbi>
```

4.15 The `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 [156] and its `multicols` environment. We set `\mtcindent` to zero. We test several combinations. If a `minitoc` is long enough to be splitted on both columns, the result may be funny.

```
1640 < *2c>
1641 \documentclass[12pt,a4paper]%
1642   {report}
1643 \usepackage{multicol}
1644 \usepackage{lipsum}
```

```
1645 \usepackage%
1646   [tight,latin,listfiles]{minitoc}
1647 \usepackage[latin]{babel}
1648 \setlength{\mtcindent}{0pt}
1649 \begin{document}
1650 \dominitoc
1651 \tableofcontents
```

Two first chapters with a standard chapter head, a `minitoc` on one full width column, then the body of the chapter on two columns:

```
1652 \chapter{Primum capitulum}
1653 \begin{multicols}{2}[\minitoc]
```

```

1654 \section{Prima sectio}
1655 \lipsum[1-2]
1656 \section{Secunda sectio}
1657 \lipsum[3-4]
1658 \end{multicols}
1659 \chapter{Secundum capitulum}
1660 \begin{multicols}{2}[\minitoc]
1661 \section{Prima sectio}
1662 \lipsum[5-6]
1663 \section{Secunda sectio}
1664 \lipsum[7-8]
1665 \section{Tertia sectio}
1666 \lipsum[9-10]
1667 \end{multicols}

```

A third chapter entirely on two columns, so the chapter head and the minitoc are in the first column:

```

1668 \begin{multicols}{2}
1669 \chapter{Tertium capitulum}
1670 \minitoc
1671 \section{Prima sectio}
1672 \lipsum[11-12]

```

```

1673 \section{Secunda sectio}
1674 \lipsum[13-14]
1675 \section{Tertia sectio}
1676 \lipsum[15-16]
1677 \end{multicols}

```

A fourth chapter, with the chapter head on one column, and the minitoc and the chapter body on two columns (the minitoc is in the first column):

```

1678 \chapter{Quadrum capitulum}
1679 \begin{multicols}{2}
1680 \minitoc
1681 \section{Prima sectio}
1682 \lipsum[16-17]
1683 \section{Secunda sectio}
1684 \lipsum[18-19]
1685 \section{Tertia sectio}
1686 \lipsum[20-21]
1687 \end{multicols}
1688 \end{document}
1689 </2c>

```

4.16 The `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 [207]. 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 some useful packages for the french language, the `geometry` package [196], which defines the global page layout, the `multicol` package [156], the `color` package [81], because we want a colored background for the minitoc, the `tocloft` package [207], to change some parameters of the minitoc, and, at least, the `minitoc` package.

```

1690 < *bo>
1691 \documentclass[10pt]{book}
1692 \usepackage[paperwidth=8.5in, paperheight=11in,%
1693     lmargin=1.25in, rmargin=1.25in, tmargin=1in, bmargin=1in]{geometry}
1694 \usepackage[francais]{babel}
1695 \usepackage{franc, frnew}
1696 \usepackage[T1]{fontenc}
1697 \usepackage[isolatin]{inputenc}
1698 \usepackage{mypatches}
1699 \usepackage{multicol}
1700 \usepackage{color}

```


We use the `tocloft` package and its commands to set the indentations in the TOC and the minitoc:

```
1701 \usepackage{tocloft}
1702 \setlength{\cftsecindent}{0cm}
1703 \setlength{\cftsecnumwidth}{15 pt}
1704 \setlength{\cftsubsecindent}{\cftsecindent}
1705 \addtolength{\cftsubsecindent}{\cftsecnumwidth}
1706 \setlength{\cftsubsecnumwidth}{20 pt}
1707 \setlength{\cftsubsubsecindent}{\cftsubsecindent}
1708 \addtolength{\cftsubsubsecindent}{\cftsubsecnumwidth}
```

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

```
1709 \cftpagenumbersoff{sec}
1710 \cftpagenumbersoff{subsec}
```

We load the `minitoc` package and change the indentation, suppress the rules and change the minitoc title. The `hyperref` package [180] can also be loaded (*after* `minitoc`).

```
1711 \usepackage[francais,tight,listfiles]{minitoc}
1712 \usepackage{hyperref}
1713 \setlength{\mtcindent}{0pt}
1714 \nomtcrule           % pas de filets en haut et en bas de la mini-tdm
1715 \nomtcpagenumbers    % pas de numéro de pages
1716 %                   (non fonctionnel avec tocloft)
1717 \renewcommand{\mtctitle}{Contenu de la rencontre}
```

This code redefines the format of the chapter head:

```
1718 %%%%%%%%%%%
1719 %  MACRO POUR AVOIR LE MOT RENCONTRE AU LIEU DE CHAPITRE
1720 %  Sans saut de ligne
1721 %  (modification du code qui se trouve dans la FAQ)
1722 %%%%%%%%%%%
1723 \makeatletter
1724 \def\@makechapterhead#1{%
1725   \vspace*{10\p@}%
1726   {\parindent \z@ \raggedleft \normalfont
1727    \interlinepenalty\@M
1728    \ifnum \c@secnumdepth >\m@ne
1729     \Huge\bfseries\sffamily Rencontre \thechapter\%\quad
1730     \fi
1731     \Huge\bfseries\sffamily #1\par\nobreak
1732     \vskip 10\p@
1733   }}
1734 \def\@makeschapterhead#1{%
1735   \vspace*{10\p@}%
1736   {\parindent \z@ \raggedright \normalfont
1737    \interlinepenalty\@M
```

```

1738 \Huge \sffamily #1\par\nobreak
1739 \vskip 10\p@
1740 }} \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.

```

1741 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1742 %: ENVIRONNEMENT POUR LA PAGE 1 DES RENCONTRES
1743 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1744 % param1: date de la rencontre
1745 % param2: nombre de périodes
1746 % param3: liste des documents distribués
1747 % param4: messages
1748 % param5: lecture
1749 % param6: exercices
1750 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1751 \newenvironment{pageUn}[6]{%
1752 \parindent = 0.0in
1753 \rule{\linewidth}{1pt}
1754 \begin{multicols}{2}
1755     {\large \bfseries Math. pour médecine nucléaire\}
1756     \textit{(#2)}}
1757     \vfill\columnbreak
1758     \raggedleft\bfseries Automne 2003\}
1759     #1
1760 \end{multicols}
1761 \vspace{-18pt}
1762 \rule{\linewidth}{1pt}
1763
1764 \setlength{\columnseprule}{.3pt}
1765 \setlength{\columnsep}{1cm}
1766 \begin{multicols}{2}%
1767
1768 %: TABLE DES MATIÈRES (col. gauche)
1769 \colorbox{cmyk}{.1,0,0,0}{%
1770     \parbox{\linewidth}{%
1771         \setcounter{minitocdepth}{3}%
1772         \minitoc%
1773     }}
1774 %\vfill \columnbreak ~ \vfill
1775
1776 \mtcskip
1777
1778 %: DOCUMENTS DISTRIBUÉS (début col. droite)
1779
1780 {\large \bfseries Documents distribués}
1781 \begin{itemize} \renewcommand{\labelitemi}{$\star$} #3 \end{itemize}
1782
1783 %: MESSAGES AUX ÉTUDIANTS
1784

```

```

1785 \vspace{12pt}{\large \bfseries Messages}
1786 \begin{itemize} \renewcommand{\labelitemi}{$\star$}\#4\end{itemize}
1787
1788 %: LECTURE
1789 \vspace{12pt}{\large \bfseries Lecture}\vspace{-6pt} \par#5 \par
1790
1791 % EXERCICES
1792 \vspace{12pt}{\large \bfseries Exercices}\vspace{-6pt}\par#6\par
1793
1794 %\newpage
1795 \end{multicols}%
1796 }
1797 %{\newpage}
1798 {\hrule}
1799
1800 %\renewcommand{\baselinestretch}{1.2} %interligne
1801
1802 %\pagestyle{empty} %pas de # de page
1803 %\parindent = 0.0in
1804 %\parskip = 0.1in
1805
1806 %%%%%%%%%%%
1807 %: REDÉFINIR LES SECTIONS
1808 %%%%%%%%%%%
1809 \renewcommand{\thesection}{\Alph{section}}
1810 %\renewcommand{\thesection}{\thechapter-\Alph{section}}
1811

```

The body of the document:

```

1812 %%%%%%%%%%%
1813 %: BEGIN
1814 %%%%%%%%%%%
1815 \begin{document}
1816 \dominitoc
1817 \faketableofcontents
1818 \chapter{Première étape (début)}
1819 %\minitoc
1820 \begin{pageUn}
1821     {%1      Date du cours
1822             lundi 25 août
1823             }
1824     {%2      Durée du cours
1825             2 périodes
1826             }
1827     {%3      Liste des documents à distribuer
1828             \item Plan de cours
1829             \item Fiche d'identification
1830             \item Grille horaire
1831             \item Feuilles d'exercices supplémentaires
1832             }
1833     {%4      Ne pas oublier
1834             \item Acheter le livre de référence

```

```

1835          \item Apporter une disquette
1836          }
1837      {%5    Lecture
1838          Lire les pages ppp à ppp et ppp à ppp
1839          }
1840      {%6    Exercice
1841          Faire les exercices nnn de la page ppp
1842          }
1843 \end{pageUn}
1844
1845 %%%%%%%%%%%
1846 \section{Titre de la section}
1847 %%%%%%%%%%%
1848
1849 Texte dans la section
1850
1851 %%
1852 \subsection{Une sous-section}
1853 %%
1854 Bla bla bla
1855
1856 %%
1857 \subsection{Une autre sous-section}
1858 %%
1859 Bla bla bla
1860
1861
1862 %%%%%%%%%%%
1863 \section{Titre d'une autre section}
1864 %%%%%%%%%%%
1865
1866 Texte dans la section
1867
1868 %%
1869 \subsection{Une sous-section}
1870 %%
1871 Bla bla bla
1872
1873 %%
1874 \subsection{Une autre sous-section}
1875 %%
1876 Bla bla bla
1877
1878
1879 %%%%%%%%%%%
1880 \section{Encore une autre section}
1881 %%%%%%%%%%%
1882
1883 Texte dans la section
1884
1885 \end{document}
1886 </bo>

```

4.17 The `hop.tex` document file

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

```
1887 \<hop>
1888 \documentclass[oneside,12pt]{scrbook}
1889 \usepackage{lipsum}
1890 \usepackage[hints,listfiles]{minitoc}
1891 \begin{document}
```

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

```
1892 \dominitoc
1893 \dominilof
1894 \tableofcontents
1895 \fakelistoffigures
```

A starred part with its entry in the TOC:

```
1896 \part*{Part 1: Strategic Marketing}
1897 \mtcaddpart[Part 1: Strategic Marketing]
```

Then two chapters with their minitocs and minitocs:

```
1898 \chapter{Chapter 1}
1899 \minitoc
1900 \minilof
1901 \section{Section one of first chapter}
1902 \lipsum[1]
1903 \begin{figure}
1904     \centering
1905     Test
1906     \caption{Picture one of first chapter}
1907 \end{figure}
1908
1909 \section{Section two of first chapter}
1910 \lipsum[2]
1911 \begin{figure}
1912     \centering
1913     Test
1914     \caption{Picture two of first chapter}
1915 \end{figure}
1916
1917 \chapter{Chapter 2}
1918 \minitoc \minilof
1919
1920 \section{Section one of second chapter}
1921 \lipsum[3]
1922 \begin{figure}
```

```

1923 \centering
1924 Test
1925 \caption{Picture one of second chapter}
1926 \end{figure}
1927 \cleardoublepage
1928 \section{Section two of second chapter}
1929 \lipsum[4]
1930 \begin{figure}
1931 \centering
1932 Test
1933 \caption{Picture two of second chapter}
1934 \end{figure}
1935 \end{document}
1936 </hop>

```

4.18 The 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: follow the insertion of the mini-tables in the `cri.log` file.

<pre> 1937 <{*cri} 1938 \documentclass[12pt,a4paper]% 1939 {report} 1940 \usepackage[francais]{babel} 1941 \usepackage[T1]{fontenc} 1942 \usepackage[latin1]{inputenc} 1943 \usepackage% 1944 [french2,tight,listfiles]% 1945 {minitoc} 1946 1947 \setcounter{parttocdepth}{2} 1948 1949 \begin{document} 1950 \doparttoc 1951 \dominitoc 1952 \tableofcontents 1953 \part*{Présentation générale} 1954 \mtcaddpart[Présentation générale] 1955 \adjustptc[-2] 1956 \parttoc 1957 1958 Texte de la 1959 présentation générale\ldots 1960 </pre>	<pre> 1961 \chapter*{Les auteurs} 1962 \mtcaddchapter[Les auteurs] 1963 Présentation des auteurs\ldots 1964 1965 \chapter*{Les lecteurs} 1966 \mtcaddchapter[Les lecteurs] 1967 Présentation des lecteurs\ldots 1968 1969 \part{Première partie} 1970 \parttoc 1971 \chapter*{Introduction} 1972 \mtcaddchapter[Introduction] 1973 \chapter{Premier chapitre} 1974 \minitoc 1975 \section{Première section~A} 1976 \section{Deuxième section~A} 1977 \chapter{Deuxième chapitre} 1978 \minitoc 1979 \section{Première section~B} 1980 \section{Deuxième section~B} 1981 1982 \part{Deuxième partie} 1983 \parttoc 1984 \chapter{Premier chapitre} 1985 \minitoc 1986 \section{Première section~C} 1987 \section{Deuxième section~C} 1988 \chapter{Deuxième chapitre} 1989 \minitoc 1990 \section{Première section~D} 1991 \section{Deuxième section~D} 1992 \end{document} 1993 </cri> </pre>
--	---

4.19 The `livre.tex` document file

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

```
1994 <*livre>
1995 \documentclass[10pt,twoside,openright]{book}
```

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

```
1996 \makeatletter
1997 \def\ps@chapterverso{\ps@empty}%
1998 \def\cleardoublepage{\clearpage
1999   \if@twoside
2000     \ifodd\c@page\else
2001       \null\thispagestyle{chapterverso}\newpage
2002     \if@twocolumn\null\newpage\fi
2003   \fi
2004   \fi
2005 }%
2006 \def\ps@chapterverso{\ps@empty}%
2007 \makeatother
```

We define the encodings, for input and output, because the document is in french and uses accented letters:

```
2008 \usepackage[latin1]{inputenc}
2009 \usepackage[TS1,T1]{fontenc}
```

We load two packages, `tocloft` [207], to customize the TOC and the minitocs, and `sectsty` [153], to customize the sectionning commands:

```
2010 \usepackage{tocloft}
2011 \usepackage{sectsty}
```

We load the `minitoc` package then some complementary local packages for the french language:

```
2012 \usepackage[french,undotted,tight,listfiles]{minitoc}
2013 \usepackage[english,français]{babel}
2014 \usepackage{franc,frnew}
2015 \usepackage{mypatches}
2016 \providecommand{\fup}{\textsuperscript}
```

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:

```

2017 \addtolength{\cftsubsecindent}{1em} % for tocloft
2018 \setcounter{tocdepth}{3}
2019 \setcounter{secnumdepth}{1}
2020 \setcounter{minitocdepth}{4}
2021 \chapterfont{\huge\bfseries\sffamily} % for sectsty
2022 \renewcommand{\thesection}{\arabic{section}}
2023 \sectionfont{\Large} % for sectsty

```

Some informations for the title page:

```

2024 \title{Systèmes d'occultation}
2025 \author{Laurent~\textsc{Bloch}}

```

And the document body⁴:

```

2026 \begin{document}
2027
2028 \maketitle
2029
2030 \dominitoc
2031 \tableofcontents
2032
2033 \chapter{Définition et contrôle du travail à~faire}
2034 \label{chap+controle}
2035 \minitoc
2036
2037 \section{Le modèle de la grande industrie et le taylorisme}%
2038 \index{taylorisme}
2039 C'est au \textsc{xviii}\fup{e}~siècle que la vision du travail comme
2040 marchandise est vraiment devenue dominante, pour s'imposer au
2041 \textsc{xix}\fup{e}~siècle dans l'organisation type de la grande usine
2042 industrielle.
2043
2044 \subsection*{Après l'usine, le centre d'appel}\index{centre d'appel}
2045 \addcontentsline{toc}{subsection}{Après l'usine, le centre d'appel}
2046 Aujourd'hui le taylorisme\index{taylorisme} au sens
2047 strict est en déclin parce qu'il n'est plus guère adapté aux
2048 besoins de la production industrielle contemporaine non plus qu'aux
2049 nouvelles normes de comportement individuel et collectif.
2050
2051 \section{Tout travail émet de la pensée}
2052 Le travail a vocation à~produire du sens, pour son auteur comme pour
2053 son destinataire.
2054
2055 \section{Théorie et pratique de la commande publique}
2056 En France, les prestations de service commandées par les
2057 services publics à~des entreprises font l'objet de contrôles
2058 de leur bonne réalisation selon des procédures et des règles
2059 qui sont des cas particuliers d'un ensemble plus vaste, la
2060 réglementation des marchés publics de l'État, dont nous
2061 allons donner ci-dessous une brève description.

```

⁴ The text has been shortened, so there is an undefined reference.

2062
2063 \subsection{Réglementation des marchés publics}
2064 Le dispositif juridique, réglementaire et comptable qui encadre les
2065 actes contractuels de la puissance publique en France est très~[...]
2066
2067 \subsubsection{Premier principe: séparation de l'ordonnateur et du comptable}
2068 Le premier élément du dispositif est le principe de
2069 séparation de l'ordonnateur et du comptable. Il a été
2070 instauré en 1319 par l'ordonnance portant création de la~[...]
2071
2072 \subsubsection{Second principe: contrôle \emph{a~priori}}
2073 Le second élément du dispositif est le principe du contrôle
2074 \emph{a~priori}. Lorsque le directeur de l'organisme public
2075 de recherche pris ici comme exemple (l'ordonnateur) décide~[...]
2076
2077 \subsubsection{Le Code des Marchés Publics}
2078 Le troisième pilier de la commande publique est le Code des Marchés
2079 Publics (CMP), qui régit tous les contrats, conclus par des organismes
2080 publics ou des collectivités territoriales, dont le montant excède un~[...]
2081
2082 \subsection{La pratique des marchés publics}
2083 Lorsque l'administration française fait réaliser un système
2084 informatique par un prestataire, elle est en position de maître
2085 d'ouvrage\index{maîtrise d'ouvrage}. Elle rédige (ou fait rédiger) un
2086 cahier des charges\index{cahier des charges} qui décrit les
2087 spécifications du système à~réaliser. Ce cahier des charges constitue~[...]
2088
2089 \subsection{Quels sont les services publics «~rentables~»?}
2090 Pour parler comme les informaticiens, nous pouvons identifier un
2091 « effet de bord », c'est-à-dire une conséquence non intentionnelle de
2092 la réglementation des marchés publics: les administrations ne disposent
2093 d'aucun moyen pour envisager la notion d'investissement. Le~[...]
2094
2095 \section{Projet et cahier des charges}\index{cahier des charges}
2096 Jean-Pierre~\textsc{Boutinet} nous guidera ici pour ce qui concerne
2097 l'histoire de la notion de~[...]
2098
2099 \subsection{La frontière entre conception et fabrication}
2100 La vision classique de la conduite d'un projet informatique de gestion
2101 est la suivante: le maître d'ouvrage\index{maîtrise d'ouvrage}~[...]
2102
2103 \subsection{Bâtiment, mécanique, programmation}
2104
2105 Nous y reviendrons au chapitre~\ref{chap+travail}, mais nous savons
2106 déjà que la mise en {\oe}uvre de l'informatique s'est beaucoup
2107 inspirée des procédures de travail les plus élaborées du
2108 \textsc{xx}\fup{e}~siècle~[...]
2109
2110 \chapter{Le travail informatique}
2111 \minitoc
2112
2113 \section{De la nature de l'informatique}
2114
2115 \subsection{Premières croyances}\label{sub+premcroyances}

2116 Les premiers ordinateurs, qui entrèrent en fonction à l'extrême fin
 2117 des années 1940 et durant les années 1950, étaient consacrés à des
 2118 travaux militaires ou scientifiques puisque, à cette époque,
 2119 on pensait [...]

2120

2121 \subsection{Comment l'informatique diffère des mathématiques}
 2122 J'aimerais à l'occasion de cette analyse attirer l'attention du
 2123 lecteur sur une question qui est une source constante de malentendus
 2124 au sujet de la programmation.

2125

2126 \subsubsection{Les preuves de programme}
 2127 L'écriture de programmes informatiques obéit à de tout autres
 2128 principes. Il convient de préciser cette affirmation pour la
 2129 préserver [...]

2130

2131 \section{Programmation dans le monde réel}
 2132 \subsection{La vraie nature de la programmation des ordinateurs}

2133

2134 Alors, comment s'écrivent les programmes informatiques? Et
 2135 d'ailleurs, qu'est-ce qu'une erreur\index{erreur} de programmation?
 2136 Ces questions sont liées et elles sont, bien sûr, au cœur de notre
 2137 préoccupation.

2138

2139 \subsection{Méthodes de programmation}
 2140 Un processeur quelconque est caractérisé par le jeu des actions
 2141 élémentaires qu'il est capable d'effectuer. Ces actions élémentaires
 2142 sont appelées les \emph{primitives} du processeur, ou, si le
 2143 processeur est une machine, les «instructions machine». Un [...]

2144

2145 \subsection{Méthodes de construction de programmes}
 2146 Nous avons décrit ci-dessus le processus élémentaire de la
 2147 programmation, celui qui consiste à écrire les instructions ou les
 2148 expressions qui vont composer un programme.

2149

2150 \subsubsection{La programmation structurée}
 2151 Le premier courant de pensée qui associa la recherche d'une syntaxe
 2152 claire et expressive à une organisation logique et commode des unités
 2153 de programme fut la \emph{programmation
 2154 structurée}\index{programmation!structurée} des années 1970, dont [...]

2155

2156 \subsubsection{La programmation par objets}
 2157 Après la programmation structurée vint un autre courant significatif:
 2158 la programmation par objets\index{programmation!par objets}, inventée
 2159 en Norvège à la fin des années 1960 par l'équipe de [...]

2160

2161 \subsubsection{Excès dans la pensée}
 2162 Il y a eu beaucoup de verbiage autour de l'aptitude supposée du
 2163 modèle [...]

2164 \end{document}

2165 \</livre>

4.20 The 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.

```

2166 (*ch0)
2167 \documentclass[12pt,a4paper]{report}
2168 \usepackage[tight,english,listfiles]{minitoc}
2169 \begin{document}
2170 \dominitoc
2171 \tableofcontents

```

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:

```

2172 \chapter*{Chapter One (starred)}
2173 \mtcaddchapter[Fake chapter one]
2174 \minitoc
2175 \section{Chap 1, section 1}
2176 That's right, folks -- we're close to the release of Firefox and
2177 Thunderbird 1.0 and, just like our last 1.0 release, we want to organize
2178 worldwide parties to celebrate.
2179
2180 Thanks to Dominik 'Aeneas' Schnitzer, we have an all-new and improved
2181 Mozilla Party Webtool 2.0. You can create your own party, or sign up for
2182 one already in progress -- and, in an improvement on Webtool 1.0,
2183 organizers can now edit and update party details. The tool allows you to
2184 organize a celebration in any of 243 countries, principalities,
2185 dominions and islands around the world. Never let it be said that we do
2186 things by halves around here.
2187
2188 \subsection{Chap 1, section 1, subsection 1}
2189 That's right, folks -- we're close to the release of Firefox and
2190 Thunderbird 1.0 and, just like our last 1.0 release, we want to organize
2191 worldwide parties to celebrate.
2192
2193 Thanks to Dominik 'Aeneas' Schnitzer, we have an all-new and improved
2194 Mozilla Party Webtool 2.0. You can create your own party, or sign up for
2195 one already in progress -- and, in an improvement on Webtool 1.0,
2196 organizers can now edit and update party details. The tool allows you to
2197 organize a celebration in any of 243 countries, principalities,
2198 dominions and islands around the world. Never let it be said that we do
2199 things by halves around here.

```

The second chapter is normal:

```

2200 \chapter{Chapter Two (numbered one)}
2201 \minitoc
2202 \section{Chapter 2, section 1}
2203 That's right, folks -- we're close to the release of Firefox and
2204 Thunderbird 1.0 and, just like our last 1.0 release, we want to organize

```

```

2205 worldwide parties to celebrate.
2206
2207 Thanks to Dominik 'Aeneas' Schnitzer, we have an all-new and improved
2208 Mozilla Party Webtool 2.0. You can create your own party, or sign up for
2209 one already in progress -- and, in an improvement on Webtool 1.0,
2210 organizers can now edit and update party details. The tool allows you to
2211 organize a celebration in any of 243 countries, principalities,
2212 dominions and islands around the world. Never let it be said that we do
2213 things by halves around here.
2214 \end{document}
2215 </ch0>

```

4.21 The 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 [159, page 58], modernized.

```

2216 <*tlc>
2217 \documentclass{article}

```

Dimensions of the text on the page:

```

2218 \setlength{\textwidth}{%
2219     {124.20126pt}
2220 \setlength{\textheight}{%
2221     {19\baselineskip}

```

We load the minitoc package and set some parameters (indentation, base font and depth) for the secttocs:

```

2222 \usepackage[listfiles]{minitoc}
2223 \setlength{\stcindent}{0pt}
2224 \mtcsetfont{secttoc}{*}%
2225     {\footnotesize}
2226 \mtcsetdepth{secttoc}{3}

```

We prepare the secttocs, without title, and the table of contents which is not printed:

```

2227 \begin{document}
2228 \dosecttoc[e]
2229 \faketableofcontents
2230
2231 \section{Afghanistan}
2232 \secttoc
2233 \subsection{Geography}
2234 \subsubsection{Total area}
2235     647,500 km2
2236 \subsubsection{Land area}
2237     647,500 km2
2238 \subsection{History} \ldots
2239
2240 \section{Albania}
2241 \secttoc
2242 \subsection{Geography}
2243 \subsubsection{Total area}
2244     28,750 km2
2245 \subsubsection{Land area}
2246     27,400 km2
2247 \subsection{History} \ldots
2248 \end{document}
2249 </tlc>

```

4.22 The `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 [6].

```

2250 \mu
2251 \documentclass[12pt]{report}
2252 \usepackage[tight,listfiles]{minitoc}
2253 \setlength{\mtcindent}{0pt}
2254 \usepackage{wrapfig}
2255 \newcommand{\LangSig}[1]{\textsc{[#1]}} % smallcaps
2256 \begin{document}
2257 \dominitoc
2258 \tableofcontents

```

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:

```

2259 \chapter{Mulspren}\label{chapter+mulspren}
2260 \begin{wrapfigure}{r}{0.5\linewidth}
2261 \begin{minipage}{\linewidth}
2262 \vspace{-2.\baselineskip}
2263 \minitoc
2264 \vspace{-1.\baselineskip}
2265 \end{minipage}
2266 \end{wrapfigure}

```

The remaining of the text:

```

2267 The previous chapter examined many end-user programming environments
2268 and found that most contain cognitive programming gulfs.
2269 These gulfs were often created when programming environments used
2270 multiple notations, and could manifest themselves in a variety of
2271 usability problems, ranging from users being unable to understand
2272 a program representation, to not wanting to execute their programs.
2273 Conversely, the previous chapter also found circumstances where multiple
2274 notations helped users understand programs.
2275 It concluded that there was a place for multiple notation programming
2276 environments, but developers had to be very careful to avoid creating
2277 programming gulfs.
2278 It concluded that there was a place for multiple notation programming
2279 environments, but developers had to be very careful to avoid creating
2280 programming gulfs.
2281
2282 This chapter introduces our programming environment, Mulspren.
2283 Mulspren was designed to avoid these gulfs and gain the potential
2284 benefits of multiple notations.
2285 Users program using two notations, one similar to English and one
2286 similar to conventional code.
2287 Changes in one notation are immediately reflected in the other notation,
2288 and users can move rapidly and seamlessly between the notations.

```

```

2289 This is programming using dual notations.
2290 When the program is executed, both notations are animated.
2291 Mulspren's language signature is \LangSig{Re/Wr/Wa + Re/Wr/Wa + Wa}.
2292
2293 Papers describing Mulspren have been published in~\cite{Wright02-2}
2294 and~\cite{Wright03-3}.
2295
2296 \section{section 1}
2297 \section{section 2 bla bla bla bla bla bla bla bla bla bla bla bla
2298 bla bla bla bla bla bla bla bla bla bla}
2299 \section{section 3}
2300 \section{section 4}
2301 \section{section 5 bla bla bla bla bla bla bla bla bla bla bla bla
2302 bla bla bla bla bla}

```

I tried to find some articles of the net to fill the citations:

```

2303 \begin{thebibliography}{1}
2304 \bibitem{Wright02-2}
2305 Tim Wright and Andy Cockburn.
2306 \newblock Mulspren: a multiple language simulation programming
2307 environment.
2308 \newblock In {\em HCC '02: Proceedings of the IEEE 2002 Symposia
2309 on Human Centric Computing Languages and Environments (HCC'02)},
2310 page 101, Washington, DC, USA, 2002. IEEE Computer Society.
2311
2312 \bibitem{Wright03-3}
2313 Tim Wright and Andy Cockburn.
2314 \newblock Evaluation of two textual programming notations for children.
2315 \newblock In {\em AUIC '05: Proceedings of the Sixth Australasian
2316 conference on User interface}, pages 55--62, Darlinghurst, Australia,
2317 Australia, 2005.
2318 Australian Computer Society, Inc.
2319 \end{thebibliography}
2320 \end{document}
2321 </mu>

```

4.23 The hir.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 report class.

<pre> 2322 (*hir) 2323 \documentclass[a4paper]{report} 2324 \usepackage{lipsum} 2325 \usepackage% </pre>	<pre> 2326 [tight,listfiles]{minitoc} 2327 \dominilot 2328 \begin{document} 2329 \listoftables 2330 \chapter{First chapter} 2331 \minilot 2332 \lipsum[1] 2333 \begin{table}[hb] 2334 \caption{My first visible table} 2335 \end{table} 2336 \lipsum[2] </pre>
--	--

```

2337 \begin{table}[ht]
2338 \caption{A second visible table}
2339 \end{table}
2340 \lipsum[3]

```

For the *first* hidden table, we add `\mtchideinmainlot` *before* its caption:

```

2341 \begin{table}[hb]
2342 \mtchideinmainlot
2343 \caption{My first hidden table}
2344 \end{table}
2345 \lipsum[4-6]
2346 \begin{table}[ht]
2347 \caption{A second hidden table}
2348 \end{table}
2349 \lipsum[7]
2350 \chapter{Second chapter}
2351 \minilot
2352 \lipsum[8]

```

For the *last* hidden table, we add `\endmtchideinmainlot` *after* its caption:

```

2353 \begin{table}[hb]
2354 \caption{My last hidden table}
2355 \endmtchideinmainlot
2356 \end{table}
2357 \lipsum[9]
2358 \begin{table}[ht]
2359 \caption{A third visible table}
2360 \end{table}
2361 \lipsum[10]
2362 \begin{table}[hb]
2363 \caption{A fourth visible table}
2364 \end{table}
2365 \lipsum[11]
2366 \end{document}
2367 </hir>

```

4.24 The 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.

```

2368 <#hia>
2369 \documentclass%
2370 [oneside,a4paper]{article}
2371 \usepackage{lipsum}
2372 \usepackage%
2373 [tight,insection,listfiles]%
2374 {minitoc}
2375 \dosectlot
2376 \begin{document}
2377 \listoftables
2378 \section{First section}
2379 \sectlot
2380 \lipsum[1]
2381 \begin{table}[hb]
2382 \caption{My first visible table}
2383 \end{table}
2384 \lipsum[2]
2385 \begin{table}[ht]

```

```

2386 \caption{A second visible table}
2387 \end{table}
2388 \lipsum[3]

```

For the *first* hidden table, we add `\mtchideinmainlot` *before* its caption:

```

2389 \begin{table}[hb]
2390 \mtchideinmainlot
2391 \caption{My first hidden table}
2392 \end{table}
2393 \lipsum[4-6]
2394 \begin{table}[ht]
2395 \caption{A second hidden table}
2396 \end{table}
2397 \lipsum[7]
2398 \section{Second section}
2399 \sectlot
2400 \lipsum[8]

```

For the *last* hidden table, we add
`\endmtchideinmainlot` *after* its
caption:

```
2401 \begin{table}[hb]
2402 \caption{My last hidden table}
2403 \endmtchideinmainlot
2404 \end{table}
2405 \lipsum[9]
```

```
2406 \begin{table}[ht]
2407 \caption{A third visible table}
2408 \end{table}
2409 \lipsum[10]
2410 \begin{table}[hb]
2411 \caption{A fourth visible table}
2412 \end{table}
2413 \lipsum[11]
2414 \end{document}
2415 </h1a>
```


Chapter 5

Messages

Contents

5.1	Messages from the <code>minitoc</code> package	138
5.1.1	Informative messages	138
5.1.1.1	Informative messages for hints	146
5.1.2	Warning messages	149
5.1.2.1	Warning messages for hints	153
5.1.3	Error messages	169
5.2	Messages from the <code>mtcoff</code> package	180
5.2.1	Warning messages	180
5.3	Message from the <code>mtcpatchmem</code> package	183

This chapter lists and comments the messages given by the `minitoc`, `mtcoff` and `mtcpatchmem` packages¹.

- 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, 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 `LATEX` 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 `LATEX` stops, so you can ask for help by typing “h”.

¹ The texts of the messages given in this chapter may slightly differ from the real text, because some messages contain some variable elements and the layout may differ.

- In this chapter, some words in the messages are typeset in italic characters; they represent the variable parts of the message:
 - *ARG1* The first argument of the command.
 - *ARG2* The second argument of the command.
 - *ARG3* The third argument of the command.
 - *CLASS* The name of the class of your document.
 - *counter* The name of a \LaTeX counter.
 - *document* The name of your document, without its `.tex` extension.
 - *extension* The extension part of the name of a file.
 - *file* The name of a file (often a minitoc auxiliary file, the `document.toc` file, the `document.lof` file, or the `document.lot` file).
 - *internal_name* The name of an internal macro redefined by a `\mtcset...` command.
 - *LANGUAGE* The name of the language (for `\mtcselectlanguage`).
 - *LINE* The number of the line in the source file.
 - *macro* The name of a minitoc font command redefined by `\mtcsetfont`.
 - *new_title* The new value of a title redefined by `\mtcsettitle`.
 - *NUMBER* The number of mini-tables of the given type in your document (when you are using short extensions and that number is greater than 99).
 - *PREPARATION* The name of a minitoc preparation command.
 - *sequence* A sequence of commands used to redefine an internal macro via a `\mtcset...` command.
 - *STRING* A string of characters, often part of a command name.
 - *value* The new value of a counter.

5.1 Messages from the minitoc package

5.1.1 Informative messages

Package: minitoc 2006/08/03 v47 Package minitoc (JPFD)

This is the announce message of the minitoc package, with its name, date and version. “JPFD” are my initials. This message has no identifier because it is emitted by `\ProvidesPackage`.

```
Package minitoc Info: <I0001>
(minitoc)           *** minitoc package, version 47 ***.
```

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.

```

Package minitoc Info: <I0006>
(minitoc)           document.extension is empty on input line LINE.

```

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.

```

Package minitoc Info: <I0007>
(minitoc)           Horizontal rules are activated
(minitoc)           for the ARGIs on input line LINE.

```

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

```

Package minitoc Info: <I0008>
(minitoc)           Horizontal rules are inhibited
(minitoc)           for the ARGIs on input line LINE.

```

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

```

Package minitoc Info: <I0009>
(minitoc)           Listing minitoc auxiliary files
(minitoc)           Creating the document.maf file.

```

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 49.

```

Package minitoc Info: <I0010>
Package minitoc Info: The LANGUAGE language is selected.
(minitoc)           on input line LINE.

```

The *LANGUAGE.mld* file has been successfully loaded for the *LANGUAGE* language² 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.

```
Package minitoc Info: <I0011>
(minitoc)           LANGUAGE language object selected.
(minitoc)           on input line LINE.
```

The `\mtcselectlanguage` macro has successfully (indirectly) loaded the `LANGUAGE.mlo` minitoc object file.

```
Package minitoc Info: <I0012>
(minitoc)           Long extensions (Unix-like) will be used.
```

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

```
Package minitoc Info: <I0013>
(minitoc)           \mtcsetdepth redefines the counter
(minitoc)           "counter" as "value" on input line LINE.
```

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

```
Package minitoc Info: <I0014>
(minitoc)           \mtcsetfeature redefines the macro
(minitoc)           "\internal_name" as
(minitoc)           "\sequence" on input line LINE.
```

The `\mtcsetfeature` macro has redefined the internal macro *internal_name* with the given *sequence*.

```
Package minitoc Info: <I0015>
(minitoc)           \mtcsetfont redefines the macro
(minitoc)           "macro" as "sequence" on input line LINE.
```

The `\mtcsetfont` command redefines the (old style) *macro* by the given *sequence* of font commands.

```

Package minitoc Info: <I0016>
(minitoc)           \mtcsetformat redefines the macro
(minitoc)           "\internal_name" as "ARG3" on input line LINE.

```

The macro `\mtcsetformat` redefines an internal macro with the value given by its third argument.

```

Package minitoc Info: <I0017>
(minitoc)           \mtcsettitle redefines the macro
(minitoc)           "internal_name" as
(minitoc)           "new_title" on input line LINE.

```

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

```

Package minitoc Info: <I0018>
(minitoc)           \mtcsettitlefont redefines the macro
(minitoc)           "\internal_name" as
(minitoc)           "sequence" on input line LINE.

```

The `\mtcsettitlefont` macro redefines the (old style) `\internal_name` macro which the given sequence *sequence*.

```

Package minitoc(hints) Info: <I0019>
(minitoc(hints))    No hints have been written
(minitoc(hints))    in the document.log file.

```

The hints package option has detected no potential problem.

```

Package minitoc Info: <I0020>
(minitoc)           old version of the memoir class.

```

The version of the memoir class is old. The minitoc package does not need to patch this class.

```

Package minitoc Info: <I0021>
(minitoc)           Page numbers are activated
(minitoc)           for the ARGIs on input line LINE.

```

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

```

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

```

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

```

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

```

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

```

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*.

```

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*.

```

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*.

```
Package minitoc Info: <I0027>
(minitoc)             recent version of the memoir class.
```

The version of the memoir class is recent. The minitoc package will try to patch it.

```
Package minitoc Info: <I0028>
(minitoc)             section level macros available.
```

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

```
Package minitoc Info: <I0029>
(minitoc)             section level macros NOT available.
```

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

```
Package minitoc Info: <I0030>
(minitoc)             the memoir class is loaded:
(minitoc)             compatibility attempted.
```

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

```
Package minitoc Info: <I0031>
(minitoc)             ==> this version is configured for UNIX-like
(minitoc)             (long extensions) file names.
```

The autoconfiguration has detected that your operating uses UNIX-like (long extensions) file names.

```

Package minitoc Info: <I0032>
(minitoc)           This version of the memoir class uses
(minitoc)           a version of \chapter which is
(minitoc)           incompatible with the minitoc package.
(minitoc)           We try to patch.

```

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

```

Package minitoc Info: <I0033>
(minitoc)           Writing document.extension.

```

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

```

Package minitoc Info: <I0034>
(minitoc)           PREPARING MINILOFS FROM file on input line LINE.

```

A `\dominilof` command prepares the minilof auxiliary files for minilofs from *file*.

```

Package minitoc Info: <I0035>
(minitoc)           PREPARING PARTLOFS FROM file on input line LINE.

```

A `\dopartlof` command prepares the partlof auxiliary files for partlofs from *file*.

```

Package minitoc Info: <I0036>
(minitoc)           PREPARING SECTLOFS FROM file on input line LINE.

```

A `\dosectlof` command prepares the sectlof auxiliary files for sectlofs from *file*.

```
Package minitoc Info: <I0037>
(minitoc)             PREPARING MINILOTS FROM file on input line LINE.
```

A \dominilot command prepares the minilot auxiliary files for minilots from *file*.

```
Package minitoc Info: <I0038>
(minitoc)             PREPARING PARTLOTS FROM file on input line LINE.
```

A \dopartlot command prepares the partlot auxiliary files for partlots from *file*.

```
Package minitoc Info: <I0039>
(minitoc)             PREPARING SECTLOTS FROM file on input line LINE.
```

A \dosectlot command prepares the sectlot auxiliary files for sectlots from *file*.

5.1.1.1 Informative messages for hints

```
Package minitoc(hints) Info: <I0040>
(minitoc(hints))      The ‘‘abstract’’ package has been
(minitoc(hints))      loaded with the ‘‘addtotoc’’ option.
(minitoc(hints))      You need to look at the
(minitoc(hints))      documentation to adjust.
```

As you are using the `abstract` package with its `addtotoc` option, you should look at the `minitoc` package documentation for specific precautions. See section 2.27 on page 67.

```
Package minitoc(hints) Info: <I0041>
(minitoc(hints))      --- The amsbook class is loaded.
(minitoc(hints))      See the minitoc package documentation
(minitoc(hints))      for specific precautions.
```

As you are using the `amsbook` class, you should look at the `minitoc` package documentation for specific precautions. See section 2.24 on page 63.

```

Package minitoc(hints) Info: <I0042>
(minitoc(hints))          --- The appendix package is loaded.
(minitoc(hints))          See the minitoc package documentation
(minitoc(hints))          for specific precautions.

```

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 60.

```

Package minitoc(hints) Info: <I0043>
(minitoc(hints))          --- The KOMAScript CLASS class is loaded.
(minitoc(hints))          See the minitoc package documentation
(minitoc(hints))          for specific precautions.

```

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 47. The classes involved here are `scrbook`, `scrreprt`, and `scrarcl`, i.e., the KOMA-Script classes [164] compatible with `minitoc`.

```

Package minitoc(hints) Info: <I0044>
(minitoc(hints))          --- The memoir class is loaded.
(minitoc(hints))          See the minitoc package documentation
(minitoc(hints))          for specific precautions.

```

As you are using the `memoir` class, you should look at the `minitoc` package documentation for specific precautions. See section 2.22 on page 62.

```

Package minitoc(hints) Info: <I0045>
(minitoc(hints))          The \PREPARATION command
(minitoc(hints))          has been invoked more than once
(minitoc(hints))          on input line LINE.

```

A `minitoc` preparation command has been invoked more than once.

```

Package minitoc(hints) Info: <I0046>
(minitoc(hints))          --- The tocbibind package is loaded.
(minitoc(hints))          See the minitoc package documentation
(minitoc(hints))          for specific precautions.

```

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 47.

```

Package minitoc(hints) Info: <I0047>
(minitoc(hints))          --- The tocloft package is loaded.
(minitoc(hints))          See the minitoc package documentation
(minitoc(hints))          for specific precautions.

```

As you are using also the tocloft package, you should look at the minitoc package documentation for specific precautions. See section 2.21 on page 61.

```

Package minitoc(hints) Info: <I0048>
(minitoc(hints))          Using \mtcprepare may induce some
(minitoc(hints))          hints about the preparation commands,
(minitoc(hints))          because it invokes ALL the preparation
(minitoc(hints))          commands allowed by the document class,
(minitoc(hints))          without any previous check.

```

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.

```

Package minitoc(hints) Info: <I0049>
(minitoc(hints))          ==> You requested the hints option
(minitoc(hints))          Some hints are eventually given below.

```

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.

5.1.2 Warning messages

```
Package minitoc Warning: <W0001>
(minitoc)                \chapter and \section are undefined.
(minitoc)                Cannot use \mtcfixglossary without
(minitoc)                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
(minitoc)                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
(minitoc)                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
(minitoc)                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
(minitoc)                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
(minitoc)                with the [part] optional argument,
(minitoc)                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
(minitoc)                with the [part] optional argument,
(minitoc)                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 `\dominilof` command. Please check if you have called `\dominilof` 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 `\dominilot` command. Please check if you have called `\dominilot` 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)                PARTTocs 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
(minitoc)                (8+3) file names.
```


The autofiguration has found that the operating systems use 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 `shorttext` 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 `shorttext` 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 try to continue. It would be better to update your \LaTeX installation.

```
Package minitoc Warning: <W0022>
Package minitoc Warning: 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.

5.1.2.1 Warning messages for hints

```
Package minitoc(hints) Warning: <W0023>
(minitoc(hints))          --- It may be the consequence
(minitoc(hints))          of loading the ‘‘hyperref’’ package.
```

Some sectionning 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 sectionning commands only *before* loading the minitoc package. See section 2.17 on page 59.

```

Package minitoc(hints) Warning: <W0024>
(minitoc(hints))                Some hints have been written
(minitoc(hints))                in the document.log file.

```

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.

```

Package minitoc(hints) Warning: <W0025>
(minitoc(hints))                --- The alphanum package is loaded.
(minitoc(hints))                It is incompatible
(minitoc(hints))                with the minitoc package.

```

You are using the `alphanum` package which is incompatible with the `minitoc` package. The compilation can continue, but the result could be unsatisfactory.

```

Package minitoc(hints) Warning: <W0026>
(minitoc(hints))                --- The amsart class is loaded.
(minitoc(hints))                It is incompatible
(minitoc(hints))                with the minitoc package.

```

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 sectionning 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 sectionning 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>
Package 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 sectionning 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 sectionning 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: <W0031>
(minitoc(hints))          --- The placeins package is loaded
(minitoc(hints))          without the section option,
(minitoc(hints))          but minitoc used the insection option
(minitoc(hints))          which implies it. Try to inverse the
(minitoc(hints))          loading order and use coherent options.
(minitoc(hints))          You may have got a message
(minitoc(hints))          ! LaTeX Error: Option clash for package placeins.

```



You are using the `placeins` package, but without its `section` options, 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.

```
Package minitoc(hints) Warning: <W0032>
(minitoc(hints))              --- The placeins package loaded is
(minitoc(hints))              too old. It should have a version
(minitoc(hints))              date of 2005/04/18 at least.
```

You are using an obsolete version of the `placeins` package. Please update it from the CTAN archives or a recent distribution.

```
Package minitoc(hints) Warning: <W0033>
(minitoc(hints))              The caption package should be
(minitoc(hints))              loaded BEFORE the minitoc package.
```

The `caption` package alters some commands and must be loaded *before* the `minitoc` package. See section 2.31 on page 69.

```
Package minitoc(hints) Warning: <W0034>
(minitoc(hints))              The caption2 package should be
(minitoc(hints))              loaded BEFORE the minitoc package.
```

The `caption2` package alters some commands and must be loaded *before* the `minitoc` package. See section 2.31 on page 69.

```
Package minitoc(hints) Warning: <W0035>
(minitoc(hints))              The ccaption package should be
(minitoc(hints))              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 69.

```

Package minitoc(hints) Warning: <W0036>
(minitoc(hints))               The mcaption package should be
(minitoc(hints))               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 69.

```

Package minitoc(hints) Warning: <W0037>
(minitoc(hints))               The sectsty package should be
(minitoc(hints))               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 67.

```

Package minitoc(hints) Warning: <W0038>
(minitoc(hints))               The varsects package should be
(minitoc(hints))               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 69.

```

Package minitoc(hints) Warning: <W0039>
(minitoc(hints))               --- The \section command is altered
                                after minitoc.

```

Some packages alter the sectionning 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 sectionning 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
(minitoc(hints))               with the minitoc package.

```

You are trying to use also the `titletoc` package, but it is incompatible with the `minitoc` package. See note 17 on page 50.

```
Package minitoc(hints) Warning: <W0041>
(minitoc(hints))              You have attempted to insert
                               empty minilofs.
```

You have attempted to insert empty minilofs. 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.

```
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
                               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
                               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
                               command: \firstsectionis.
```

You have used an obsolete command (`\firstsectionis`). You should remove it.

```
Package minitoc(hints) Warning: <W0053>
(minitoc(hints))              You have used short extensions
(minitoc(hints))              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. *NUMBER* is the number of effective chapters in your document. See section 1.9 on page 51.

```
Package minitoc(hints) Warning: <W0054>
(minitoc(hints))                You have used short extensions
(minitoc(hints))                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. *NUMBER* is the number of effective parts in your document. See section 1.9 on page 51.

```
Package minitoc(hints) Warning: <W0055>
(minitoc(hints))                You have used short extensions
(minitoc(hints))                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. *NUMBER* is the number of effective sections in your document. See section 1.9 on page 51.

```
Package minitoc(hints) Warning: <W0056>
(minitoc(hints))                You are using \dosectlof and/or
(minitoc(hints))                \dosectlot, \sectlof and/or \sectlot,
(minitoc(hints))                hence the ‘‘insection’’ package
(minitoc(hints))                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 `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,
(minitoc(hints))              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,
(minitoc(hints))              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,
(minitoc(hints))              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,
(minitoc(hints))              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,
(minitoc(hints))              but not \listoffigures nor
(minitoc(hints))              \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
(minitoc(hints))              \listoftables nor
(minitoc(hints))              \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
(minitoc(hints))              \tableofcontents
(minitoc(hints))              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
(minitoc(hints))              \listoffigures
(minitoc(hints))              nor \fakelistoffigures.
```

You have tried to insert some partlofs (via `\partlof`), but the *document.lof* file is not available because you have not invoked `\listoffigures` nor `\fakelistoffigures`.

```
Package minitoc(hints) Warning: <W0070>
(minitoc(hints))              You have used \partlot but not
(minitoc(hints))              \listoftables
(minitoc(hints))              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
(minitoc(hints))              \tableofcontents
(minitoc(hints))              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
(minitoc(hints))      \listoffigures
(minitoc(hints))      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
(minitoc(hints))      \listoftables
(minitoc(hints))      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
(minitoc(hints))      \tableofcontents
(minitoc(hints))      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 use 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 use 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 use 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 use 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 use one of them.

```
Package minitoc(hints) Warning: <W0080>
(minitoc(hints))              You have used \dominilot
(minitoc(hints))              but not \minilot.
```

You have prepared some minilots (via `\dominilot`), but you never use one of them.

```
Package minitoc(hints) Warning: <W0081>
(minitoc(hints))              You have used \dosecttoc
(minitoc(hints))              but not \secttoc.
```

You have prepared some secttocs (via `\dosecttoc`), but you never use one of them.

```
Package minitoc(hints) Warning: <W0082>
(minitoc(hints))                You have used \dosectlof
(minitoc(hints))                but not \sectlof.
```

You have prepared some sectlofs (via `\dosectlof`), but you never use one of them.

```
Package minitoc(hints) Warning: <W0083>
(minitoc(hints))                You have used \dosectlot
(minitoc(hints))                but not \sectlot.
```

You have prepared some sectlots (via `\dosectlot`), but you never use one of them.

```
Package minitoc(hints) Warning: <W0084>
(minitoc(hints))                --- The placeins package is loaded
(minitoc(hints))                with the above option,
(minitoc(hints))                but minitoc used the insection option
(minitoc(hints))                which is incompatible with it.
(minitoc(hints))                Try to remove the above option and
(minitoc(hints))                use coherent options.
```

You are using the `placeins` package, but with its `above` options, while `minitoc` is called with its `insection` option which is incompatible with it. See page 29, near a “dangerous bend” symbol like the one shown in the margin.



```
Package minitoc(hints) Warning: <W0085>
(minitoc(hints))                --- The placeins package is loaded
(minitoc(hints))                with the below option,
(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 coherent options.
```

You are using the `placeins` package, but with its `below` options, while `minitoc` is called with its `insection` option which is incompatible with it. See page 29, near a “dangerous bend” symbol like the one shown in the margin.



```
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 72.

```
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 72.

```
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 sectionning commands and must be loaded *before* the minitoc package. See section 2.40 on page 72.

```
Package minitoc(hints) Warning: <W0089>
(minitoc(hints))               The sfheaders package should be
(minitoc(hints))               loaded BEFORE the minitoc package.
```

The sfheaders package alters the sectionning commands and must be loaded *before* the minitoc package. See section 2.41 on page 72.

```
Package minitoc(hints) Warning: <W0090>
(minitoc(hints))               The alnumsec package should be
(minitoc(hints))               loaded BEFORE the minitoc package.
```

The alnumsec package alters the sectionning commands and must be loaded *before* the minitoc package. See section 2.42 on page 73.

```

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 73.

5.1.3 Error messages

```

! Package minitoc Error: <E0001>
(minitoc)                But \part is undefined.
See the minitoc package documentation for explanation.
Type H <return> for immediate help.
? h
\mtcfixglossary not usable

```

There are no adequate sectionning 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.
? h
\mtcfixindex not usable

```

There are no adequate sectionning 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.
? h
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.

```
! 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
```

Some `mtchideinmainlot` environments are incorrectly imbricated (overlapping), so the hiding in the main list of tables will be strange.

```
! 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
```

Some `mtchideinmaintoc` environments are incorrectly imbricated (overlapping), so the hiding in the main table of contents will be strange.

```
! 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
```

The `\mtcselectlanguage` 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>
(minitoc)          LANGUAGE is not a known minitoc
(minitoc)          language object file (.mlo),
(minitoc)          LANGUAGE.mlo 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

```

The `\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>
(minitoc)          \mtcsetdepth attempts to use
(minitoc)          an undefined counter (ARG/depth).
See the minitoc package documentation for explanation.
Type H <return>  for immediate help.
Correct the source code.
Type <return> and rerun LaTeX

```

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>
(minitoc)          \mtcsetdepth has a wrong first argument
(minitoc)          (ARGI).
(minitoc)          It should be a mini-table type
(minitoc)          (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 `\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 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 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, pagestyle).
See the minitoc package documentation for explanation.
Type 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`, or `thispagestyle`.

```

! Package minitoc Error: <E0013>
(minitoc)          \mtcsetfont 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 H <return> for immediate help.
? h
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>
(minitoc)          \mtcsetfont has a wrong second argument
(minitoc)          (ARG2).
(minitoc)          It should be a sectionning level
(minitoc)          (part...subparagraph) or * .
See the minitoc package documentation for explanation.
Type H <return> for immediate help.
? h
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>
(minitoc)          \mtcsetformat 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 H <return> for immediate help.
? h
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`).

```

! Package minitoc Error: <E0016>
(minitoc)                \mtcsetformat has a wrong second argument
(minitoc)                (ARG2).
(minitoc)                It should be a formatting param choosen from:
(minitoc)                pagenumwidth, tocrightmargin, dotinterval.
See the minitoc package documentation for explanation.
Type H <return> for immediate help.
? h
Correct the source code.
Type <return> and rerun LaTeX

```

The second argument of the `\mtcsetformat` macro is wrong. It should be one of the following keywords: `pagenumwidth`, `tocrightmargin`, or `dotinterval`.

```

! Package minitoc Error: <E0017>
(minitoc)                \mtcsetpagenumbers has a wrong first
(minitoc)                argument (ARG1)..
See the minitoc package documentation for explanation.
Type H <return> for immediate help.
? h
It should be a mini-table type
(minitoc)                (parttoc...sectlot)
Correct the source code.
Type <return> and rerun LaTeX

```

The first argument of the `\mtcsetpagenumbers` macro must be a type of minitable (`parttoc`, ..., `sectlot`).

```

! Package minitoc Error: <E0018>
(minitoc)                \mtcsetpagenumbers has a wrong second
(minitoc)                argument (ARG2).
See the minitoc package documentation for explanation.
Type H <return> for immediate help.
? h
It should be a boolean value (0/1, yes/no, on/off)
Correct the source code.
Type <return> and rerun LaTeX

```

The second argument of the `\mtcsetpagenumbers` must be a keyword chosen in the following lists:

- 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, N, false, FALSE, faux, FAUX, f, F, non, NON, and 0.

```
! Package minitoc Error: <E0019>
(minitoc)          \mtcsetrules has a wrong first argument
(minitoc)          (ARG1).
See the minitoc package documentation for explanation.
Type H <return> for immediate help.
? h
It should be a mini-table type
(minitoc)          (parttoc...sectlot)
Correct the source code.
Type <return> and rerun LaTeX
```

The first argument of the `\mtcsetrules` is incorrect. It should be a mini-table type (parttoc, ..., sectlot).

```
! Package minitoc Error: <E0020>
(minitoc)          \mtcsetrules has a wrong second argument
(minitoc)          (ARG2).
See the minitoc package documentation for explanation.
Type H <return> for immediate help.
? h
It should be a boolean value (0/1, yes/no, on/off)
Correct the source code.
Type <return> and rerun LaTeX
```

The second argument of the `\mtcsetrules` must be a keyword chosen in the following lists:

- 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, N, false, FALSE, faux, FAUX, f, F, non, NON, and 0.

```

! Package minitoc Error: <E0021>
(minitoc)          \mtcsettitle 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 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>
(minitoc)          \mtcsettitlefont 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 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>
(minitoc)          The macro \mtcsetfeature 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 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 sectionning 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.

```
! Package minitoc Error: <E0031>
(minitoc)                Unbalanced mtchideinmaintoc environment.
See the minitoc package documentation for explanation.
Type H <return> for immediate help.
? h
The hiding in main ToC could be incorrect
```

A `mtchideinmaintoc` environment is unbalanced, so the hiding in the main table of contents could be incorrect.

```
! Package minitoc Error: <E0032>
(minitoc)                You are using the \mtcloadmlo command
(minitoc)                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.

```
! Package minitoc Error: <E0033>
(minitoc)                The macro \mtcsettitle uses
(minitoc)                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`.

```

! Package minitoc Error: <E0034>
(minitoc)          The macro \mtcsettitlefont uses
(minitoc)          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.

```

! Package minitoc Error: <E0035>
(minitoc)          You have used the 'insection' option in
(minitoc)          a document where chapters are defined.
(minitoc)          This is not compatible: option ignored.
Type H <return> for immediate help.
? h
Remove this option.
Type <return> and rerun LaTeX

```

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.

5.2 Messages from the mtcoff package

The mtcoff package gives only warning messages; their numbers begin with F.

5.2.1 Warning messages

```

Package mtcoff Warning: <F0001>
(mtcoff)          \addstarredchapter{...} should be replaced
(mtcoff)          by \addcontentsline{toc}{chapter}{...}
(mtcoff)          on input line LINE.

```

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: <F00006>
(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: <F00007>
(mtcoff)                You should scan (backwards) your .log
(mtcoff)                file to find some commands needing
(mtcoff)                to be replaced if you decide to
(mtcoff)                DEFINITELY stop using minitoc for this
(mtcoff)                document. It is more wise to keep the
(mtcoff)                \usepackage lines for minitoc and mtcoff
(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: <F00008>
(mtcoff)                The macro \string\kernafterSTRING
(mtcoff)                should not be used out of context
(mtcoff)                on line LINE.

```

You are using one of the `\kernafterSTRING` macro 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.

5.3 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 [410](#).

Chapter 6

Jargon

Tables

6.1	Category codes	190
6.2	Encoding schemes implemented in CJK	191
6.3	Document classes	192
6.4	Depths for sectionning commands	193
6.5	Most common font encodings	204
6.6	Most common font families	205
6.7	Most common font series	205
6.8	Most common font shapes	205
6.9	The five font parameters for some fonts	205
6.10	Author commands for fonts	206

This chapter attempts to explain some terms used in this documentation, and describes some useful files and suffixes. Many data come from various documentations of cited packages, and from [3, 4, 87, 89, 124, 133, 139, 140, 141, 149, 150, 151, 159, 164, 166, 167, 201].

- .aux** The suffix of the name for an *auxiliary* file of a \LaTeX document. It carries some information from a \LaTeX run to the next.
- .cls** The suffix for the name of a document class file, loaded via `\documentclass`.
- .dtx** The suffix of the name for a documented source file of a \LaTeX package or class. This file is often associated with a `.ins` file to generate the package or class. Compiling a `.dtx` file with \LaTeX generates the documentation.
- .F** The base suffix of the name for a minilof file when short extensions (suffixes) are used. The full suffix is `.Fnn` where *nn* is the absolute number of the minilof.
- .G** The base suffix of the name for a partlof file when short extensions (suffixes) are used. The full suffix is `.Gnn` where *nn* is the absolute number of the partlof.

.H	The base suffix of the name for a sectlof file when short extensions (suffixes) are used. The full suffix is <code>.Hnn</code> where <i>nn</i> is the absolute number of the sectlof.
.ins	The suffix of the name for an installation file of a \LaTeX package or class. When compiled with \LaTeX , it extracts the files of the package or class from an <code>.dtx</code> file.
.lof	The suffix of the name of the “list of figures” file.
.log	The suffix of the name of the log file (compilation report).
.lot	The suffix of the name of the “list of tables” file.
.M	The base suffix of the name for a minitoc file when short extensions (suffixes) are used. The full suffix is <code>.Mnn</code> where <i>nn</i> is the absolute number of the minitoc.
.maf	The suffix of the name of the file generated by the <code>listfiles</code> package option. This file contains the list of the minitoc auxiliary files.
.mld	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.
.mlf	The base suffix of the name for a minilof file when long extensions (suffixes) are used. The full suffix is <code>.mlfnn</code> where <i>nn</i> is the absolute number of the minilof.
.mlo	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 <code>.mld</code> file; hence the <code>.mld</code> file must load a <code>.mlo</code> file.
.mlt	The base suffix of the name for a minilot file when long extensions (suffixes) are used. The full suffix is <code>.mltnn</code> where <i>nn</i> is the absolute number of the minilot.
.mtc	The base suffix of the name for a minitoc file when long extensions (suffixes) are used. The full suffix is <code>.mtcnn</code> where <i>nn</i> is the absolute number of the minitoc.
.P	The base suffix of the name for a parttoc file when short extensions (suffixes) are used. The full suffix is <code>.Pnn</code> where <i>nn</i> is the absolute number of the parttoc.
.plf	The base suffix of the name for a partlof file when long extensions (suffixes) are used. The full suffix is <code>.plfnn</code> where <i>nn</i> 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 <code>.pltnn</code> where <i>nn</i> 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 <code>.ptcnn</code> where <i>nn</i> 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.
- .slf** The base suffix of the name for a sectlof file when long extensions (suffixes) are used. The full suffix is *.slfnn* where *nn* is the absolute number of the sectlof.
- .slt** The base suffix of the name for a sectlot file when long extensions (suffixes) are used. The full suffix is *.slt nn* where *nn* is the absolute number of the sectlot.
- .stc** The base suffix of the name for a secttoc file when long extensions (suffixes) are used. The full suffix is *.stc nn* 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 $\mathrm{T\!E\!X}$ or $\mathrm{L\!A\!T\!E\!X}$ 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.
- 2c.tex** An example file showing the use of the `minitoc` package with a two columns page layout.
- 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 [208] (by Peter R. WILSON) needs some precautions if used with its `addtotoc` option.
- add.bib** A small bibliographic data base for the `add.tex` and `addsec.tex` example documents.
- add.tex** An example document showing how to use `\mtcaddchapter` and the `tocbibind` package [210] with `minitoc`. See section 4.3 on page 96.

addsec.tex An example document showing how to use `\mtcaddsection` and the `tocbibind` package [210] with `minitoc`. See section 4.4 on page 101. It also shows how it is challenging to manage the mini-lists of floats at the section level.

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`, `\mtcaddpart`, `\mtcaddchapter`, `\mtcaddsection`, `\mtcfixglossary`, `\mtcfixindex`, `mtchideinmaintoc`, `mtchideinmainlof`, and `mtchideinmainlot`. It is *strongly recommended* that the user verifies the result of such adjustments in the final document.



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 45.

alnumsec The `alnumsec` package [128] allows you to use alphanumeric section numbering, e.g. A. Introduction; III. International Law. It’s output is similar to the `alphanum` package (part of the `jura` class [77]), but you can use the standard \LaTeX sectioning commands. Thus it is possible to switch numbering 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 51 and section 2.42 on page 73).

alphanum The `alphanum` package, which is part of the specialized `jura` class [77], by Felix BRAUN, is incompatible with the `minitoc` package.

\mathcal{AMS} the American Mathematical Society. This society has developed some document classes: `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 (\mathcal{AMS}). This class is incompatible with the `minitoc` package.

amsbook, amsbook.cls A document class for books, provided by the American Mathematical Society (\mathcal{AMS}). This class is compatible with the `minitoc` package, but needs some precautions. See section 2.24 on page 63.

amsproc, amsproc.cls A document class for conference proceedings, provided by the American Mathematical Society (\mathcal{AMS}). This class is incompatible with the `minitoc` package.

Antomega Antomega [127] (by Alexej M. KRYUKOV and Dmitry IVANOV) is a language support package for **Lambda** (Λ), based on the original `omega.sty` file of the Omega project (Ω). 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 [209] (by Peter R. WILSON) provides various ways of formatting the titles of appendices. Also `(sub)appendices` environments are provided that can be used, for example, for per chapter/section appendices. If this package is used with `minitoc`, some precautions are needed (see section 2.20 on page 60).

app-mem.tex An example file showing the use of the `appendices` environment in a memoir class document with the `minitoc` package.

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.

Arabi `Arabi` [116] is a system (by Youssef JABRI) to prepare \LaTeX documents in the arabic or farsi languages. The titles in `arabi.mld` and `farsi3.mld` come from the `arabic.ldf` and `farsi.ldf` files of this system.

ArabTeX `ArabTeX` [130, 131] is a package (by Klaus 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 `ArabTeX`, while the titles of `hebrew2.mld` comes from `Babel`.

ArmTeX `ArmTeX` [90] is a package (by Serguei DACHIAN and others) to prepare \LaTeX documents in the armenian language. The titles in `armenian.mld` come from `ArmTeX`.

article A standard \LaTeX document class. It has sectioning 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 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 `shorttext` 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 `document.aux` (for cross-reference labels, counters, etc.), `document.toc` for the table of contents, `document.lof` for the list of figures, and `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 *document.extension*, the table 1.11 on page 52 lists the possible extensions.

babel The babel package [30, 31] (by Johannes BRAAMS and others) is a large package useful to write L^AT_EX documents in many languages, not only english. Many titles for mini-tables come from the babel package.

BangT_EX A package for typesetting documents in the Bangla language using the T_EX/L^AT_EX systems; see [169].

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 45.

BibT_EX A program by Oren PATASHNIK to make bibliographies in L^AT_EX documents. Distributed with L^AT_EX. See [151, 171, 172].

bibtopic A L^AT_EX package [12] 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 [104] 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, overcite and KOMA-Script classes [164].

bo.tex An example file showing the use of the minitoc package with a two columns page layout and using the tocloft package [207].

book A standard L^AT_EX document class. 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).

cappuccino See “minutes” below.

captcont The captcont package [85] 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 [86] in particular.

caption The caption package [186] provides many ways to customize the captions in floating environments such figure and table and cooperates with many other packages.

caption2 The¹ caption2 package [185] used to be an experimental side-version of the regular caption package [186] and has been superseded by the new release of the regular caption package version 3.0 in December 2003.

¹ This text comes from the documentation of the caption package. The caption and caption2 packages have the same author, Axel SOMMERFELDT.

`caption2` is still 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.

catalog A plain text file which contains basic infos about the `minitoc` package (abstract, date, author, version, licence).

catcode Short for “category code”. In² 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;

Table 6.1: Category codes

Category	Meaning
0	Escape character (<code>\</code> usually)
1	Beginning of group (<code>{</code> usually)
2	End of group (<code>}</code> usually)
3	Math shift (<code>\$</code> usually)
4	Alignment tab (<code>&</code> usually)
5	End of line (<code>\return</code> usually)
6	Parameter (<code>#</code> usually)
7	Superscript (<code>^</code> usually)
8	Subscript (<code>_</code> usually)
9	Ignored character (<code>\null</code> usually)
10	Space (<code>_</code> usually)
11	Letter (<code>A, \dots, Z</code> and <code>a, \dots, z</code>)
12	Other character (none of the above or below)
13	Active character (<code>~</code> usually)
14	Comment character (<code>%</code> usually)
15	Invalid character (<code>\delete</code> usually)

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.

ccaption The `ccaption` package [212] provides commands for “continuation captions”, unnumbered captions, and a legend heading for any environment.

² This definition is taken from “*The \TeX book*” [121].

Table 6.2: Encoding schemes implemented in CJK

Encoding	1 byte	2 bytes	3 bytes
GB	0xA1–0xF7	0xA1–0xFE	—
Big 5	0xA1–0xF9	0x40–0xFE	—
JIS	0xA1–0xF4	0xA1–0xFE	—
SJIS	0xA1–0xFE	0x40–0xFC	—
KS	0xA1–0xFD	0xA1–0xFE	—
UTF 8	0xC0–0xEF	0x80–0xBF	0x80–0xBF
CNS	0xA1–0xFE	0xA1–0xFE	—

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.

ch0.tex An example file showing the use of the `minitoc` package in a document with a starred first chapter.

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 (`\firstchapteris` 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 46.

chapterbib The `chapterbib` package [7] 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 de default. The opposite package option (`nocheckfiles`) prints even the empty mini-tables, which look ugly. See section 9.74.2 on page 364.

CJK The CJK system [143, 144] (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 Hangûl or Hanja fonts), and thai, plus some variants of russian. The titles of mini-tables for these languages come from some files CJK files and inserted in `.mld` files when possible, or in `.mlo` files when the encoding is incompatible with the `.ins/.dtx` mechanism; the `.mld` 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 \TeX 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

L^AT_EX; 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 [149, 150]; Ken LUNDE 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 1.4.12 on page 41.

class The **class**³ is the first information L^AT_EX needs to know when processing an input file; it is the type of document the author wants to create. This is specified with the `\documentclass` command.

`\documentclass[options]{class}`

Here *class* specifies the type of document to be created. Table 6.3 lists the standard document classes. The L^AT_EX 2_ε distribution provides additional classes for other documents, including letters and slides, but the `minitoc` package has not been tested with all these classes. The *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 56.

Table 6.3: Document classes

Class	Usage
<code>article</code>	For articles in scientific journals, presentations, short reports, program documentation, invitations, ...
<code>report</code>	For longer reports containing several chapters, small books, PhD theses, ...
<code>book</code>	For real books
<code>slides</code>	For slides; the class uses big sans serif letters. You might want to consider using BeamerT _E X ^a instead. Do not use <code>minitoc</code> with these classes.

^a <http://www.dante.de/tex-archive/macros/latex/contrib/beamer/doc/beameruserguide.pdf>

CMR For “Computer Modern Roman”. The roman subset of the Computer Modern fonts. See “Computer Modern” below.

comp.text.tex The Usenet news group about T_EX and L^AT_EX, in english.

Computer Modern A set of fonts [120] designed by Donald KNUTH for T_EX. Initially they were built with METAFONT [122] (a program also created by KNUTH), but PostScript versions exist today, with extensions (for accented characters, mainly): the EC-fonts (European Computer Modern), the cm-super fonts, etc.

counter A T_EX register containing an integer value. There are 256 counters (from 0 to 255) in T_EX, but L^AT_EX uses some of them, and many packages need some counters for their own usage. An extended version derived from T_EX, ε -T_EX [78], allows more counters. Omega (Ω) also offers more counters.

cri.tex An example file showing the use of the `minitoc` package with starred parts and chapters.

³ This note is extracted from [167], then adapted.

CTAN The *Comprehensive T_EX Archive Network*, a set of computer archives containing most of the T_EX related resources (like fonts, software, documentations, packages). They are accessible via Internet. The participating hosts in the Comprehensive TeX Archive Network are:

- `ftp.dante.de` (Hamburg, Germany)
 - anonymous ftp `/tex-archive (/pub/tex /pub/archive)`
 - rsync access on <rsync://rsync.dante.de/CTAN/>
 - World Wide Web access on <http://www.dante.de/>
 - Administrator: <mailto:ftpmaint@dante.de>
- `ftp.tex.ac.uk` (Cambridge, UK)
 - anonymous ftp `/tex-archive (/pub/tex /pub/archive)`
 - rsync access on <rsync://rsync.tex.ac.uk/CTAN/>
 - NFS mountable from <nfs.tex.ac.uk/tex-archive>
 - World Wide Web access on <http://www.tex.ac.uk/>
 - Administrator: <mailto:ctan-uk@tex.ac.uk>
- `tug.ctan.org` (Colchester, Vermont, USA)
 - anonymous ftp `/tex-archive (/pub/archive)`
 - World Wide Web access on <http://www.ctan.org/>
 - Administrator: <mailto:tex@ctan.tug.org>

There are also many sites mirroring these main archives.

depth In the standard documents classes (and in most classes) with sectionning commands, we have a notion of **depth**. The depth of a sectionning command determines the numbering level in its title (from the value of the `secnumdepth` counter), and the entries for a given sectionning command appear in the main table of contents if the depth of this sectionning command is lower than or equal to the value of the `tocdepth` counter; see table 6.4 for the depths of the sectionning commands in the main document classes.

Table 6.4: Depths for sectionning commands

Class:	book	report	article
<code>secnumdepth</code>	2	2	3
<code>\part</code>	-1	-1	0
<code>\chapter</code>	0	0	×
<code>\section</code>	1	1	1
<code>\subsection</code>	2	2	2
<code>\subsubsection</code>	3	3	3
<code>\paragraph</code>	4	4	4
<code>\subparagraph</code>	5	5	5

The mechanism is analog for the `parttocs`, `minitocs`, and `secttocs`, using the values of 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 \LaTeX), there are 16 file descriptors for writing and 16 file descriptors for reading.

duplex2v.pro A PostScript header file, which tries to activate recto-verso printing. It works with some HP and Canon PostScript printers.

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 `imk` must have been run previously. See item 9 on page 216.

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), etc. See [142].

environment An environment is a delimited domain in a document, where special rules apply. Such a domain is delimited by `\begin{env} ... \end{env}` and may take arguments, like this:

```
\begin{minipage}[t]{.5\textwidth}
\end{minipage}
```

ethiop A \LaTeX package [23] giving the ethiopian language support for the Babel package [30, 31].

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 51 and 2.5 on page 56. It is strongly recommended to not have more than one dot in a file name.

farsi.sty See \FarsiTeX below.

\FarsiTeX A package to typeset a document in the farsi (iranian, persian) language. See <http://www.farsitex.org>. But this package is today available only for \LaTeX 2.09. See also sections 13.51 on page 440 and 13.52 on page 440.

features A feature (for the `minitoc` package) is a set of actions executed at each occurrence of a mini-table of a given type. Three features are associated to each mini-table type: a “before” feature (executed before the mini-table), an “after” feature (executed after the mini-table), and a “thispagestyle” feature, which is executed with the mini-table to set its page style. Look at the documentation of the `\mtcsetfeature` command, in section 1.10 on page 45.

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 1.4.12 on page 41) and some files used in the compilation of the documentation.

\firstchapteris An obsolete command, temporary used as a workaround for the Chapter 0 problem; see **Chapter 0** and **absolute numbering** above, and section 1.5.4 on page 46.

\firstpartis Analog to `\firstchapteris` above.

\firstsectionis Analog to `\firstchapteris` above

\FloatBarrier A macro from the `placeins` package [9]. It set up a “barrier” against the drift of floats (like figures or tables).

fminitoc.bib A bibliographic data base for the french documentation of the `minitoc` package.

fminitoc.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 `minitoc.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 `fminitoc.lan` (i.e., by `\jobname.lan`). `fminitoc.dtx` is generated when compiling `minitoc.ins`.

fminitoc.ist This file contains a style for formatting the index in the french documentation. It is generated when compiling `minitoc.ins`.

fminitoc.lan A file used to force the french language in the documentation. It is generated when compiling `minitoc.ins`.

fminitoc.pdf The french documentation in PDF format.

fminitoc.ps The french documentation in PostScript format. No more distributed (but look at the `fmk` and `pmk` scripts).

fmk An example of shell script to prepare the french documentation of the `minitoc` package. The script `imk` must have be run previously. See item 9 on page 216.

fncychap The `fncychap` package [146] provides a set of commands for changing the format used for some headings (chapters) in the standard L^AT_EX 2_ε document classes: `book`, and `report`. It must be loaded *before* the `minitoc` package (see point 1.8 on page 50 and section 2.38 on page 72).

fo1.tex An example file showing the use of the `minitoc` package with changing some fonts.

fo2.tex Another example file showing the use of the `minitoc` package with changing some fonts.

- 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`.
- frnew.sty** A small package file used to prepare the french documentation. It is generated when compiling `minitoc.ins`.
- frplain1.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), then modified with the `urlbst` [103] tool to add an URL field.
- gaps.tex** An example file showing the use of the `\mtcgapbeforeheads` and `\mtcgapafterheads` commands.
- guarani** A \LaTeX package to compose text in Guarani, the main language spoken in Paraguay. The file `guarani.ldf`, included in this package, defines the titles. See [26] and section 13.71 on page 451.
- hangcaption** The `hangcaption` package [119] defines a variant of the `\caption` command to produce captions with hanging indentation. This package is likely obsolete (1992, \LaTeX 2.09).
- 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.
- hide1.tex** An example file showing the use of the `mtchideinmainlof` and `mtchideinmainlot` environments.
- hide2.tex** An example file showing the use of the following pairs of commands: `\mtchideinmainlof` and `\endmtchideinmainlof`, `\mtchideinmainlot` and `\endmtchideinmainlot`.
- 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 coherence between main `minitoc` commands, etc., and gives warnings and other useful hints (mainly in the `.log` file). This is a default option (use the `nohints` option to skip these checks).
- 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.
- H \LaTeX** A system to write documents in the Korean language, using Lambda (Λ) (see below). Written by KOAUNGH Un [124, in korean]. It uses special Hangûl or Hanja fonts and the UTF-8 input encoding.

- hop.tex** An example file showing the use of the `minitoc` package with the `scrbook` document class.
- 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 [179] is used to emend cross-referencing commands in \LaTeX to produce some sort of `\special` commands; there are backends for the `\special` set defined for `HyperTeX` dvi processors, for embedded pdfmark commands for processing by Acrobat Distiller (`dvips` and `dvipson`), for `dviwindo`, for `pdfTeX`, for `TeX4ht`, and for `VTeX`'s pdf and HTML backends.
- This package derives from, and builds on, the work of the `HyperTeX` project, described at <http://xxx.lanl.gov/hypertex/>. It extends the functionality of all the \LaTeX 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 [216] provides an if-then-else command for testing if a macro argument is empty (“empty” meaning zero or more spaces only).
- 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 9 on page 216.
- insection** The `insection` package option loads the `placeins` package [9] 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 28.
- 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:
- ```
\parttoc, \partlof, \partlot,
\minitoc, \minilof, \minilot,
\secttoc, \sectlof, \sectlot,
\mtcprepare
```
- INSTALL** A text file describing the installation of the `minitoc` package. See chapter 7 on page 214.
- \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.
- jura** The `jura` class [77], by Felix BRAUN, is incompatible with the `minitoc` package. It is used for german judicial documents.

**k-loose** A minitoc package option useful if your document is written with one of the KOMA-Script classes. This option tries to set a loose line spacing in the mini-tables. Analog to the loose package option for standard classes.

**k-tight** A minitoc package option useful if your document is written with one of the KOMA-Script classes. This option tries to set a tight line spacing in the mini-tables. Analog to the tight package option for standard classes.

**KOMA-Script** KOMA-Script [164] 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 (`scrartcl.cls`, `scrbook.cls`, `scrletter.cls`, `scrlettr2.cls`, and `scrreprt.cls`) 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 `scrbook.cls`, `scrreprt.cls`, and `scrartcl.cls` classes are compatible with the minitoc package, with some precautions (see section 1.5.5 on page 47). The `scrletter.cls` and `scrlettr2.cls` have no sectioning commands, so the 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 L<sup>A</sup>T<sub>E</sub>X kernel.

The main classes of the KOMA-Script bundle are designed as counterparts to the standard L<sup>A</sup>T<sub>E</sub>X classes. This means that the KOMA-Script bundle contains replacements for the three standard classes `book`, `report`, and `article`. There is also a replacement for the standard class `letter`.

**Lambda** The L<sup>A</sup>T<sub>E</sub>X format (in the T<sub>E</sub>X meaning of that word) adapted to the special features of Omega ( $\Omega$ ) is called “Lambda” ( $\Lambda$ ).

**L<sup>A</sup>T<sub>E</sub>X** L<sup>A</sup>T<sub>E</sub>X [132] 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. L<sup>A</sup>T<sub>E</sub>X uses T<sub>E</sub>X [121, 123] as its formatting engine (from [167]).

In fact, L<sup>A</sup>T<sub>E</sub>X is a macro package that enables authors to typeset and print their work at the highest typographical quality, using a predefined, professional layout. L<sup>A</sup>T<sub>E</sub>X was originally written by Leslie LAMPORT [132]. It uses the T<sub>E</sub>X formatter as its typesetting engine. These days L<sup>A</sup>T<sub>E</sub>X is maintained by Frank MITTELBACH and his team.

In 1994 the L<sup>A</sup>T<sub>E</sub>X package was updated by the L<sup>A</sup>T<sub>E</sub>X3 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 L<sup>A</sup>T<sub>E</sub>X2.09 some years earlier. To distinguish the new version from the old, it is called L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>.

L<sup>A</sup>T<sub>E</sub>X is pronounced “Lay-tech” or “Lah-tech.” If you refer to L<sup>A</sup>T<sub>E</sub>X in an ASCII environment, you type LaTeX. L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> is pronounced “Lay-tech two e” and typed LaTeX2e.

- L<sup>A</sup>T<sub>E</sub>X2.09** An obsolete version of the L<sup>A</sup>T<sub>E</sub>X program, before 1994; it is no more supported. Do not use it. Use the current version of L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>, which is supported and much more efficient.
- L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>** The current version of the L<sup>A</sup>T<sub>E</sub>X program, after 1994; it is supported.
- 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 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 `lipsum` package [106] 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 <http://lipsum.com> for the origin of this text (pieces of *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 L<sup>A</sup>T<sub>E</sub>X run) in the `document.maf` file. See section 1.7 on page 49.
- livre.tex** An example file showing the use of the `minitoc` package in a book with customized table of contents and minitocs.
- LOF** An acronym for “list of figures”.
- lofdepth** This counter, if it exists, contains the depth of the list of figures.
- 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** 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.
- \makeatletter** and **\makeatother** Many<sup>4</sup> internal commands of L<sup>A</sup>T<sub>E</sub>X, 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 commands should never be used in a package or class file.

---

<sup>4</sup> Informations from [159, page 843].

- 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 [113] provides a `margin` environment for putting captions in the outer document margin with either a top or bottom alignment.
- mem.tex** An example file showing the use of the `minitoc` package with the `memoir` class.
- mem1.tex** An example file showing the use of the `minitoc` package with the `memoir` class, if you want to change some fonts.
- memoir, memoir.cls** A very general and powerful document class (by Peter R. WILSON, described in [213, 214]); this class is compatible with the `minitoc` package (with some precautions) if you use a recent version. See section 2.22 on page 62.
- mini-art.tex** An example of document (`article` class) using the `minitoc` package.
- 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 table 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 57.
- 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 minilots.
- minitoc** A table of contents for a chapter. Also used as a generic term for any mini-table (see “mini-table” above).
- minitoc-ex.tex** An example of document (`book` or `report` class) using the `minitoc` package.



**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 53.

**minitocdepth** This counter contains the depth of the `minitocs`.

**minitoc.dtx** The file containing the documentation and the commented code of the `minitoc` package.

**minitoc-hyper.sty** A special version of the `minitoc` package which has been prepared by Bernd JAEHNE and Didier VERNA to work with the powerful `hyperref` package [179]; Heiko OBERDIEK has integrated their work so since version #31, `minitoc` is compatible with `hyperref`. *Hence the `minitoc-hyper.sty` package is now obsolete and should no more be used. It is still present on the CTAN archives for compatibility with old documents.*



**minitoc.ins** The installation file for the `minitoc` package. Compiling it with  $\text{\LaTeX}$  produces most of the files of the `minitoc` package.

**minitoc.ist** This file contains a style for forming 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  $\text{\LaTeX}$  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 `emk` and `pmk` 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 75.

**minutes** The `minutes` package [145] (by Knut LICKERT) 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!

**MonTeX** `MonTeX` [87, 89] 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 (Russienc). Bicig is another name for Uighur. You can find many things about

Mongolia and Mongolian at the web site [88]. The following description is extracted from [89].

MonT<sub>E</sub>X 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 *Buryaad* 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, \xalx{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 name 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 Mongolian 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). Manju writing is derived from Uighur Mongolian by adding diacritics in the form of 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.

**mtcmess, mtcmess.sty** 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, mtcoff.sty** 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, mtcpatchmem.sty** 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 1mu is font dependent. The separation between dots in the dotted lines in the mini-tables is expressed in math units.

**multibib** The `multibib` package [105] allows to create references to multiple bibliographies within one document. It thus provides a complementary functionality to packages like `bibunits` [104] or `chapterbib` [7], which allow to create one bibliography for multiple, but different parts of the document.

**mu.tex** A document using a `minitoc` set in a `wrapfigure` environment with the `wrapfig` package [6].

**needspace** The `needspace` package [206] 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  $\text{\LaTeX} 2_{\epsilon}$  font selection system [141] was first released as the “New Font Selection Scheme” (NFSS) in 1989, and then in release 2 in 1993.  $\text{\LaTeX} 2_{\epsilon}$  includes NFSS release 2 as standard.

Table 6.5: Most common font encodings

| Encoding      | Description                                                       |
|---------------|-------------------------------------------------------------------|
| T1            | L <sup>A</sup> T <sub>E</sub> X extended text (“Cork”)            |
| TS1           | L <sup>A</sup> T <sub>E</sub> X symbols (Latin)                   |
| T2A, T2B, T2C | L <sup>A</sup> T <sub>E</sub> X text (Cyrillic)                   |
| T3            | L <sup>A</sup> T <sub>E</sub> X phonetic alphabet                 |
| TS3           | L <sup>A</sup> T <sub>E</sub> X phonetic alphabet (extra symbols) |
| T4            | L <sup>A</sup> T <sub>E</sub> X text (African languages)          |
| T5            | L <sup>A</sup> T <sub>E</sub> X text (Vietnamien)                 |
| T7            | L <sup>A</sup> T <sub>E</sub> X text (reserved for Greek)         |
| OT1           | T <sub>E</sub> X text (as defined by Donald K <sub>N</sub> UTH)   |
| OT2           | T <sub>E</sub> X text for Cyrillic languages (obsolete)           |
| OT3           | International phonetic alphabet (obsolete)                        |
| OT4           | T <sub>E</sub> X text with extensions for the Polish language     |
| OT6           | T <sub>E</sub> X text with extensions for the Armenian language   |
| OML           | T <sub>E</sub> X math italic (Donald K <sub>N</sub> UTH)          |
| OMS           | T <sub>E</sub> X math symbols (Donald K <sub>N</sub> UTH)         |
| OMX           | T <sub>E</sub> X math large symbols (Donald K <sub>N</sub> UTH)   |
| X2            | L <sup>A</sup> T <sub>E</sub> X extended text (Cyrillic)          |
| U             | Unknown                                                           |
| L<xx>         | A local encoding                                                  |
| LTH           | Encoding used for the Thai language                               |
| LV1           | Encoding used with some V <sub>T</sub> eX fonts                   |
| LY1           | Alternative to T1 encoding, for Y&Y software                      |

Every text font in L<sup>A</sup>T<sub>E</sub>X has five *attributes*:

**encoding** This specifies the order that characters appear in the font. The two most common text encodings used in L<sup>A</sup>T<sub>E</sub>X are K<sub>N</sub>UTH’s “T<sub>E</sub>X text” encoding, and the “T<sub>E</sub>X text extended” encoding developed by the T<sub>E</sub>X Users Group members during a T<sub>E</sub>X Conference at Cork in 1990 (hence its informal name “Cork encoding”). See [142].

**family** The name for a collection of fonts, usually grouped under a common name by the font foundry. For example, “Adobe Times”, “ITC Garamond”, and K<sub>N</sub>UTH’s “Computer Modern Roman” are all font families.

**series** How heavy or expanded a font is. For example, “medium weight”, “narrow” and “bold extended” are all series.

**shape** The form of the letters within a font family. For example, “italic”, “oblique” and “upright” (sometimes called “roman”) are all font shapes.

**size** The design size of the font, for example “10pt”.

The possible values for these attributes are given short acronyms by L<sup>A</sup>T<sub>E</sub>X. The most common values for the font encoding are given in table 6.5.

The “local” encodings are intended for font encodings which are only locally available, for example a font containing an organisation’s logo in various sizes.

Table 6.6: Most common font families

| Family            | Description                     |
|-------------------|---------------------------------|
| <code>cmr</code>  | Computer Modern Roman           |
| <code>cmss</code> | Computer Modern Sans            |
| <code>cmtt</code> | Computer Modern Typewriter      |
| <code>cmm</code>  | Computer Modern Math Italic     |
| <code>cmsy</code> | Computer Modern Math Symbols    |
| <code>cmex</code> | Computer Modern Math Extensions |
| <code>ptm</code>  | Adobe Times                     |
| <code>phv</code>  | Adobe Helvetica                 |
| <code>pcr</code>  | Adobe Courier                   |

Table 6.7: Most common font series

| Series          | Description   |
|-----------------|---------------|
| <code>m</code>  | Medium        |
| <code>b</code>  | Bold          |
| <code>bx</code> | Bold extended |
| <code>sb</code> | Semi-bold     |
| <code>c</code>  | Condensed     |

Table 6.8: Most common font shapes

| Shape           | Description                           |
|-----------------|---------------------------------------|
| <code>n</code>  | Normal (that is “upright” or “roman”) |
| <code>it</code> | Italic                                |
| <code>sl</code> | Slanted (or “oblique”)                |
| <code>sc</code> | Caps and small caps                   |

Table 6.9: The five font parameters for some fonts

|     | <b>L<sup>A</sup>T<sub>E</sub>X</b> specification |                |                 |                   |  | <b>Font</b>                          | <b>T<sub>E</sub>X</b> font name |
|-----|--------------------------------------------------|----------------|-----------------|-------------------|--|--------------------------------------|---------------------------------|
| OT1 | <code>cmr</code>                                 | <code>m</code> | <code>n</code>  | <code>10</code>   |  | Computer Modern Roman 10 point       | <code>cmr10</code>              |
| OT1 | <code>cmss</code>                                | <code>m</code> | <code>sl</code> | <code>1pc</code>  |  | Computer Modern Sans Oblique 1 pica  | <code>cmssi12</code>            |
| OML | <code>cmm</code>                                 | <code>m</code> | <code>it</code> | <code>10pt</code> |  | Computer Modern Math Italic 10 point | <code>cmmi10</code>             |
| T1  | <code>ptm</code>                                 | <code>b</code> | <code>it</code> | <code>1in</code>  |  | Adobe Times Bold Italic 1 inch       | <code>ptmb8t at 1in</code>      |

There are far too many font families to list them all, but some common ones are listed in table 6.6. The most common values for the font series are listed in table 6.7. The most common values for the font shape are listed in table 6.8.

The font size is specified as a dimension, for example `10pt` or `1.5in` or `3mm`; if no unit is specified, `pt` is assumed. These five parameters specify every L<sup>A</sup>T<sub>E</sub>X font, see table 6.9, for example.

These five parameters are displayed whenever L<sup>A</sup>T<sub>E</sub>X gives an overfull box warning, for example:

```
Overfull \hbox (3.80855pt too wide) in paragraph at
lines 314--318
[]\OT1/cmr/m/n/10 Normally [] and [] will be iden-ti-cal,
```

Table 6.10: Author commands for fonts

| Author command                                    | Attribute | Value in article class |
|---------------------------------------------------|-----------|------------------------|
| <code>\textrm...</code> or <code>\rmfamily</code> | family    | cmr                    |
| <code>\textsf...</code> or <code>\sffamily</code> | family    | cmss                   |
| <code>\texttt...</code> or <code>\ttfamily</code> | family    | cmtt                   |
| <code>\textmd...</code> or <code>\mdseries</code> | series    | m                      |
| <code>\textbf...</code> or <code>\bfseries</code> | series    | bx                     |
| <code>\textup...</code> or <code>\upshape</code>  | shape     | n                      |
| <code>\textit...</code> or <code>\itshape</code>  | shape     | it                     |
| <code>\textsl...</code> or <code>\slshape</code>  | shape     | sl                     |
| <code>\textsc...</code> or <code>\scshape</code>  | shape     | sc                     |
| <code>\tiny</code>                                | size      | 5pt                    |
| <code>\scriptsize</code>                          | size      | 7pt                    |
| <code>\footnotesize</code>                        | size      | 8pt                    |
| <code>\small</code>                               | size      | 9pt                    |
| <code>\normalsize</code>                          | size      | 10pt                   |
| <code>\large</code>                               | size      | 12pt                   |
| <code>\Large</code>                               | size      | 14.4pt                 |
| <code>\LARGE</code>                               | size      | 17.28pt                |
| <code>\huge</code>                                | size      | 20.74pt                |
| <code>\Huge</code>                                | size      | 24.88pt                |

The table 6.10 lists the author commands for fonts which set the five attributes<sup>5</sup>.

**nocheckfiles** A package option of minitoc. The opposite of the checkfiles package option (see above).

**nohints** A package option of minitoc. The opposite of the hints package option (see above).

**nolistfiles** An option of the minitoc package. It is the opposite of the listfiles above. Default. See section 1.7 on page 49.

**notoccite** This option of the minitoc package loads the notoccite package [8] (by Donald ARSENEAU). It avoids problems with `\cite` commands in sectioning commands or captions. See section 1.6 on page 49.

**Omega** The Omega typesetting system<sup>6</sup> ( $\Omega$ ) (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 by Donald 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

<sup>5</sup> The values used by these commands are determined by the document class.

<sup>6</sup> Most but not all of this note is taken in the Omega documentation.

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  $\text{\TeX}$ . Omega also allows multiple input and output character sets, and uses programmable filters 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  $\text{\TeX}$  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  $\text{\LaTeX}$  format (in the  $\text{\TeX}$  meaning of that word) adapted to the special features of Omega is called “Lambda” ( $\Lambda$ ). Extending Omega with the  $\varepsilon$ - $\text{\TeX}$  extensions is a separate project, known as Aleph ( $\aleph$ ) and led by Giuseppe BILOTTA. The  $\text{\LaTeX}$  for Aleph is known as *Lamed* ( $\aleph$ ).

**package** Packages<sup>7</sup> are a very important feature of  $\text{\LaTeX}$ . These are extensions to the basic  $\text{\LaTeX}$  commands that are written to files with names that end with `.sty` and are loaded with the command `\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  $\text{\LaTeX}$  basic installation and are therefore fully standard.
- **Tools** packages are a set written by members of the  $\text{\LaTeX}$ 3 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.
- **$\mathcal{AMS}$ - $\text{\LaTeX}$**  packages, published by the American Mathematical Society, should be in any installation<sup>8</sup>.
- **Contributed** packages have been submitted by actual users; certain of these have established themselves as “essential” to standard  $\text{\LaTeX}$  usage, but all are useful.

**partlof** A list of figures for a part.

**partlofdepth** This counter, if defined, contains the depth of the partlofs.

**partlot** A list of tables for a part.

**partlotdepth** This counter, if defined, contains the depth of the partlots.

**parttoc** A table of contents for a part.

**parttocdepth** This counter contains the depth of the parttocs.

<sup>7</sup> This info is taken from [125, page 12–13] and adapted.

<sup>8</sup> They are indispensable if you use a lot of mathematics.

- PDF** Portable Document Format [100]. A descendant of the Postscript language from Adobe, optimized for navigation on the Internet. It adds hypertext, font substitution, and compression features.
- placeins** The `placeins` package [9] keeps floats “in their place”, preventing them from floating past a `\FloatBarrier` command into another section. To use it, declare `\usepackage{placeins}` in the preamble and insert `\FloatBarrier` commands at places that floats should not move past, perhaps at every `\section`. The `insection` package option of the `minitoc` package does that with adequate options, and loads also the `flafter` package (described in [138] and [159, page 286]); see section 1.3.3 on page 28.
- placeins.txt** A plain text file containing the documentation of the `placeins` package [9].
- plainurl.bst** A bibliographic style derived from the `plain.bst` standard style, but modified with the `urlbst` tool [103] to add an URL field.
- PLATEX** A version of  $\text{\LaTeX}$  customized for the polish (polski) language. See [168, 204]. But the same name refers also to a version of  $\text{\LaTeX}$  customized for the japanese language,  $\text{\LaTeX} 2_{\epsilon}$  [115].
- pmk** An example of shell script to prepare the `minitoc` package and its documentation; you should adapt it to your needs. See item 9 on page 216.
- 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  $\text{\LaTeX}$  source document, the part of it between the commands `\documentclass[...]{...}` and `\begin{document}`. In the preamble, you can insert global declarations and the loading of packages via `\usepackage` commands.
- preparation** The preparation commands of the `minitoc` package prepare the auxiliary files for the mini-tables of a given type. A *preparation* command must have been invoked (only once) before any insertion command for the mini-table type. The preparation commands are:
- ```
\doparttoc, \dopartlof, \dopartlot,
\dominitoc, \dominilof, \dominilot,
\dosecttoc, \dosectlof, \dosectlot,
\mtcprepare
```
- proc** A standard \LaTeX document class, for preparing conference proceedings. For the `minitoc` package, it is very similar to the `article` document class; see above.
- pseudo-chapter** Or starred chapter. A chapter introduced by a `\chapter*` command. By default, it has no entry in the table of contents. `\chapter*` needs some precautions with the `minitoc` package. See section 1.3.4 on page 32.

- quotchap** The quotchap package [193] provides a set of commands for adding quotations to some headings (chapters) in the standard L^AT_EX 2_ε document classes: book, and report. It must be loaded *before* the minitoc package (see point 1.8 on page 50 and section 2.39 on page 72).
- README** is a plain text file (english) describing briefly the minitoc package, plus some useful infos.
- report** A standard L^AT_EX document class. It has sectionning 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).
- rmk** An example of shell script, which sorts the files of the minitoc package into classes (one directory for each class). It should be run after the scripts imk (mandatory) and emk and/or fmk, in that sequence. See item 9 on page 216.
- romannum** The romannum package [215] changes the numbers (for sectionning commands) generated by L^AT_EX from arabic digits to roman numerals. This package uses the stdclsdv package [205]. It must be loaded *before* the minitoc package (see point 1.8 on page 50 and section 2.40 on page 72).
- rubber** Rubber [16] is a wrapper for L^AT_EX and companion programs. Its purpose is, given a L^AT_EX source to process, to compile it enough times to resolve all references, possibly running satellite programs such as B^BT_EX, makeindex, Metapost, etc., to produce appropriate data files. It has facilities to make some post-processing cleanup actions, like deleting the auxiliary files created by minitoc.
- scr.tex** An example file showing the use of the minitoc package with a KOMA-Script class, scrreprt.
- scrartcl** See KOMA-Script above.
- scrbook** See KOMA-Script above.
- scrreprt** See KOMA-Script above.
- second.tex** An example of document using the minitoc package and its french2 language option.
- sectionning commands** These are the L^AT_EX commands which specify the logical structure of your document. The main sectionning commands are \part, \chapter, \section, \subsection, \subsubsection, \paragraph, and \subparagraph. Some standard document classes have not the \chapter command (like the article and proc classes), some have no sectionning commands (like the letter class). In the later case, the minitoc package is pointless. If some of the \part, \chapter, or \section commands are not defined, the minitoc commands for that level are unavailable. If \chapter is defined, the minitoc commands at the section level are not defined in the current and older versions of the minitoc package, but if \chapter is not defined and \section is defined, then the minitoc commands at the section level are defined. See section 1.1.1 on page 25. In non-standard document classes, sectionning commands with non-standard names cannot be recognized by the 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 [153] provides a set of commands for changing the font used for the various sectional headings in the standard L^AT_EX 2_ε document classes: `article`, `book`, and `report`. This package also works with the KOMA-Script classes `scrartcl`, `scrbook`, and `scrreprt`. It must be loaded *before* the `minitoc` package (see point 1.8 on page 50 and section 2.28 on page 67).
- secttoc** A table of contents for a section.
- secttocdepth** This counter contains the depth of the secttocs.
- sfheaders** The `sfheaders` package [148] (for L^AT_EX 2_ε) 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 51 and section 2.41 on page 72).
- 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.).
- shorttext** 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 56.
- shorttoc** The `shorttoc` package [92] allows to create an other table of contents in a document, with an other depth and an other title 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 [143, 144].
- SI^AT_EX A version of L^AT_EX customized for the swedish language. See [152].
- stdclsdv** The `stdclsdv` package [205] is intended to be used by the authors of L^AT_EX 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.
- subf.tex** An example file showing the use of the `minitoc` package with the `subfigure` package [84].
- subfig** The `subfig` package [86] provides support for the inclusion of small, “sub-figures” 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 [84] is an obsolete version of the `subfig` package [86].
- suffix** See “extension” above.
- tbi.tex** An example file showing the use of the `minitoc` package with the `tocbibind` package [210].
- T_EX** T_EX is a computer program created by Donald KNUTH [121]. It is aimed at typesetting text and mathematical formulae. KNUTH started writing the T_EX 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. T_EX 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. T_EX is renowned for being extremely stable, for running on many different kinds of computers, and for being virtually bug free. The version number of T_EX is converging to π and is now at 3.141592.
- T_EX is pronounced “Tech,” with a “ch” as in the German word “Ach” or in the Scottish “Loch.” In an ASCII environment, T_EX becomes TeX.
- thailatex** The `thailatex` package [188] allows to typeset documents in the Thai language. You can also use the CJK system [143, 144].
- thispagestyle** A type of *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 `\mtcsetfeature` command, in section 1.10 on page 45.
- tight** An option of the `minitoc` package. It gives a tight line spacing in the mini-tables. The opposite option is `loose`.
- titlesec** The `titlesec` package [27] 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 *incompatible* with the `minitoc` package.
- titletoc** The `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 *incompatible* with the `minitoc` package.

The `titletoc.sty` file is not part of the `titlesec` package; it's an independent package, but it's described in the `titlesec` package documentation [27].

tlc.tex An example file showing the use of the `minitoc` package in a document of `article` class. It is the example of [159, page 58], modernized.

TOC Acronym for “table of contents”.

tocbibind The `tocbibind` package [210] 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 `minitoc` package. See section 1.5.5 on page 47.

tocdepth This counter contains the depth of the table of contents.

tocloft The `tocloft` package [207] 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 `tocloft` package and the `minitoc` package, see section 2.21 on page 61 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 token⁹ is either (a) a single character with an attached category code (see “catcode” 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 example, 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.

tsfc.tex An example file showing the use of the `minitoc` package with the `subfig` package [86].

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).

⁹ This definition is taken from “*The \TeX book*” [121].

- UNIX-like** Operating systems analog to Unix, with the same advantages. Linux is a good example, but others exist.
- UTF 8** UTF 8 (Unicode Transformation Format 8), also called UTF 2 or FSS-UTF, is a special representation of Unicode (resp. ISO 10646). 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 191.
- varsects** The varsects package [189] provides a set of commands for changing the font used for the various sectional headings in the standard \LaTeX 2 ϵ document classes: `article`, `book`, and `report`. It must be loaded *before* the `minitoc` package (see point 1.8 on page 50 and section 2.33 on page 69).
- wrapfig** The wrapfig package [6] 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 [79] implements a system for eXternal References. I wrote the first version of this package, but it had severe problems. David CARLISLE rewrote it in a much better and more robust way. With his permission, I used some of its code in the `minitoc` package to implement the preparation commands (like `\dominitoc`). If you use also the `hyperref` package [179], use `xr-hyper` [80] in place of the `xr` package.

Chapter 7

Installation

Tables

7.1 List of files (<code>minitoc.1</code>)	215
--	-----

This chapter describes the installation of the `minitoc` package (version #47).

This package contains a lot of files. The list of all files is given in `minitoc.1`. See table 7.1 on the next page. 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 `fminitoc.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`, `mtcoeff.sty`, `mtcmess.sty`, and *all* `*.mld` and `*.mlo` files are the package itself¹.

The table 1.7 on page 36 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 $\text{\LaTeX} 2_{\epsilon}$ finds the `.sty` files.

- (2) Informative text files:

- `INSTALL` is a file describing the installation of the package. You are (almost) reading it (but it is shorter).

¹ The large number `*.mld` files is (partialy) a consequence of the fact that some languages have aliases (or dialects) and hence one `*.mld` file for each name (a `*.mld` file can load another one); 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, russian, serbian), or even for spelling reforms (german, greek, norsk).

Table 7.1: List of files (minitoc.1)

class (0) : -minitoc.ins -minitoc.dtx -fminitoc.dtx class (1) : -minitoc.sty -mtcoeff.sty -mtcmess.sty -mtcpatchmem.sty -acadian.mld -acadien.mld -afrikaan.mld -afrikaans.mld -albanian.mld -american.mld -arab.mld -arab2.mld -arabi.mld -arabic.mld -armenian.mld -australian.mld -austrian.mld -bahasa.mld -bahasai.mld -bahasam.mld -bangla.mld -basque.mld -bicig.mld -bicig2.mld -bicig3.mld -bithe.mld -brazil.mld -brazilian.mld -breton.mld -british.mld -bulgarian.mld -bulgarianb.mld -buryat.mld -buryat2.mld -canadian.mld -canadien.mld -castillan.mld -castillian.mld -catalan.mld -chinese1.mld, and chinese1.mlo -chinese2.mld, and chinese2.mlo -croatian.mld -czech.mld -danish.mld -dutch.mld -english.mld -english1.mld -english2.mld -esperant.mld -esperanto.mld -estonian.mld -ethiopia.mld -ethiopian.mld -ethiopian2.mld -farsi1.mld, and farsi1.mlo -farsi2.mld, and farsi2.mlo -farsi3.mld -finnish.mld	-finnish2.mld -francais.mld -french.mld -french1.mld -french2.mld -frenchb.mld -frenchle.mld -frenchpro.mld -galician.mld -german.mld -germanb.mld -germanb2.mld -greek.mld -greek-mono.mld -greek-polydemo.mld -greek-polykatha.mld -guarani.mld -hangul1.mld, and hangul1.mlo -hangul2.mld, and hangul2.mlo -hangul3.mld, and hangul3.mlo -hangul4.mld, and hangul4.mlo -hangul-u8.mld, and hangul-u8.mlo -hanja1.mld, and hanja1.mlo -hanja2.mld, and hanja2.mlo -hanja-u8.mld, and hanja-u8.mlo -hebrew.mld -hebrew2.mld -hungarian.mld -icelandic.mld -indon.mld -indonesian.mld -interlingua.mld -irish.mld -italian.mld -italian2.mld -japanese.mld, and japanese.mlo -japanese2.mld, and japanese2.mlo -japanese3.mld, and japanese3.mlo -japanese4.mld, and japanese4.mlo -japanese5.mld, and japanese5.mlo -kannada.mld -khalkha.mld -latin.mld -latin2.mld -latvian.mld -letton.mld -lithuanian.mld -lowersorbian.mld -lsorbian.mld -magyar.mld -magyar2.mld -magyar3.mld -malay.mld -malayalam-keli.mld	-malayalam-omega.mld, and malayalam-omega.mlo -malayalam-rachana.mld -malayalam-rachana2.mld -manju.mld -meyalu.mld -mongol.mld -naustrian.mld -newzealand.mld -ngerman.mld -ngermanb.mld -ngermanb2.mld -norsk.mld -norsk2.mld -nynorsk.mld -nynorsk2.mld -polish.mld -polish2.mld -polski.mld -portuges.mld -portuguese.mld -romanian.mld -romanian2.mld -romanian3.mld -russian.mld -russianb.mld -russianc.mld -russian2m.mld -russian2o.mld -russian-cca.mld, and russian-cca.mlo -russian-cca1.mld, and russian-cca1.mlo -russian-lh.mld, and russian-lh.mlo -russian-lhcyralt.mld, and russian-lhcyralt.mlo -russian-lhcyrkoi.mld, and russian-lhcyrkoi.mlo -russian-lhcyrwin.mld, and russian-lhcyrwin.mlo -samin.mld -scottish.mld -serbian.mld -serbianc.mld -slovak.mld -slovene.mld -spanish.mld -spanish2.mld -spanish3.mld -spanish4.mld -swedish.mld -swedish2.mld -thai.mld, and thai.mlo -turkish.mld -uighur.mld -uighur2.mld -uighur3.mld -UKenglish.mld -ukraineb.mld -ukrainian.mld -uppersorbian.mld -USenglish.mld -usorbian.mld -vietnam.mld -vietnamese.mld -welsh.mld	-xalx.mld -xalx2.mld -xalx3.mld class (2) : -INSTALL -minitoc.l -README -catalog -TODO class (3) : -minitoc-ex.tex, mini-art.tex, second.tex, add.tex, addsec.tex, add.bib, hide1.tex, hide2.tex, apx.tex, app-mem.tex, mem.tex, mem1.tex, fol.tex, fo2.tex, scr.tex, subf.tex, tsfc.tex, tbi.tex, 2c.tex, bo.tex, hop.tex, cri.tex, livre.tex, ch0.tex, tlc.tex, mu.tex, hir.tex, hia.tex, gaps.tex class (4) : -minitoc.bug -minitoc.sum class (5) : -minitoc.ins -minitoc.dtx -minitoc.bib -minitoc.ist -minitoc.lan -minitoc.pre -plainurl.bst -lamed.eps, lamed.pdf, lamed.tex class (6) : -minitoc.dtx -fminitoc.dtx -fminitoc.bib -fminitoc.ist -fminitoc.lan -minitoc.pre -franc.sty -frbib.sty -frnew.sty -frplain1.bst class (7) : -minitoc.pdf class (8) : -fminitoc.pdf class (9) : -pmk -imk -emk -fmk -rmk -xmk -duplex2v.pro
--	---	---	--

- `minitoc.l` contains the list of all files of the `minitoc` distribution. See table 7.1 on the page before.
 - `README` is a file describing briefly the `minitoc` package, plus some useful infos.
 - `catalog` contains basic infos about the `minitoc` package (abstract, date, author, version, licence).
 - `TODO` lists some suggested developments of the package, not yet implemented. Comments and suggestions are welcome.
- (3) `minitoc-ex.tex`, `mini-art.tex`, `second.tex`, `add.tex`, `addsec.tex`, `add.bib`, `hide1.tex`, `hide2.tex`, `apx.tex`, `app-mem.tex`, `mem.tex`, `mem1.tex`, `fo1.tex`, `fo2.tex`, `scr.tex`, `subf.tex`, `tsfc.tex`, `tbi.tex`, `2c.tex`, `bo.tex`, `hop.tex`, `cri.tex`, `livre.tex`, `ch0.tex`, `tlc.tex`, `mu.tex`, `hir.tex`, `hia.tex`, `gaps.tex`, are example files, to play with. The associated `.pdf` files are provided.
- (4) `minitoc.bug`, `minitoc.sum` are plain text documentation: list of problems (faq, see chapter 2 on page 53) and summary of commands (see chapter 3 on page 75).
- (5) `minitoc.ins`, `minitoc.ist`, `minitoc.pre`, `minitoc.lan`, `plainurl.bst`, `minitoc.dtx`, and `minitoc.bib` are the source of the documentation in (non perfect) english, `lamed.eps` and `lamed.pdf` are images to include, `lamed.tex` is the source code².
- (6) `minitoc.dtx`, `fminitoc.dtx`, `fminitoc.bib`, `fminitoc.ist`, `fminitoc.lan`, `fminitoc.pre`, `franc.sty`, `frbib.sty`, `frnew.sty`, `frplain1.bst` are the source (and tools) of the documentation in french³. `fminitoc.pre` is the preamble code for the documentation.
- (7) `minitoc.pdf`, is the documentation in (non perfect) english, in PDF format.
- (8) `fminitoc.pdf` is the documentation in french, in PDF format. The french documentation and source files must not be omitted.
- (9) `pmk` is a shell script⁴ to prepare the package and its documentation; you should adapt it to your needs. There are also four partial scripts (to be adapted):
- `imk`, which prepares the package from `minitoc.ins` and `minitoc.dtx`; note that `imk` must be run before running `emk` or `fmk`; it creates also some `.sty` files necessary to prepare the documentation but that are not to be installed;
 - `emk`, which prepares the english documentation from `minitoc.dtx`;
 - `fmk`, which prepares the french documentation from `fminitoc.dtx` and `minitoc.dtx`;
 - `rmk`, which sorts the files into classes (one directory for each class),
 - `xmk`, which typesets the example files (into PDF).

² Compiling `lamed.tex` requires a specific hebrew font and a specific package, plus conversion into EPS or PDF. I prefer to provide also the image files.

³ This seems rather strange. In fact, the english and french documentations are both in the `minitoc.dtx` file. `fminitoc.dtx` sets a flag then loads `minitoc.dtx`; hence the file `fminitoc.dtx` is much smaller than `minitoc.dtx`. Thus, `minitoc.ins` contains also some utility files which are automatically created (some `.sty` files, `minitoc.ist`, `fminitoc.ist`, `minitoc.lan`, `fminitoc.lan`). The english and french versions are not word-by-word translations, but they are in parallel in the file `minitoc.dtx`, and this helps the maintenance. Note that there is no more a `fminitoc.ins` file.

⁴ You can sip a big cappuccino ☕ while this script is running!

These scripts are currently written in C-shell, but there 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 sites, but these scripts prepare it (using the `duplex2v.pro` PostScript header file to try printing recto-verso).

- The files of (0) must be installed in a directory where $\text{\LaTeX} 2_{\epsilon}$ finds `.dtx` and `.ins` files.
- The files of (1) must be installed in a directory where $\text{\LaTeX} 2_{\epsilon}$ finds `.sty` files.
- The files of (2), (3), (4), (5), (6) and (9) must be installed in a separate directory, but must not be omitted.
- The files of (7) and (8) must be installed as on-line documentation.

Note that `minitoc.dtx` and hence `fminitoc.dtx` are (not so trivial) examples of using `minitoc` (with `hyperref`).

Chapter 8

Postface

Contents

8.1	The origins	219
8.2	New design in 1993	219
8.3	Developments in 1994	219
8.4	No developments in 1995	220
8.5	Developments en 1996	220
8.6	Developments in 1997	220
8.7	Developments in 1998	220
8.8	Developments in 1999	221
8.9	Developments in 2000	221
8.10	Developments in 2001	221
8.11	Developments in 2002	221
8.12	Developments in 2003	222
8.13	Developments in 2004	222
8.14	Developments in 2005	222
8.15	Developments in 2006	224

This chapter summarizes the evolution of the `minitoc` package, year by year. A more detailed history is available in “Changes History”, page [522](#). 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 the package) to change the allocation 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 `mtcoff` (`minitocoff` at this time) package and a rewrite of `minitoc` to extract the data from the `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 forming of the mini-tables: spacing was improved and the position of the title became ajustable (with the optional parameter of `\minitoc` or `\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 $\text{\LaTeX} 2_{\epsilon}$, replacing the ageing $\text{\LaTeX} 2.09$, some work was necessary to support the compatibility with this new version. This was not easy, but Denis ROEGEL and Frank MITTELBACH gave me many helpful hints.

Another major addition is the language option feature, with the concept of the *minitoc language definition file* (or `.mld` file), coming from the `babel` package [30, 31], by Johannes BRAAMS. But at this time, these files were still named as `.sty` files.

An important simplification is introduced by the notion of “absolute numbering”, which 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 en 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 `.mld`, more specific, and english is the default language. Some new languages are added.

The work on the starred sectionning commands continues.

The `minitocoff` companion package is renamed `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 `shorttext` package option and by the new autoconfiguration feature.

For the starred sectionning commands, the `\addstarredchapter` commande is added (with analog commands for starred parts and sections).

8.7 Developments in 1998

The `tight` and `loose` package options are added to improve the line spacing in the mini-tables.

8.8 Developments in 1999

The dotted and 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 `hyperref` package [179], thanks to Heiko OBERDIEK, Didier VERNA and Bernd JAEHNE.

Some corrections about the starred sectionning commands are added by Heiko OBERDIEK.

The mini-tables features (`\beforeparttoc` 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 [210] 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 [9] (by Donald ARSENEAU) is used.

The font commands are made compatible with the `memoir` class [213, 214].

Added compatibility with the `notoccite` package [8].

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 [209].

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 `\mtcsetttitlefont` commands (from a suggestion by Benjamin BAYART) to replace many font commands by only one command with a better user interface.

Comments about the \mathcal{AMS} classes (some ones are incompatible with `minitoc`).

Added the `\mtcsetformat` and `\mtcsetttitle` commands, again to have a simpler user interface.

Added various hints (`insection` package, 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 coherence of `\dominitoc`/`\minitoc` and co.

Added the `\mtcsetfeature` command (very complex).

Added a hint about the `abstract` package [208].

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 [153], 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 document classes [164].

Added a patch for the recent version of the `memoir` class [213, 214].

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 contains 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 [189].

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 [164], 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 [103]).

Added `mtcmess.sty` 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).

Added a hint about the `fncychap` package [146].

Added a hint about the `quotchap` package [193].

Added a hint about the `romannum` package [215].

Added a hint about the `sfheaders` package [148].

Added a hint about the `alnumsec` package [128].

Added a hint about the `captcont` package [85].

Added FAQ 44 and the `\mtcgapbeforeheads` and `\mtcgapafterheads` formatting commands.

Added the chapter 4, "Examples of documents", page 83.

Added FAQ 45 and the `\kernafterparttoc` and `co.` commands for the vertical space between a minitable and its bottom rule.

Acknowledgments

I need to thank the following peoples¹, for their help, their questions, their interventions in the news groups², and/or for their packages, classes and documents:

Hassan ABOLHASSANI, Nabil ABU EL-ATA, Juan M. AGUIRREGABIRIA, Alex AJ, Jérôme ANDRIEUX, Achod André ARADIAN, Einar ÁRNASON, Tim ARNOLD, Donald ARSENEAU, David ASPINALL, Gonvcal BADENES, Philipp BACHMANN, Marin BALGARENSKY, Benjamin BAYART, Thierry BAYET, Stephan P. VON BECHTOLSHEIM, Claudio BECCARI, Emmanuel BEFFARA, József BÉRCES, Karl BERRY, Javier BEZOS, Árpád BÍRÓ, Laurent BLOCH, Georgi BOSHPANOV, Daniel BOURBONNAIS, Victor BOYKO, Johannes BRAAMS, Mustafa BURC, Olivier CARDI, David CARLISLE, Manuel CARRIBA, Régis CASPAR, CHANOPD Silpa-Anan, Jean-Côme CHARPENTIER, Jana CHLEBIKOVA, Steven Douglas COCHRAN, Maurizio CODOGNO, David B. COOK, Oliver CORFF, Prakash COUNTCHAM, Serguei DACHIAN, Adrian DAERR, Sébastien DEMOUSTIER, Éric DEPARDIEU, Ben DE RYDT, Dorjpalam DORJ, Lyndon DUDING, Marko ÈEHAJA, Victor EIJKHOUT, Behdad ESFAHDOD, Thomas ESSER, Karl F. EVERITT, Mike FABIAN, Robin FAIRBAIRNS, Simon FEAR, Jürgen FENN, 'l2tabu Jeff FESSLER, Ulrike FISCHER, Daniel FLIPO, Jim FOX, Diego GARCIA MORATE, Hubert GÄSSLEIN, Mohammad GHODSI, Markus GLEISZNER, Vitali GONTSHARUK, Michel GOOSSENS, Mathieu GOUTELLE, Fraser GRANT, Norman GRAY, Loïc GRENON, Sébastien GROT, Marion GUNN, Boumediene HAMZI, Patrick HAPPEL, Yannis HARALAMBOUS, Alexander HARIN, Danny HEAP, Thorsten HEIN, Sten HELLMAN, Haavard HELSTRUP, Yvon HENEL, Stephan HENNIG, Florence HENRY, Morten HØGHOLM, Alv Kjetil HOLME, Umstatter HORST, Don HOSEK, Yufan HU, Dave W. HUSEBY, Dmitry IVANOV, Per Steinar IVERSEN, Zunbeltz IZAOLA, Youssef JABRI, 'Arabi Bernd JAEHNE, Radwan JALAM, Michael JANISH, Regnor JERNSLETEN, Zhuhan JIANG, Loïc JOLY, Christophe JORSSEN, Robert JUHASZ, JUNGSHIK Shin, Dan JURAFSKY, Mikko KANERVA, David KASTRUP, Ronan KERYELL, Axel KIELHORN, Bil KLEB, Peter KLEIWEG, Rune KLEVELAND, Jörg KNAPPEN, KOAUNGHU Un, Markus KOHM, Attila KOPPANYI, Thankmar KRONZUCKER, Alexej M. KRYUKOV, Toshiaki KUMAZAWA, Frank KÜSTER, Klaus LAGALLY, Fabio LANARI, Robert LANGE, Dag LANGMYHR, Olga LAPKO, Henning LARSEN, Jean-Marc LASGOUTTES, Jean-Philippe LAUFFENBURGER, Arnaud LAUNAY, Claire LAUVERNET, Boris LAVVA, Werner LEMBERG, Thomas LEONHARDT, Erwan LE PENNEC, Stéphane LEPOLOZEC, Julien LE THUAUT, Adam LEWENBERG, Knut LICKERT, Arnaldo Viegas DE LIMA, Ulf A. LINDGREN, Anselm LINGNAU, Pierre LOBEL, Milos LOKAJICEK, Maurizio LORETI,

¹ And I apologize to whose I forgot.

² Mainly, `fr.comp.text.tex` (in french) and `comp.text.tex` (in english), but also `de.comp.text.tex` (in german, but I do not read it well).

Tristan LORINO, Dan LUECKING, Ken LUNDE, Anders LYHNE, Pierre MACKAY,
 Lars MADSEN, Irina MAKHOVAYA, Bob MARGOLIS, Nicolas MARKEY, Marcus MARR,
 Françoise MARRE-FOURNIER, Terry MART, Ruiz-Altaba MARTI, Vadim MASLOV,
 Sven MATTISSON, Rowland McDONNELL, Ben McKAY, Sébastien MENGIN, Yanick MICHOU,
 Frank MITTELBACH, Lapo Filippo MORI, Michael A. MORRISON, Dejan MUHAMEDAGIĆ,
 Sergei O. NAOUMOV, Cuong NGUYEN, Julien NICOLAS, Rolf NIEPRASCHK, Josselin NOIREL,
 Heiko OBERDIEK, Mariusz OLKO, Piet VAN OOSTRUM, Erik ÖSTHOLS, Alan PAIĆ,
 Palash Baran PAL, Oren PATASHNIK, Matthias PELGER, Terje Engeset PETTERST,
 Bruno PIGUET, John PLAICE, POONLAP Veerathanabutr, Nico POPPELIER, Rama PORRAT,
 Camille-Aimé POSSAMAÏ, Roozbeh POURNADER, Sebastian RAHTZ, Bernd RAICHLE,
 Jose Pedro RAMALHETE, Keranen REINO, Adrian REZUŞ, Tony ROBERTS, Denis ROEGEL,
 Christian ROLLAND, Jan Michel RYNNING, Young RYU, Enn SAAR, David SAMSOEN,
 Julio SANCHEZ, Elmar SCHALUECK, Tobias SCHLEMMER, Uwe SCHNEIDER, Martin SCHRÖDER,
 Ali SHOUKAT, Andrij SHVAIKA, Jankovic SLOBODAN, Robin S. SOCHA, Arjen STEINER,
 Éric STREIT, Ralf STUBNER, SURAPANT Meknavin, Apostolos SYROPOULOS,
 Dung TA QUANG, Daniel TAUPIN[†], Hàn Thế THÀNH, THEPPITAK Karoonboonyanan,
 Karsten TINNEFELD, Sigitas TOLUSIS, Laurent TORDELLA, Mark TRETTIN, Ahto TRUU,
 Stefan ULRICH, Turgut UYAR, Jari VAARIO, Thomas VAN OUDENHOVE, Vincent VAQUIN,
 Didier VERNA, Sylvain VESCO, Alexandre VIAL, Vladimir VOLOVITCH, Nigel WARD,
 Staszek WAWRYKIEWICZ, Stephan WEBANCK, Eduard WERNER, Peter R. WILSON,
 Marcin WOLIŃSKI, Élisabeth WOLKOWSKI, Tim WRIGHT, Damien WYART,
 C.S. YOGANANDA, Adi ZAIMI, Danilo ZAVRTANIK, Krzysztof Konrad ŻELECHOWSKI,
 Vadim V. ZHYTNIKOV, and Leon ŽLAJPAH.

Part II

Implementation

Contents of the Second Part

9	Commented code of <code>minitoc.sty</code>	230
10	Commented code of <code>mtcoff.sty</code>	395
11	Commented code of <code>mtcmess.sty</code>	409
12	Patch for the <code>memoir</code> class	410
13	The language definition (<code>.mld</code>) and object (<code>.mlo</code>) files	412

Chapter 9

Commented code of minitoc.sty

Contents

9.1	Introduction	233
9.2	Identification code	234
9.3	A file descriptor to write	234
9.4	Indentation and skip	234
9.5	Tests and flags	235
9.5.1	Flags for the <code>hints</code> option	236
9.5.2	Use of section-level mini-lists of floats	236
9.5.3	Presence of some packages and classes	237
9.5.4	Presence or absence of some sectionning commands	238
9.5.5	Flags to check to see if some commands are used	239
9.5.6	Check if the document has exactly 2 parts	240
9.6	Preparation for the <code>notoccite</code> option	240
9.7	Preparation for the <code>tight</code> and <code>k-tight</code> options	241
9.8	Preparation to work with <code>hyperref</code>	241
9.9	Checking the presence of some packages	242
9.9.1	Check if the <code>sectsty</code> package is loaded, and when	242
9.9.2	Check if the <code>varsects</code> package is loaded, and when	242
9.9.3	Check if the <code>fncychap</code> package is loaded, and when	242
9.9.4	Check if the <code>quotchap</code> package is loaded, and when	242
9.9.5	Check if the <code>romannum</code> package is loaded, and when	243
9.9.6	Check if the <code>sfheaders</code> package is loaded, and when	243
9.9.7	Check if the <code>alnumsec</code> package is loaded, and when	243
9.9.8	Check if the <code>captcont</code> package is loaded, and when	243
9.9.9	Check if the <code>caption</code> package is loaded, and when	244
9.9.10	Check if the <code>caption2</code> package is loaded, and when	244
9.9.11	Check if the <code>ccaption</code> package is loaded, and when	244
9.9.12	Check if the <code>mcaption</code> package is loaded, and when	244
9.10	Is the <code>memoir</code> class loaded?	245
9.11	Testing the emptiness of a file	245
9.12	Internal macros to decrement <code>minitoc</code> counters	247

9.13	Patching the <code>\part</code> command	247
9.14	Adding an entry in the TOC for a starred part	248
9.15	Section level macros	248
9.16	Corrections for numbering	249
9.17	Patching the <code>\section</code> command	249
9.18	Adding an entry in the TOC for a starred section	249
9.19	Chapter level macros	250
9.20	Patching the <code>\chapter</code> command	250
9.21	Adding an entry in the TOC for a starred chapter	250
9.22	Miscellaneous declarations	251
9.23	Autoconfiguration of extensions	252
9.24	Detecting obsolete versions of L ^A T _E X	253
9.25	A macro to make a TOC entry without leaders nor page numbers	253
9.26	Default values for the page-number customization macros	254
9.27	“Features” for the mini-tables	256
9.28	Fake tables of contents	259
9.29	Depth counters for minilofs and minilots	259
9.30	Chapter level commands	260
9.31	Starred parts, chapters or sections	260
9.32	Font commands for the mini-tables	261
9.33	Internal commands to position the mini-table titles	262
9.34	The <code>mtc@verse</code> environment	263
9.35	The <code>\minitoc</code> , <code>\minilof</code> , and <code>\minilot</code> commands	263
9.35.1	The <code>\minitoc</code> command	263
9.35.2	The <code>\minilof</code> command	266
9.35.3	The <code>\minilot</code> command	269
9.36	Patching the <code>\chapter</code> command, continued	272
9.37	The <code>\addstarred...</code> commands	273
9.38	TOC entries without leaders	274
9.39	Mini-tables with or without leaders	274
9.40	The <code>\dominitoc</code> command and its siblings	275
9.40.1	Analysis and splitting of the TOC file	278
9.41	Mini-lists of figures	282
9.41.1	Analysis and splitting of the list of figures file	282
9.42	Mini-lists of tables	284
9.42.1	Analysis and splitting of the list of tables file	285
9.43	Macro to write a contents line	287
9.44	Depth counters for partlofs and partlots	288
9.45	Part level commands	289
9.46	Fonts for the parttocs	289
9.47	Default titles for part-level mini-tables	290
9.48	The <code>ptc@verse</code> environment	292
9.49	The part level mini-tables: <code>\parttoc</code> , <code>\partlof</code> , and <code>\partlot</code>	293
9.49.1	The <code>\parttoc</code> command	293
9.49.2	The <code>\partlof</code> command	296
9.49.3	The <code>\partlot</code> command	299
9.50	Auxiliary commands for printing parttocs	301
9.51	Patching the <code>\part</code> command, continued	302
9.52	The <code>\doparttoc</code> command and its siblings	302
9.52.1	Processing macros for the parttocs	305
9.52.2	Processing macros for the partlofs	308
9.52.3	Processing macros for the partlots	310
9.53	Depth counters for sectlofs and sectlots	313

9.54	Section-level commands	313
9.55	Fonts commands for secttocs and co.	314
9.56	Internal macros for title positioning	314
9.57	The stc@verse environment	315
9.58	The \secttoc, \sectlof, and \sectlot commands	315
9.58.1	The \secttoc command	315
9.58.2	The \sectlof command	318
9.58.3	The \sectlot command	321
9.59	Auxiliary internal commands, section level	323
9.60	Patching the \section command (continued)	324
9.61	The \dosecttoc command and siblings	325
9.62	End of section-level commands	334
9.63	The \mtcprepare command	335
9.64	Necessary \l@... commands	335
9.65	The horizontal rules and their default values	336
9.66	The \mtcset... commands	336
9.66.1	Keywords for the \mtcset... commands	337
9.66.2	The \mtcsetfont command	338
9.66.3	The \mtcsetttitlefont command	342
9.66.4	The \mtcsetttitle command	344
9.66.5	The \mtcsetformat command	345
9.66.6	The \mtcsetpagenumbers command	349
9.66.7	The \mtcsetrules command	351
9.66.8	The \mtcsetfeature command	353
9.66.9	The \mtcsetdepth command	356
9.67	The mtchideinmaintoc environment and siblings	357
9.68	Fixing the “Glossary” entry in the TOC	359
9.69	Fixing the “Index” entry in the TOC	361
9.70	The \mtcselectlanguage command	362
9.71	The \mtcloadmlo internal command	362
9.72	The “coffee breaks”	363
9.73	Initialization of counters	363
9.74	Declarations for simple options	363
9.74.1	Options tight and loose, k-tight and k-loose	364
9.74.2	Options checkfiles and nocheckfiles	364
9.74.3	Options dotted and undotted	364
9.74.4	Option notoccite	364
9.74.5	Option shorttext	364
9.75	The insection option	365
9.76	The listfiles and nolistfiles options	365
9.77	Language options	365
9.78	The hints option	369
9.78.1	First part: \mtc@hints@begindoc	369
9.78.1.1	Hint about the alphanum package	370
9.78.1.2	Hint about the appendix package	370
9.78.1.3	Hint about the tocbibind package	370
9.78.1.4	Hint about the KOMA-Script classes	370
9.78.1.5	Hint about the tocloft package	371
9.78.1.6	Hint about the titletoc package	371
9.78.1.7	Hint about the placeins package	371
9.78.1.8	Hint about the memoir class	373
9.78.1.9	Hint about the amsart and amsproc classes	373
9.78.1.10	Hint about the amsbook class	373
9.78.1.11	Hint about the abstract package	374

9.78.1.12	Hint about the <code>jura</code> class	374
9.78.1.13	Hint about the alteration of the sectionning commands	374
9.78.1.13.1	Alteration of <code>\part</code>	374
9.78.1.13.2	Alteration of <code>\chapter</code>	375
9.78.1.13.3	Alteration of <code>\section</code>	375
9.78.1.14	Hint about the coherence of the calling sequences of some commands	376
9.78.2	Final part: <code>\mtc@hints@enddoc</code>	377
9.78.2.1	Hint about <code>\sect-lof lot</code> and the <code>insection</code> option	377
9.78.2.2	Final part of the coherence tests	378
9.78.2.3	Check if the main tables have been prepared (first part)	380
9.78.2.4	Check if the main tables have been prepared (second part)	381
9.78.2.5	Check the number of mini-tables, in case of short extensions	383
9.78.2.6	Final part of the hint about the <code>sectsty</code> package	384
9.78.2.7	Final part of the hint about the <code>varsects</code> package	385
9.78.2.8	Final part of the hint about the <code>fncychap</code> package	385
9.78.2.9	Final part of the hint about the <code>quotchap</code> package	385
9.78.2.10	Final part of the hint about the <code>romannum</code> package	386
9.78.2.11	Final part of the hint about the <code>sfheaders</code> package	386
9.78.2.12	Final part of the hint about the <code>alnumsec</code> package	386
9.78.2.13	Final part of the hint about the <code>captcont</code> package	387
9.78.2.14	Final part of the hint about the <code>caption</code> package	387
9.78.2.15	Final part of the hint about the <code>caption2</code> package	387
9.78.2.16	Final part of the hint about the <code>ccaption</code> package	388
9.78.2.17	Final part of the hint about the <code>mcaption</code> package	388
9.78.2.18	Check if empty mini-tables have been detected	388
9.78.2.19	Check if obsolete commands have been used	390
9.78.2.20	Check if some hints have been written	391
9.79	Processing of options	391
9.79.1	Processing the <code>insection</code> option	391
9.79.2	Processing the <code>notoccite</code> option	392
9.79.3	Processing the <code>listfiles</code> option	392
9.79.4	Processing the <code>hints</code> option	393
9.79.5	Saving the sectionning commands	393
9.80	The <code>fminitoc.dtx</code> file	394

Figures

9.1	Hack to detect the limitation to short extensions.	252
-----	--	-----

9.1 Introduction

This long chapter presents the code of the `minitoc` package and attempts to explain it. Some comments of the original source file ¹ are skipped, like the history, because they do not need further examination (they will be put in the change history).

¹ The source file of version #42. Version #43 includes the conversion of the package to `.dtx-.ins` format. Version #42 has not been distributed because of that.

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 `minitoc` external commands have `mtc`, `ptc`, `stc`, or one of the mini-table names (`parttoc`, ..., `sectlot`) in their names. Most of the `minitoc` internal commands have `@mtc`, `@ptc`, `@stc`, or `parttoc@`, ..., `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 `minitoc.sty` starts here:

```
1 (*minitoc)

\NeedsTeXFormat{LaTeX2e}[1996/06/02]%
\ProvidesPackage{minitoc}%
\mtcPackageInfo{minitoc}%
\RequirePackage{mtcmess}[2006/03/14]
2 \NeedsTeXFormat{LaTeX2e}[1996/06/02]%
3 \ProvidesPackage{minitoc}%
4   [2006/08/03 v47 Package minitoc (JPFD)]
5 \RequirePackage{mtcmess}[2006/03/14]
6 \mtcPackageInfo[<I0001>]{minitoc}%
7   {*** minitoc package, version 47 ***\@gobble}
```

9.3 A file descriptor to write

```
\tf@mtc A file descriptor is needed to write the files containing the mini-tables, it is \tf@mtc.
\newwrite
8 \newwrite\tf@mtc
```

9.4 Indentation and skip

```
\mtcindent We define the indentation \mtcindent (both sides) of the mini-tables and the command
\mtcskip \mtcskip to make a vertical skip before a mini-table, its value is \mtcskipamount
\mtcskipamount (default: \bigskipamount).
\parskip
9 \newlength\mtcindent
10 \newskip\mtcskipamount
11 \setlength{\mtcskipamount}{\bigskipamount}
12 % %%%\def\mtcskip{\leavevmode\unskip\removelastskip%
```

```

13 % %%%          {\parskip=\z@\addvspace{\mtcskipamount}}
14 \def\mtcskip{{\parskip=\z@\addvspace{\mtcskipamount}}}

```

Note that `\mtcskip` uses a local group to avoid the influence of `\parskip`.

`\mtcgapbeforeheads` We define the default values for the vertical gaps before and after titles part level
`\mtcgapafterheads` mini-tables.

```

15 \def\mtcgapbeforeheads{50\p@}
16 \def\mtcgapafterheads{40\p@}

```

`\@ifundefined` We define the vertical kernings between the minitables and their before the bottom rule.
`\kernafterparttoc` The names of these macros is rather explicit. The values are empirical and can be changed
`\kernafterpartlof` via `\renewcommand`.
`\kernafterpartlot`

```

\kernaftersecttoc 17 \ifundefined{part}{}%
\kernaftersectlof 18 \def\kernafterparttoc{\kern-1.\baselineskip\kern.5ex}%
\kernaftersectlot 19 \def\kernafterpartlof{\kern-1.\baselineskip\kern.5ex}%
\kernafterminitoc 20 \def\kernafterpartlot{\kern-1.\baselineskip\kern.5ex}%
\kernafterminilof 21 }%
\kernafterminilot 22 \@ifundefined{chapter}{}%
23 \ifundefined{section}{}%
24 {}%
25 \def\kernaftersecttoc{\kern-1.\baselineskip\kern.5ex}%
26 \def\kernaftersectlof{\kern-1.\baselineskip\kern.5ex}%
27 \def\kernaftersectlot{\kern-1.\baselineskip\kern.5ex}%
28 }%
29 }%
30 {}%
31 \def\kernafterminitoc{\kern-.5\baselineskip\kern.5ex}%
32 \def\kernafterminilof{\kern-1.\baselineskip\kern0.ex}%
33 \def\kernafterminilot{\kern-1.\baselineskip\kern0.ex}%
34 }%

```

9.5 Tests and flags

We need to declare some flags² (via `\newif`) to detect the loading of some packages or classes and the availability of some commands (this will be used by the `hints` option (section 9.78 on page 369) or to allow the definition of some `minitoc` commands).

² 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 TeX engine (much more with ϵ -TeX, 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 than a code programmed with counters, IMHO.

```

\if@mtc@hints@
\if@mtc@hints@w@
\if@mtc@hints@giben@

```

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 `.log` file. It will be used at the end of the document to signal that you should look for hints in the `.log` file.

```

35 \newif\if@mtc@hints@ \@mtc@hints@true
36 \newif\if@mtc@hints@w@ \@mtc@hints@w@false
37 \newif\if@mtc@hints@given@ \@mtc@hints@given@false

```

9.5.2 Use of section-level mini-lists of floats

```

\if@mtc@sect@floats@
\dosectlof
\dosectlot

```

We will check if the commands `\dosectlof` and `\dosectlot` are used:

```

38 \newif\if@mtc@sect@floats@ \@mtc@sect@floats@false

```

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 version), then if the `sectsty` package is loaded (before or after `minitoc`). We do the same for some caption-related packages and the `varsects` package and others package altering the sectionning commands.

```

\if@mtc@placeinsLoaded@
\if@mtc@memoirLoaded@
\if@mtc@memoirnew@
\if@mtc@sectstyLoaded@
\if@mtc@sectstyLoaded@a@
\if@mtc@captionLoaded@
\if@mtc@captionLoaded@a@
\if@mtc@captionIILoaded@
\if@mtc@captionIILoaded@a@
\if@mtc@ccaptionLoaded@
\if@mtc@ccaptionLoaded@a@
\if@mtc@mcaptionLoaded@
\if@mtc@mcaptionLoaded@a@
\if@mtc@varsectsLoaded@
\if@mtc@varsectsLoaded@a@
\if@mtc@fncychapLoaded@
\if@mtc@fncychapLoaded@a@
\if@mtc@quotchapLoaded@
\if@mtc@quotchapLoaded@a@
\if@mtc@romannumLoaded@
\if@mtc@romannumLoaded@a@
\if@mtc@sfheadersLoaded@
\if@mtc@sfheadersLoaded@a@
\if@mtc@alnumsecLoaded@
\if@mtc@alnumsecLoaded@a@
\if@mtc@captcontLoaded@
\if@mtc@captcontLoaded@a@

39 \newif\if@mtc@placeinsLoaded@ \@mtc@placeinsLoaded@false
40 \newif\if@mtc@memoirLoaded@ \@mtc@memoirLoaded@false
41 \newif\if@mtc@memoirnew@ \@mtc@memoirnew@false
42 \newif\if@mtc@sectstyLoaded@ \@mtc@sectstyLoaded@false
43 \newif\if@mtc@sectstyLoaded@a@ \@mtc@sectstyLoaded@a@false
44 \newif\if@mtc@captionLoaded@ \@mtc@captionLoaded@false
45 \newif\if@mtc@captionLoaded@a@ \@mtc@captionLoaded@a@false
46 \newif\if@mtc@captionIILoaded@ \@mtc@captionIILoaded@false
47 \newif\if@mtc@captionIILoaded@a@ \@mtc@captionIILoaded@a@false
48 \newif\if@mtc@ccaptionLoaded@ \@mtc@ccaptionLoaded@false
49 \newif\if@mtc@ccaptionLoaded@a@ \@mtc@ccaptionLoaded@a@false
50 \newif\if@mtc@mcaptionLoaded@ \@mtc@mcaptionLoaded@false
51 \newif\if@mtc@mcaptionLoaded@a@ \@mtc@mcaptionLoaded@a@false
52 \newif\if@mtc@varsectsLoaded@ \@mtc@varsectsLoaded@false
53 \newif\if@mtc@varsectsLoaded@a@ \@mtc@varsectsLoaded@a@false
54 \newif\if@mtc@fncychapLoaded@ \@mtc@fncychapLoaded@false
55 \newif\if@mtc@fncychapLoaded@a@ \@mtc@fncychapLoaded@a@false
56 \newif\if@mtc@quotchapLoaded@ \@mtc@quotchapLoaded@false
57 \newif\if@mtc@quotchapLoaded@a@ \@mtc@quotchapLoaded@a@false
58 \newif\if@mtc@romannumLoaded@ \@mtc@romannumLoaded@false
59 \newif\if@mtc@romannumLoaded@a@ \@mtc@romannumLoaded@a@false
60 \newif\if@mtc@sfheadersLoaded@ \@mtc@sfheadersLoaded@false
61 \newif\if@mtc@sfheadersLoaded@a@ \@mtc@sfheadersLoaded@a@false
62 \newif\if@mtc@alnumsecLoaded@ \@mtc@alnumsecLoaded@false
63 \newif\if@mtc@alnumsecLoaded@a@ \@mtc@alnumsecLoaded@a@false
64 \newif\if@mtc@captcontLoaded@ \@mtc@captcontLoaded@false
65 \newif\if@mtc@captcontLoaded@a@ \@mtc@captcontLoaded@a@false

\if@mtc@empty@parttoc@
\if@mtc@empty@partlof@
\if@mtc@empty@partlot@
\if@mtc@empty@minitoc@
\if@mtc@empty@minitoc@
\if@mtc@empty@minilof@
\if@mtc@empty@minilof@
\if@mtc@empty@sectlot@
\if@mtc@empty@sectlof@
\if@mtc@empty@sectlot@

66 \newif\if@mtc@empty@parttoc@ \@mtc@empty@parttoc@false
67 \newif\if@mtc@empty@partlof@ \@mtc@empty@partlof@false
68 \newif\if@mtc@empty@partlot@ \@mtc@empty@partlot@false
69 \newif\if@mtc@empty@minitoc@ \@mtc@empty@minitoc@false
70 \newif\if@mtc@empty@minilof@ \@mtc@empty@minilof@false
71 \newif\if@mtc@empty@minilof@ \@mtc@empty@minilof@false
72 \newif\if@mtc@empty@secttoc@ \@mtc@empty@secttoc@false
73 \newif\if@mtc@empty@sectlof@ \@mtc@empty@sectlof@false
74 \newif\if@mtc@empty@sectlot@ \@mtc@empty@sectlot@false

```

We will check if you have attempted to insert some empty mini-tables:

9.5.4 Presence or absence of some sectionning commands

We define and set flags about the presence of the sectionning commands (in fact, the counters associated with these commands).

`\if@mtc@part@def@` The part counter:

```
75 \newif\if@mtc@part@def@ \@mtc@part@def@false
76 \ifundefined{part}{\@mtc@part@def@false}{\@mtc@part@def@true}
```

`\if@mtc@chapter@def@` The chapter counter:

```
77 \newif\if@mtc@chapter@def@ \@mtc@chapter@def@false
78 \ifundefined{chapter}{\@mtc@chapter@def@false}{\@mtc@chapter@def@true}
```

`\if@mtc@section@def@` The section counter:

```
79 \newif\if@mtc@section@def@ \@mtc@section@def@false
80 \ifundefined{section}{\@mtc@section@def@false}{\@mtc@section@def@true}
```

We define and set flags about the absence of the sectionning commands.

`\if@mtc@part@undef@` The part counter:

```
81 \newif\if@mtc@part@undef@ \@mtc@part@undef@true
82 \ifundefined{part}{\@mtc@part@undef@true}{\@mtc@part@undef@false}
```

`\if@mtc@chapter@undef@` The chapter counter:

```
83 \newif\if@mtc@chapter@undef@ \@mtc@chapter@undef@true
84 \ifundefined{chapter}{\@mtc@chapter@undef@true}{\@mtc@chapter@undef@false}
```

`\if@mtc@section@undef@` The section counter:

```
85 \newif\if@mtc@section@undef@ \@mtc@section@undef@true
86 \ifundefined{section}{\@mtc@section@undef@true}{\@mtc@section@undef@false}
```

9.5.5 Flags to check to see 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 9.78 on page 369).

```
\if@parttoc@used@ For the part level:
\if@partlof@used@
\if@partlot@used@ 87 \newif\if@parttoc@used@ \global\@parttoc@used@false
\if@doparttoc@used@ 88 \newif\if@partlof@used@ \global\@partlof@used@false
\if@dopartlof@used@ 89 \newif\if@partlot@used@ \global\@partlot@used@false
\if@dopartlot@used@ 90 \newif\if@doparttoc@used@ \global\@doparttoc@used@false
91 \newif\if@dopartlof@used@ \global\@dopartlof@used@false
92 \newif\if@dopartlot@used@ \global\@dopartlot@used@false
```

```
\if@minitoc@used@ For the chapter level:
\if@minilof@used@
\if@minilot@used@ 93 \newif\if@minitoc@used@ \global\@minitoc@used@false
\if@dominitoc@used@ 94 \newif\if@minilof@used@ \global\@minilof@used@false
\if@dominilof@used@ 95 \newif\if@minilot@used@ \global\@minilot@used@false
\if@dominilot@used@ 96 \newif\if@dominitoc@used@ \global\@dominitoc@used@false
97 \newif\if@dominilof@used@ \global\@dominilof@used@false
98 \newif\if@dominilot@used@ \global\@dominilot@used@false
```

```
\if@secttoc@used@ For the section level:
\if@sectlof@used@
\if@sectlot@used@ 99 \newif\if@secttoc@used@ \global\@secttoc@used@false
\if@dosecttoc@used@ 100 \newif\if@sectlof@used@ \global\@sectlof@used@false
\if@dosectlof@used@ 101 \newif\if@sectlot@used@ \global\@sectlot@used@false
\if@dosectlot@used@ 102 \newif\if@dosecttoc@used@ \global\@dosecttoc@used@false
103 \newif\if@dosectlof@used@ \global\@dosectlof@used@false
104 \newif\if@dosectlot@used@ \global\@dosectlot@used@false
```

```
\if@firstpartis@used@ We also detect the use of some obsolete commands:
\if@firstchapteris@used@
\if@firstsectionis@used@ 105 \newif\if@firstpartis@used@ \global\@firstpartis@used@false
106 \newif\if@firstchapteris@used@ \global\@firstchapteris@used@false
107 \newif\if@firstsectionis@used@ \global\@firstsectionis@used@false
```

9.5.6 Check if the document has exactly 2 parts

`\ifmtcsecondpart` 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. So we define a specific flag:

```
108 \newif\ifmtcsecondpart \mtcsecondpartfalse
```

`\AtBeginDocument` At the beginning of the document, we test this flag and make it global:

`\ifmtcsecondpart`

```
109 \AtBeginDocument{%
110   \ifmtcsecondpart
111     \global\mtcsecondparttrue
112   \else
113     \global\mtcsecondpartfalse
114   \fi}
```

`\AtEndDocument` At the end of the document, we set and memorize the value of the flag in the .aux file:

`\ifmtcsecondpart`

```
\\@mainaux
115 \AtEndDocument{%
116   \ifnum\value{part}=2\relax
117     \mtcsecondparttrue
118   \else
119     \mtcsecondpartfalse
120   \fi
121   \if@files
122     \ifmtcsecondpart
123       \immediate\write\@mainaux
124         {\string\global\string\mtcsecondparttrue}%
125     \else
126       \immediate\write\@mainaux
127         {\string\global\string\mtcsecondpartfalse}%
128     \fi
129   \fi}
```

So we need two \LaTeX runs to get a correct result. The `french2.mld` language definition file (see section 13.59 on page 444) uses this trick to form the titles of part level mini-tables. See the `second.tex` example file in section 4.5 on page 106.

9.6 Preparation for the notoccite option

`\mtc@hook@beforeinputfile` We declare a flag for the presence of this option and the new internal command
`\if@mtc@notoccite@` `\mtc@hook@beforeinputfile`, which is a “hook” command (redefinable command)

used by this option (requested by Donald ARSENEAU for his notoccite package [8]). See section 1.6 on page 49.

```
130 \newif\if@mtc@notoccite@ \@mtc@notoccite@false
131 \@ifundefined{mtc@hook@beforeinputfile}%
132   {\let\mtc@hook@beforeinputfile\relax}{} }
```

9.7 Preparation for the tight and k-tight options

`\iftightmtc` We just declare a flag for each of these options; they are set false by default (loose and `\ifktightmtc` k-loose options):

```
133 \newif\iftightmtc \tightmtcfalse
134 \newif\ifktightmtc \ktightmtcfalse
```

9.8 Preparation to work with hyperref

`\AtBeginDocument` This code prepares the interface with the hyperref package [179]. A flag is defined, then `\if@mtc@hyper@used@` this preparation is performed in a `\AtBeginDocument` block if this package is loaded. `\@ifpackageloaded` This action defines some commands for the hyperref package.

```
135 \mtcPackageInfo[<I0005>]{minitoc}{compatible with hyperref\@gobble}
136 \newif\if@mtc@hyper@used@ \global\@mtc@hyper@used@false
137 \AtBeginDocument{%
138   \@ifpackageloaded{hyperref}{%
139     \global\@mtc@hyper@used@true
140     \def\toclevel@xpart{1000}%
141     \def\toclevel@xchapter{1000}%
142     \def\toclevel@xsect{1000}%
143     \def\toclevel@xsection{1000}%
144     \let\toclevel@starpart\toclevel@part
145     \let\toclevel@starchapter\toclevel@chapter
146     \let\toclevel@starsection\toclevel@section
147     \let\toclevel@starsubsection\toclevel@subsection
148     \let\toclevel@starsubsubsection\toclevel@subsubsection
149     \let\toclevel@starparagraph\toclevel@paragraph
150     \let\toclevel@starsubparagraph\toclevel@subparagraph
151   }{}}
```

9.9 Checking the presence of some packages

9.9.1 Check if the sectsty package is loaded, and when

```

\AtBeginDocument We must test if the sectsty package [153] is loaded before or after minitoc, so we
\if@mtc@sectstyLoaded@ test when minitoc is loaded and also in a \AtBeginDocument block, when all packages
\if@mtc@sectstyLoaded@a@ have been loaded. See section 9.78.2.6 on page 384.
\@ifpackageloaded
152 \@ifpackageloaded{sectsty}{\@mtc@sectstyLoaded@true}{}
153 \AtBeginDocument{\@ifpackageloaded{sectsty}{\@mtc@sectstyLoaded@a@true}{}

```

9.9.2 Check if the varsects package is loaded, and when

```

\@ifpackageloaded We must test if the varsects package [189] is loaded before or after minitoc, so
\AtBeginDocument we test when minitoc is loaded and also in a \AtBeginDocument block, when all
\if@mtc@varsectsLoaded@ packages have been loaded. See section 9.78.2.7 on page 385.
\if@mtc@varsectsLoaded@a@
154 \@ifpackageloaded{varsects}{\@mtc@varsectsLoaded@true}{}
155 \AtBeginDocument{\@ifpackageloaded{varsects}{\@mtc@varsectsLoaded@a@true}{}

```

9.9.3 Check if the fncychap package is loaded, and when

```

\@ifpackageloaded We must test if the fncychap package [146] is loaded before or after minitoc, so
\AtBeginDocument we test when minitoc is loaded and also in a \AtBeginDocument block, when all
\if@mtc@fncychapLoaded@ packages have been loaded. See section 9.78.2.8 on page 385.
\if@mtc@fncychapLoaded@a@
156 \@ifpackageloaded{fncychap}{\@mtc@fncychapLoaded@true}{}
157 \AtBeginDocument{\@ifpackageloaded{fncychap}{\@mtc@fncychapLoaded@a@true}{}

```

9.9.4 Check if the quotchap package is loaded, and when

```

\@ifpackageloaded We must test if the quotchap package [193] is loaded before or after minitoc, so
\AtBeginDocument we test when minitoc is loaded and also in a \AtBeginDocument block, when all
\if@mtc@quotchapLoaded@ packages have been loaded. See section 9.78.2.9 on page 385.
\if@mtc@quotchapLoaded@a@
158 \@ifpackageloaded{quotchap}{\@mtc@quotchapLoaded@true}{}
159 \AtBeginDocument{\@ifpackageloaded{quotchap}{\@mtc@quotchapLoaded@a@true}{}

```

9.9.5 Check if the romannum package is loaded, and when

```

\@ifpackageloaded We must test if the romannum package [215] is loaded before or after minitoc, so
\AtBeginDocument we test when minitoc is loaded and also in a \AtBeginDocument block, when all
\if@mtc@romannumLoaded@ packages have been loaded. See section 9.78.2.10 on page 386.
\if@mtc@romannumLoaded@a@

160 \@ifpackageloaded{romannum}{\@mtc@romannumLoaded@true}{}
161 \AtBeginDocument{\@ifpackageloaded{romannum}{\@mtc@romannumLoaded@a@true}{} }

```

9.9.6 Check if the sfheaders package is loaded, and when

```

\@ifpackageloaded We must test if the sfheaders package [148] is loaded before or after minitoc, so
\AtBeginDocument we test when minitoc is loaded and also in a \AtBeginDocument block, when all
\if@mtc@sfheadersLoaded@ packages have been loaded. See section 9.78.2.11 on page 386.
\if@mtc@sfheadersLoaded@a@

162 \@ifpackageloaded{sfheaders}{\@mtc@sfheadersLoaded@true}{}
163 \AtBeginDocument{\@ifpackageloaded{sfheaders}{\@mtc@sfheadersLoaded@a@true}{} }

```

9.9.7 Check if the alnumsec package is loaded, and when

```

\@ifpackageloaded We must test if the alnumsec package [128] is loaded before or after minitoc, so
\AtBeginDocument we test when minitoc is loaded and also in a \AtBeginDocument block, when all
\if@mtc@alnumsecLoaded@ packages have been loaded. See section 9.78.2.12 on page 386.
\if@mtc@alnumsecLoaded@a@

164 \@ifpackageloaded{alnumsec}{\@mtc@alnumsecLoaded@true}{}
165 \AtBeginDocument{\@ifpackageloaded{alnumsec}{\@mtc@alnumsecLoaded@a@true}{} }

```

9.9.8 Check if the captcont package is loaded, and when

```

\@ifpackageloaded We must test if the captcont package [85] is loaded before or after minitoc, so
\AtBeginDocument we test when minitoc is loaded and also in a \AtBeginDocument block, when all
\if@mtc@captcontLoaded@ packages have been loaded. See section 9.78.2.13 on page 387.
\if@mtc@captcontLoaded@a@

166 \@ifpackageloaded{captcont}{\@mtc@captcontLoaded@true}{}
167 \AtBeginDocument{\@ifpackageloaded{captcont}{\@mtc@captcontLoaded@a@true}{} }

```

9.9.9 Check if the caption package is loaded, and when

```

\@ifpackageloaded
\AtBeginDocument
\if@mtc@captionLoaded@
\if@mtc@captionLoaded@a@
    We must test if the caption package [186] is loaded before or after minitoc, so we
    test when minitoc is loaded and also in a \AtBeginDocument block, when all packages
    have been loaded. See section 9.78.2.14 on page 387.
168 \@ifpackageloaded{caption}{\@mtc@captionLoaded@true}{}
169 \AtBeginDocument{\@ifpackageloaded{caption}{\@mtc@captionLoaded@a@true}{}

```

9.9.10 Check if the caption2 package is loaded, and when

```

\@ifpackageloaded
\AtBeginDocument
\if@mtc@captionIILoaded@
\if@mtc@captionIILoaded@a@
    We must test if the caption2 package [185] is loaded before or after minitoc, so
    we test when minitoc is loaded and also in a \AtBeginDocument block, when all
    packages have been loaded. See section 9.78.2.15 on page 387.
170 \@ifpackageloaded{caption2}{\@mtc@captionIILoaded@true}{}
171 \AtBeginDocument{\@ifpackageloaded{caption2}{\@mtc@captionIILoaded@a@true}{}

```

9.9.11 Check if the ccaption package is loaded, and when

```

\@ifpackageloaded
\AtBeginDocument
\if@mtc@ccaptionLoaded@
\if@mtc@ccaptionLoaded@a@
    We must test if the ccaption package [212] is loaded before or after minitoc, so
    we test when minitoc is loaded and also in a \AtBeginDocument block, when all
    packages have been loaded. See section 9.78.2.16 on page 388.
172 \@ifpackageloaded{ccaption}{\@mtc@ccaptionLoaded@true}{}
173 \AtBeginDocument{\@ifpackageloaded{ccaption}{\@mtc@ccaptionLoaded@a@true}{}

```

9.9.12 Check if the mcaption package is loaded, and when

```

\@ifpackageloaded
\AtBeginDocument
\if@mtc@mcaptionLoaded@
\if@mtc@mcaptionLoaded@a@
    We must test if the mcaption package [113] is loaded before or after minitoc, so
    we test when minitoc is loaded and also in a \AtBeginDocument block, when all
    packages have been loaded. See section 9.78.2.17 on page 388.
174 \@ifpackageloaded{mcaption}{\@mtc@mcaptionLoaded@true}{}
175 \AtBeginDocument{\@ifpackageloaded{mcaption}{\@mtc@mcaptionLoaded@a@true}{}

```

9.10 Is the memoir class loaded?

`\ifclassloaded` We test if the memoir [213, 214] class is loaded. This class needs some compatibility adjustments or may be incompatible if too recent. In the later case, a patch is inserted
`\if@mtc@memoirLoaded@` (see chapter 12 on page 410). This correction is no more necessary after the 2005/09/25
`\if@mtc@memoirnew@` version of memoir.cls.
`\if@mtcpatchmemoir@`

```

176 \newif\if@mtcpatchmemoir@ \@mtcpatchmemoir@false
177 \ifclassloaded{memoir}{\@mtc@memoirLoaded@true\relax%
178     \mtcPackageInfo[<I0030>]{minitoc}%
179     {the memoir class is loaded:\MessageBreak
180     compatibility attempted\@gobble}}%
181     {\@mtc@memoirLoaded@false}
182 \if@mtc@memoirLoaded@
183 \ifundefined{@m@chapter}%
184     {\@mtc@memoirnew@false\mtcPackageInfo[<I0020>]{minitoc}%
185     {old version of the memoir class\@gobble}}
186     {\@mtc@memoirnew@true\mtcPackageInfo[<I0027>]{minitoc}%
187     {recent version of the memoir class\@gobble}}
188     \mtcPackageInfo[<I0032>]{minitoc}%
189     {This version of the memoir class uses \MessageBreak
190     a version of \string\chapter\space which is \MessageBreak
191     incompatible with the minitoc package. \MessageBreak
192     We try to patch\@gobble}%
193     \@mtcpatchmemoir@true}
194 \fi

```

`\if@mtcpatchmemoir@` And now the patch:

```

195 \if@mtcpatchmemoir@
196 \IfFileExists{mtcpatchmem.sty}{%
197     \@ifclasslater{memoir}{2005/09/25}{}{\RequirePackage{mtcpatchmem}}}{%
198     \mtcPackage[<E0028>]{Error}{minitoc}
199     {Unable to patch the memoir class}%
200     {So it remains incompatible. Sorry.}}
201 \fi

```

9.11 Testing the emptiness of a file

`\mtc@ifmtarg` Some macros for testing if an argument of a macro is empty (taken from the package
`\mtc@xifmtarg` ifmtarg [216], by Peter R. WILSON and Donald ARSENEAU, and from while.tip, by
`\mtc@EndWhile` Stephan P. von BECHTOLSHEIM [203]). The group is necessary to keep local the catcode
`\mtc@WhilePreCondition` change of “Q”, hence a \gdef is needed for \mtc@ifmtarg.
`\mtc@WhileCondition`
`\mtc@WhileBody` 202 \begingroup
`\mtc@While` 203 \catcode'\Q=3
`\mtc@WhileNext`

```

204 \long\gdef\mtc@ifmtarg#1{%
205 \mtc@xifmtarg#1QQ\@secondoftwo\@firstoftwo\@nil}
206 \long\gdef\mtc@xifmtarg#1#2Q#3#4#5\@nil{#4}
207 \endgroup
208 \let\mtc@EndWhile = \fi
209 \def\mtc@While #1#2#3\mtc@EndWhile{%
210   \def\mtc@WhilePreCondition{#1}%
211   \def\mtc@WhileCondition{#2}%
212   \def\mtc@WhileBody{#3}%
213   \mtc@@While
214 }
215 \def\mtc@@While{%
216   \mtc@WhilePreCondition
217   \mtc@WhileCondition
218   \def\mtc@WhileNext{%
219     \mtc@WhileBody
220     \mtc@@While
221   }%
222   \else
223   \def\mtc@WhileNext{}%
224   \fi
225   \mtc@WhileNext
226 }

```

\if@mtc@checkfiles Some macros to test if a file is empty or not: \mtc@CkFile{file} returns

\if@mtc@FE \mtc@FEtrue if the file is empty, \mtc@FEfalse if the is file not empty. An

\if@mtc@LI inexistent file is empty. A file full of white space (space, tabulation, newline) is empty.

\mtc@While Comments are empty.

\mtc@Body

\mtc@EndWhile **Note:** on a big empty file, the \mtc@While loop may be time consuming, but not an

\mtc@CkFile eternity (33 s for 10⁶ lines on my computer), and the first non-empty line stops the loop.

\mtc@CkStr \jobname.mtc is used as scratch file. It is erased after use.

\mtc@Rline

\tf@mtc

\@inputcheck

```

227 \newif\if@mtc@LI\@mtc@LItrue
228 \newif\if@mtc@FE\@mtc@FEtrue
229 \newif\if@mtc@checkfiles\@mtc@checkfilestrue
230 \def\mtc@Body{\immediate\read\@inputcheck to
231                                     \mtc@Rline\relax
232   \ifeof\@inputcheck\relax\@mtc@LIfalse\fi
233   \expandafter\ifx\mtc@Rline\par\relax
234     \def\mtc@Rline{}
235   \else
236     \ifeof\@inputcheck\relax\global\@mtc@LIfalse\fi
237     \mtc@ifmtarg{\mtc@Rline}{\relax}%
238     {\@mtc@FEfalse\@mtc@LIfalse}
239   \fi
240 }
241 \def\mtc@CkFile#1{%
242 \@mtc@LItrue\@mtc@FEtrue
243 \if@mtc@checkfiles
244 \IfFileExists{#1}{%
245 \immediate\openin\@inputcheck #1\relax

```

```

246 \mtc@While{}{\if@mtc@LI\relax}%
247   {\mtc@Body}%
248 \mtc@EndWhile}%
249 {\@mtc@FEtrue}%
250 \else
251 \@mtc@FEfalse%
252 \fi}
253 \closein\@inputcheck\relax
254 \def\mtc@CkStr#1{%
255 \immediate\openout\tf@mtc \jobname.mtc
256 \immediate\write\tf@mtc{#1}%
257 \immediate\closeout\tf@mtc
258 \mtc@CkFile{\jobname.mtc}%
259 \immediate\openout\tf@mtc \jobname.mtc
260 \immediate\closeout\tf@mtc}

```

9.12 Internal macros to decrement minitoc counters

\mtc@onebackpart It is sometimes necessary to decrement a minitoc counter (ptc, mtc or stc) by one.
\mtc@onebackchapter These macros are:
\mtc@onebacksection
\addtocounter

```

261 \def\mtc@onebackpart{\addtocounter{ptc}{-1}}
262 \def\mtc@onebackchap{\addtocounter{mtc}{-1}}
263 \def\mtc@onebacksect{\addtocounter{stc}{-1}}

```

9.13 Patching the \part command

\part If the \part command is not defined (by the document class, usually), we cannot patch it
\mtc@svspart and a warning is displayed³. Else, we patch its two branches, \@part (for the unstarred
\mtc@svpart version) or \@spart (for the starred version, \part*): we add \stepcounter{ptc} to
\@spart increment the parttoc counter ptc. See also section 9.51 on page 302.
\@part
\stepcounter The code of the next section (section 9.14 on the next page) is also skipped if \part is
not defined.

```

264 \@ifundefined{part}{%
265   \mtcPackageWarningNoLine[<W0018>]{minitoc}%
266   {part level macros NOT available}
267 }{%% else undefined part (\part defined)
268   \mtcPackageInfo[<I0023>]{minitoc}%
269   {part level macros available\@gobble}
270   \let\mtc@svspart\@spart

```

³ Document classes with sectioning commands but no \part command are likely non standard, hence the warning displayed on the terminal.

```

271 \def\@spart{\stepcounter{ptc}\mtc@svspart}
272 \let\mtc@svpart\@part
273 \def\@part{\stepcounter{ptc}\mtc@svpart}

```

9.14 Adding an entry in the TOC for a starred part

`\mtcaddpart` 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.

`\mtc@ifmtarg`

`\contentsline` 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 `\l@xpart` `\contentsline{xpart}{}...` line in the `.lof` and `.lot` files. Then we increment the `ptc` counter, via `\adjustptc` (defined in section 9.45 on page 289). 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 (10 000) for this entry, hence it will never be really printed (except if you cheat).

```

274 \newcommand{\mtcaddpart}[1][{}]{%
275     \mtc@ifmtarg{#1}{\addcontentsline{toc}{xpart}{}}%
276         {\addcontentsline{toc}{part}{#1}}%
277     \addcontentsline{lof}{xpart}{}%
278     \addcontentsline{lot}{xpart}{}%
279     \adjustptc}

```

This code terminates (temporarily) the part level commands.

```

280 }%

```

9.15 Section level macros

`\chapter` 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`.

`\section` 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.

```

281 \@ifundefined{chapter}{\mtcPackageInfo[<I0004>]{minitoc}%
282     {chapter level macros NOT available\@gobble}}%
283 \@ifundefined{section}{\mtcPackageInfo[<I0029>]{minitoc}%
284     {section level macros NOT available\@gobble}}%
285 \mtcPackageWarningNoLine[<W0017>]{minitoc}%
286     {no section or chapter level macros available
287     \MessageBreak}

```



```

288     PLEASE VERIFY YOUR MAIN DOCUMENT CLASS}}%
289   {\mtcPackageInfo[<I0028>]{minitoc}%
290     {section level macros available\@gobble}%

```

9.16 Corrections for numbering

```

\mtc@onebacksect As the TOC, the LOF and the LOT are considered as (starred) sections, we must
\tableofcontents decrement the secttoc counter (stc) via \mtc@onebacksect when the corresponding
\listoffigures commands are executed. Hence we patch these commands.
\listoftables
\mtcsv@tableofcontents 291   \let\mtcsv@tableofcontents\tableofcontents
\mtcsv@listoffigures 292   \let\mtcsv@listoffigures\listoffigures
\mtcsv@listoftables 293   \let\mtcsv@listoftables\listoftables
294   \def\tableofcontents{\mtcsv@tableofcontents\mtc@onebacksect}
295   \def\listoffigures{\mtcsv@listoffigures\mtc@onebacksect}
296   \def\listoftables{\mtcsv@listoftables\mtc@onebacksect}

```

9.17 Patching the \section command

```

\mtc@svsection If the \section command is not defined (by the document class, usually), we can-
\mtc@svss not patch it and a warning is displayed. Else, we patch its two branches, \@sect
\@ssect (for the unstarred version) or \@ssect (for the starred version, \section*): we add
\@sect \stepcounter{stc} to increment the secttoc counter stc, only in the unstarred case
\section (the version #25 has removed a spurious decrementation of this counter).
\stepcounter
297   \let\mtc@svsection\section
298   \def\section{\stepcounter{stc}\mtc@svsection}
299   \let\mtc@svss\@ssect

```

9.18 Adding an entry in the TOC for a starred section

```

\mtcaddsection To add an entry in the TOC for a starred section, we need the \mtcaddsection macro,
\mtc@ifmtarg which has an optional argument, the title of the section as it should appear in the
\contentsline TOC. By default, this argument is empty. If it is empty (tested via \mtc@ifmtarg) or
\adjuststc omitted, we add a \contentsline{xsect}{}... line in the .toc file. If it is not
\l@xsect empty, we add a \contentsline{section}{title...}... line in the .toc file.
\l@xsection We always add a \contentsline{xsect}{}... line in the .lof and .lot files.
\l@section Then we increment the stc counter, via \adjuststc (this command is defined in
section 9.54 on page 313). Using xsect as first argument of \contentsline means
that \l@xsect will be invoked in place of \l@section to print the entry in the TOC,

```

but `\l@xsect` uses a huge depth (10 000) for this entry, hence it will never be really printed (except if you cheat).

```

300 \newcommand{\mtcaddsection}[1][1]{%
301   \mtc@ifmtarg{#1}{\addcontentsline{toc}{xsect}{}}{%
302     {\addcontentsline{toc}{section}{#1}}}%
303   \addcontentsline{lof}{xsect}{}%
304   \addcontentsline{lot}{xsect}{}%
305   \adjuststc}

```

This code terminates (temporarily) the section level commands, and we continue with chapter level macros.

```

306   }}%
307   {%

```

9.19 Chapter level macros

`\chapter` 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}`.

```

308 \mtcPackageInfo[<I0003>]{minitoc}{chapter level macros available\@gobble}

```

9.20 Patching the `\chapter` command

`\chapter` 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`.
`\mtc@svchapter` Only the unstarred branch (`\@chapter`) is patched here. The other branch is patched later (section 9.36 on page 272).
`\stepcounter`

```

309 \let\mtc@svchapter\@chapter
310 \def\@chapter{\stepcounter{mtc}\mtc@svchapter}

```

9.21 Adding an entry in the TOC for a starred chapter

`\mtcaddchapter` 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 it should appear in the TOC. By default, this argument is empty. If it is empty (tested via `\mtc@ifmtarg`)
`\mtc@ifmtarg`
`\contentsline`
`\adjustmtc`
`\l@xchapter`
`\l@chapter`

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 260). 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 (10 000) for this entry, hence it will never be really printed (except if you cheat).

```

311 \newcommand{\mtcaddchapter}[1][\relax]{%
312     \mtc@ifmtarg{#1}{\addcontentsline{toc}{xchapter}{}}{%
313         \addcontentsline{toc}{chapter}{#1}}%
314     \addcontentsline{lof}{xchapter}{}%
315     \addcontentsline{lot}{xchapter}{}%
316     \adjustmtc}

```

This code terminates (temporarily) the chapter level commands, i.e., terminates the `\ifundefined{chapter}` at the beginning of section 9.15 on page 248.

```

317 }%

```

9.22 Miscellaneous declarations

<pre> \newread \mtc@toks \mtc@string \mtc@strut \mtc@strutbox \mtc@hstrut \mtc@hstrutbox \mtc@v \mtc@zrule \mtc@BBR </pre>	<p>The <code>\newread</code> command must be redeclared as being <code>\outer</code> (as Donald ARSENEAU told me). We need a token register (<code>\mtc@toks</code>), a temporary string (<code>\mtc@string</code>), struts (two kinds, each one using a box containing an invisible vertical rule), a rule with all dimensions equal to zero (<code>\mtc@zrule</code>) and a command discouraging page breaks (<code>\mtc@BBR</code>, for “bad break”). For the struts, which are boxes containing an invisible vertical rule, we use “ex” units, to follow the current font.</p> <pre> 318 \def\newread{\alloc@6\read\chardef\sixt@@n} 319 \newtoks\mtc@toks 320 \def\mtc@string{\relax} 321 \newbox\mtc@strutbox 322 \setbox\mtc@strutbox=\hbox{\rule[1.8ex]{\z@}{2.5ex}} 323 \def\mtc@strut{\relax\ifmmode\copy\mtc@strutbox 324 \else\unhcopy\mtc@strutbox\fi} 325 \newbox\mtc@hstrutbox 326 \setbox\mtc@hstrutbox=\hbox{\rule[1.ex]{\z@}{1.ex}} 327 \def\mtc@hstrut{\relax\ifmmode\copy\mtc@hstrutbox 328 \else\unhcopy\mtc@hstrutbox\fi} 329 \def\mtc@v{\leavevmode\mtc@strut} 330 \def\mtc@zrule{\rule[\z@]{\z@}{\z@}} 331 \def\mtc@BBR{\unpenalty\nopagebreak[4]} </pre>
--	---

Phase (time runs from left to right):		1	2	3
OS with long extensions	<code>\jobname.mtc1</code>	TRUE	TRUE	*
	<code>\jobname.mtc</code>		FALSE	
OS with short extensions	<code>\jobname.mtc(1)</code>	TRUE	FALSE	*

Figure 9.1: Hack to detect the limitation to short extensions.

9.23 Autoconfiguration of extensions

`\tf@mtc` This code is a hack to determine if the operating system is able or unable to use long extensions (> 3 characters) in file names. We define a file descriptor (`\tf@mtc`) to write files⁴. This code is verbose if long extensions cannot be used, else the messages are only written in the `.log` file. The sequencing of these operations is vital. The figure 9.1 shows this sequence. A star (*) denotes which file is read in phase 3.

```
\if@longextensions@ (0) First, a message and a new flag:
\tf@mtc
\openout 332 \mtcPackageInfo[<I0002>]{minitoc}%
\write 333 {Autoconfiguration of extensions\@gobble}
\closeout 334 \newif\if@longextensions@\@longextensions@false
\input (1) We write "\@longextensions@true" in \jobname.mtc1. But if the OS has short
\jobname extensions, the real name of the file will be truncated to \jobname.mtc.

335 \immediate\openout\tf@mtc \jobname.mtc1
336 \immediate\write\tf@mtc{\string\@longextensions@true}
337 \immediate\closeout\tf@mtc

(2) We write "\@longextensions@false" in \jobname.mtc.

338 \immediate\openout\tf@mtc \jobname.mtc
339 \immediate\write\tf@mtc{\string\@longextensions@false}
340 \immediate\closeout\tf@mtc

(3) We read \jobname.mtc1. But if the OS has short extensions, the real name of the
file will be truncated to \jobname.mtc.

341 \input{\jobname.mtc1}
```

⁴ Note that 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 `.log` file. In fact, minitoc writes also in the `.toc`, `.lof` and `.lot` files, but via file descriptors already used by standard commands like `\tableofcontents`, `\listoffigures` and `\listoftables`. 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 (TEX offers only 16 file descriptors for writing), because they called the standard internal `\starttoc` macro, which invokes `\newwrite`, for each mini-table. As minitoc writes into only one file at a time (and in the `.log` file, and in the standard contents files, of course), we can reuse the same file descriptor and avoid this serious problem. The minitoc package writes in the contents files when it encounters a major sectioning command (`\part`, `\chapter`, or `\section`), if necessary. It writes into the mini-table auxiliary files only via the mini-table preparing commands (`\doparttoc`, ..., `\dosectlot`), once at a time. You do not need a new hammer for each nail.

- (4) Hence, the flag is true if we read really from `\jobname.mtc1`, but false if we read from `\jobname.mtc`. The text and the severity of the messages are different.

```

342 \if@longextensions@
343 \mtcPackageInfo[<I0012>]{minitoc}%
344 {Long extensions (Unix-like) will be used@gobble}
345 \mtcPackageInfo[<I0031>]{minitoc}%
346 {==> this version is configured for UNIX-like \MessageBreak
347 \space\space\space\space(long extensions) file names@gobble}%
348 \else
349 \mtcPackageWarningNoLine[<W0019>]{minitoc}%
350 {Short extensions (MSDOS-like) will be used\MessageBreak
351 ==> this version is configured for MSDOS-like \MessageBreak
352 \space\space\space\space(8+3) file names}
353 \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 245).

```

354 \immediate\openout\tf@mtc \jobname.mtc
355 \immediate\closeout\tf@mtc
356 \immediate\openout\tf@mtc \jobname.mtc1
357 \immediate\closeout\tf@mtc

```

9.24 Detecting obsolete versions of L^AT_EX

`\@inputcheck` This code detects old versions of the L^AT_EX kernel that are no more supported and with which the minitoc package can hardly work. The trick is to detect the absence of some internal L^AT_EX commands, `\@inputcheck` and `\reset@font`. If you get one of these messages, you are in bad luck and should *urgently* update your L^AT_EX installation, which is rusting since... many years!



```

358 \@ifundefined{@inputcheck}%
359 {\mtcPackageWarningNoLine[<W0021>]{minitoc}%
360 {Your version of latex.tex is obsolete.\MessageBreak
361 Trying to continue..}\newread\@inputcheck\relax}{}}
362 \@ifundefined{reset@font}%
363 {\mtcPackageWarningNoLine[<W0022>]{minitoc}%
364 {Your version of latex.tex is very obsolete.\MessageBreak
365 Trying to continue... crossing fingers}%
366 \let\reset@font\relax}{}

```

9.25 A macro to make 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.

```

\ifundottedmtc
\undottedmtcfalse

```

```

367 \newif\ifundottedmtc\undottedmtcfalse
368 \def\@undottedtocline#1#2#3#4#5{%
369   \ifnum #1>\c@tocdepth\relax \else
370     \vskip \z@ plus.2\p@
371     {\leftskip #2\relax \rightskip \@tocrmarg \parfillskip -\rightskip
372      \parindent #2\relax\@afterindenttrue
373      \interlinepenalty\@M
374      \leavevmode
375      \@tempdima #3\relax \advance\leftskip \@tempdima \hbox{}}%
376      \hskip -\leftskip
377      #4\nobreak\hfill \nobreak
378      \null\par}%
379 \fi}

```

9.26 Default values for the page-number customization macros

`\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 [213, 214] is loaded, it does the job. So, we test first `\if@mtc@memoirLoaded@` to use the commands of memoir when they are available.

```

380 \if@mtc@memoirLoaded@

```

```

\mtcpagenumbers For entries in minitocs:
\nomtcpagenumbers
381 \def\mtcpagenumbers{%
382   \cftpagenumberon{section}
383   \cftpagenumberon{subsection}
384   \cftpagenumberon{subsubsection}
385   \cftpagenumberon{paragraph}
386   \cftpagenumberon{subparagraph}}
387 \def\nomtcpagenumbers{%
388   \cftpagenumbersoff{section}
389   \cftpagenumbersoff{subsection}
390   \cftpagenumbersoff{subsubsection}
391   \cftpagenumbersoff{paragraph}
392   \cftpagenumbersoff{subparagraph}}

```

```

\stcpagenumbers For entries in secttocs:
\nostcpagenumbers
393 \def\stcpagenumbers{%
394   \cftpagenumberon{subsection}
395   \cftpagenumberon{subsubsection}
396   \cftpagenumberon{paragraph}

```

```

397 \cftpagenumberon{subparagraph}}
398 \def\nostcpagenumbers{%
399 \cftpagenumbersoff{subsection}
400 \cftpagenumbersoff{subsubsection}
401 \cftpagenumbersoff{paragraph}
402 \cftpagenumbersoff{subparagraph}}

```

\ptcpagenumbers For entries in parttocs:
\noptcpagenumbers

```

403 \def\ptcpagenumbers{%
404 \cftpagenumberon{chapter}
405 \cftpagenumberon{section}
406 \cftpagenumberon{subsection}
407 \cftpagenumberon{subsubsection}
408 \cftpagenumberon{paragraph}
409 \cftpagenumberon{subparagraph}}
410 \def\noptcpagenumbers{%
411 \cftpagenumbersoff{chapter}
412 \cftpagenumbersoff{section}
413 \cftpagenumbersoff{subsection}
414 \cftpagenumbersoff{subsubsection}
415 \cftpagenumbersoff{paragraph}
416 \cftpagenumbersoff{subparagraph}}

```

\mlfpagenumbers For entries in minilofs, sectlofs, and partlofs:
\nomlfpagenumbers

```

\slfpagenumbers 417 \def\mlfpagenumbers{\cftpagenumberon{figure}}
\noslfpagenumbers 418 \def\nomlfpagenumbers{\cftpagenumbersoff{figure}}
\plfpagenumbers 419 \def\slfpagenumbers{\cftpagenumberon{figure}}
\noplfpagenumbers 420 \def\noslfpagenumbers{\cftpagenumbersoff{figure}}
421 \def\plfpagenumbers{\cftpagenumberon{figure}}
422 \def\noplfpagenumbers{\cftpagenumbersoff{figure}}

```

\mltpagenumbers For entries in minilots, sectlots, and partlots:
\nomltpagenumbers

```

\sltpagenumbers 423 \def\mltpagenumbers{\cftpagenumberon{table}}
\nosltpagenumbers 424 \def\nomltpagenumbers{\cftpagenumbersoff{table}}
\pltpagenumbers 425 \def\sltpagenumbers{\cftpagenumberon{table}}
\nopltpagenumbers 426 \def\nosltpagenumbers{\cftpagenumbersoff{table}}
427 \def\pltpagenumbers{\cftpagenumberon{table}}
428 \def\nopltpagenumbers{\cftpagenumbersoff{table}}

```

Else, minitoc will use its own commands.

```

429 \else

```

```

\mtcpagenumbers First, for minitocs, secttocs and parttocs:
\nomtcpagenumbers
\mlfpagenumbers 430 \def\mtcpagenumbers{\let\mtc@pgno\null}
\nomlfpagenumbers 431 \def\nomtcpagenumbers{\let\mtc@pgno\relax}
\mltpagenumbers 432 \def\stcpagenumbers{\let\stc@pgno\null}
\nomltpagenumbers 433 \def\nostcpagenumbers{\let\stc@pgno\relax}
434 \def\ptcpagenumbers{\let\ptc@pgno\null}
435 \def\noptcpagenumbers{\let\ptc@pgno\relax}

```

```

\mlfpagenumbers Then, for minilofs, sectlofs and partlofs:
\nomlfpagenumbers
\mlfpagenumbers 436 \def\mlfpagenumbers{\let\mlf@pgno\null}
\nomlfpagenumbers 437 \def\nomlfpagenumbers{\let\mlf@pgno\relax}
\mltpagenumbers 438 \def\slfpagenumbers{\let\slf@pgno\null}
\nomltpagenumbers 439 \def\noslfpagenumbers{\let\slf@pgno\relax}
440 \def\plfpagenumbers{\let\plf@pgno\null}
441 \def\noplfpagenumbers{\let\plf@pgno\relax}

```

```

\mltpagenumbers Then, for minilots, sectlots and partlots:
\nomltpagenumbers
\mltpagenumbers 442 \def\mltpagenumbers{\let\mlt@pgno\null}
\nomltpagenumbers 443 \def\nomltpagenumbers{\let\mlt@pgno\relax}
\mltpagenumbers 444 \def\sltpagenumbers{\let\slt@pgno\null}
\nomltpagenumbers 445 \def\nosltpagenumbers{\let\slt@pgno\relax}
446 \def\pltpagenumbers{\let\plt@pgno\null}
447 \def\nopltpagenumbers{\let\plt@pgno\relax}
448 \fi

```

```

\ptcpagenumbers Then the default values are set; page numbers are present:
\plfpagenumbers
\pltpagenumbers 449 \ptcpagenumbers
\mtcpagenumbers 450 \plfpagenumbers
\mlfpagenumbers 451 \pltpagenumbers
\mltpagenumbers 452 \mtcpagenumbers
\stcpagenumbers 453 \mlfpagenumbers
\slfpagenumbers 454 \mltpagenumbers
\sltpagenumbers 455 \stcpagenumbers
456 \slfpagenumbers
457 \sltpagenumbers

```

9.27 “Features” for the mini-tables

Each kind of mini-table has three “features”: a “before” feature, an “after” feature and a “thispagestyle” 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 contains 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.

A “thispagestyle” 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 “thispagestyle” 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 “thispagestyle” 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   If \chapter is not defined, the part level mini-tables have no “before” feature (by
\beforeparttoc default):
\beforepartlof
\beforepartlot 458 \@ifundefined{chapter}{%
459   \let\beforeparttoc\empty
460   \let\beforepartlof\empty
461   \let\beforepartlot\empty}%

```

```

\cleardoublepage But if \chapter is defined, they have a \cleardoublepage as default “before” feature:

462   {\let\beforeparttoc\cleardoublepage
463    \let\beforepartlof\cleardoublepage
464    \let\beforepartlot\cleardoublepage}

```

```

\beforeinitoc Chapter level mini-tables have no “before” feature (by default):
\beforeminilof
\beforeminilot 465 \let\beforeinitoc\empty
466 \let\beforeminilof\empty
467 \let\beforeminilot\empty

```

```

\beforesecttoc Section level mini-tables have no “before” feature (by default):
\beforesectlof
\beforesectlot 468 \let\beforesecttoc\empty
469 \let\beforesectlof\empty
470 \let\beforesectlot\empty

```

`\chapter` If `\chapter` is not defined, the part level mini-tables have no “after” feature (by default):

```

\afterparttoc
\afterpartlof 471 \@ifundefined{chapter}{%
\afterpartlot 472   \let\afterparttoc\empty
               473   \let\afterpartlof\empty
               474   \let\afterpartlot\empty}%

```

`\cleardoublepage` But if `\chapter` is defined, they have a `\cleardoublepage` as default “after” feature:

```

475   {\let\afterparttoc\cleardoublepage
476    \let\afterpartlof\cleardoublepage
477    \let\afterpartlot\cleardoublepage}

```

`\afterminitoc` Chapter level mini-tables have no “after” feature (by default):

```

\afterminilof
\afterminilot 478 \let\afterminitoc\empty
               479 \let\afterminilof\empty
               480 \let\afterminilot\empty

```

`\aftersecttoc` Section level mini-tables have no “after” feature (by default):

```

\aftersectlof
\aftersectlot 481 \let\aftersecttoc\empty
               482 \let\aftersectlof\empty
               483 \let\aftersectlot\empty

```

`\thispagestyle` By default, all the “thispagestyle” 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 “thispagestyle” features at the part level.

```

\thispageparttocstyle
\thispagepartlofstyle
\thispagepartlotstyle
\thispageminitocstyle 484 \@ifundefined{chapter}{%
\thispageminilofstyle 485   \def\thispageparttocstyle{\empty}
\thispageminilotstyle 486   \def\thispagepartlofstyle{\empty}
\thispagesecttocstyle 487   \def\thispagepartlotstyle{\empty}}%
\thispagesectlofstyle 488 {\def\thispageparttocstyle{\thispagestyle{empty}}
\thispagesectlotstyle 489   \def\thispagepartlofstyle{\thispagestyle{empty}}
                     490   \def\thispagepartlotstyle{\thispagestyle{empty}}}
                     491   \def\thispageminitocstyle{\empty}
                     492   \def\thispageminilofstyle{\empty}
                     493   \def\thispageminilotstyle{\empty}
                     494   \def\thispagesecttocstyle{\empty}
                     495   \def\thispagesectlofstyle{\empty}
                     496   \def\thispagesectlotstyle{\empty}

```

`\mtcsetfeature` In section 9.66.8 on page 353, we will define the `\mtcsetfeature` macro which is much easier user interface to set the mini-tables “features”.

9.28 Fake tables of contents

`\faketableofcontents` 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 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 ⁵.

```

497 \def\faketableofcontents{\fake@starttoc{toc}%
498   \ifundefined{c@ptc}{\setcounter{ptc}{0}}%
499   \ifundefined{c@mtc}{\setcounter{mtc}{0}}%
500   \ifundefined{c@stc}{\setcounter{stc}{0}}%
501   }
502 \def\fakelistoffigures{\fake@starttoc{lof}}
503 \def\fakelistoftables{\fake@starttoc{lot}}
504 \def\fake@starttoc#1{\begingroup
505   \makeatletter
506   \if@filesw \expandafter\newwrite\csname tf@#1\endcsname
507             \immediate\openout \csname tf@#1\endcsname
508             \jobname.#1\relax
509   \fi
510   \global\@nobreakfalse \endgroup}

```

This code uses the same file descriptors (for writing) than the original commands.

9.29 Depth counters for minilofs and minilots

`\AtBeginDocument` 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:

```

511 \AtBeginDocument{%
512   \ifundefined{c@lofdepth}{%
513     {\newcounter{minilofdepth}\setcounter{minilofdepth}{2}}%
514   \ifundefined{c@lotdepth}{%
515     {\newcounter{minilotdepth}\setcounter{minilotdepth}{2}}%
516   }

```

⁵ Remember the infamous “`stc0`” bug.

9.30 Chapter level commands

From here, we define the chapter-level commands.

```
\mtc@markboth First, we memorize the marks (not used today, but...):
\@mkboth
517 \global\let\mtc@markboth\markboth
518 \global\let\@mkboth\markboth
```

9.31 Starred parts, chapters or sections

```
\addst@rred We define commands to manage the starred sectionning commands: \part*, \chapter*
\addcontentsline and \section*. The section-level is different depending if \chapter is defined or not.
\stepcounter Eventually, a counter is incremented. A contents line is added in the .toc file, with the
\c@ptc right depth to print it (see \l@star... later, in section 9.64 on page 335).
\c@mtc
\c@stc 519 \def\addst@rred#1#2{%
520 \addcontentsline{toc}{star#1}{#2}%
521 \@ifundefined{c@ptc}{}{%
522 \expandafter\ifx\csname #1\endcsname\part\relax
523 \stepcounter{ptc}%
524 \fi
525 }%
526 \@ifundefined{c@mtc}{}{%
527 \expandafter\ifx\csname #1\endcsname\chapter\relax
528 \stepcounter{mtc}%
529 \fi
530 }%
531 \@ifundefined{c@stc}{}{%
532 \expandafter\ifx\csname #1\endcsname\section\relax
533 % \ifundefined{chapter}{\stepcounter{stc}}{}%
534 \stepcounter{stc}%
535 \fi
536 }%
537 }%

\addstarredsection If \chapter is not defined, we just define \addstarredsection:
\chapter
\addst@rred 538 \@ifundefined{chapter}{%
539 \gdef\addstarredsection#1{\addst@rred{section}{#1}}
540 }%
```

Else we begin to define the stuff for chapter-level commands (the “else” branch of `\@ifundefined{chapter}`):

541 {%

```

\The@mtc We define now: the internal format of the mtc counter (\The@mtc), the obsolete com-
\firstchapteris mand \firstchapteris (it just emits a harmless warning), the mtc counter (initial-
\if@firstchapteris@used@ ized to 0), the \adjustmtc command (increments the mtc counter, by 1 by default),
\adjustmtc the \decrementmtc command (decrements the mtc counter by 1), the \incrementmtc
\decrementmtc command (increments the mtc counter by 1), the format of the mtc counter (\themtc),
\incrementmtc the counter minitocdepth, initialized to 2, for the depth of a minitoc (analog to the
\themtc standard tocdepth counter). We define the horizontal rules to draw before and after
\columnwidth minitocs (\mtc@rule), and we copy that definition into analog macros for other kinds
\mtc@rule of mini-tables. We also set the default value (24pt) of \mtcindent, the indentation for
\mlf@rule minitocs (both sides). The rules are 0.4pt thick. They are defined via \hrule to stay in
\mlt@rule vertical mode for the final \kern.
\plf@rule
\plt@rule 542 \def\The@mtc{\arabic{mtc}}
\slf@rule 543 \def\firstchapteris#1%
\slt@rule 544 {\mtcPackageWarning[<W0003>]{minitoc}%
\mtcindent 545 {\string\firstchapteris \space is an obsolete\MessageBreak command}%
546 \@firstchapteris@used@true}
547 \newcounter{mtc}
548 \setcounter{mtc}{0}
549 \newcommand{\adjustmtc}[1][1]{\addtocounter{mtc}{#1}}
550 \def\decrementmtc{\addtocounter{mtc}{-1}}
551 \def\incrementmtc{\addtocounter{mtc}{+1}}
552 \gdef\themtc{\arabic{mtc}}
553 \newcounter{minitocdepth}
554 \setcounter{minitocdepth}{2}
555 \def\mtc@rule{\kern-3\p@ \hrule \@width\columnwidth \kern2.6\p@}
556 \let\mlf@rule\mtc@rule
557 \let\mlt@rule\mtc@rule
558 \let\plf@rule\mtc@rule
559 \let\plt@rule\mtc@rule
560 \let\slf@rule\mtc@rule
561 \let\slt@rule\mtc@rule
562 \mtcindent=24\p@

```

9.32 Font commands for the mini-tables

```

\mtcfont We define these commands with full NFSS [141] descriptions. These definitions are
\mtcSfont effective if \chapter is defined. The fonts for titles are also defined here. See also the
\mtcSSfont \mtcsetfont macro (section 9.66.2 on page 338) and the \mtcsettitlefont macro
\mtcSSSfont later (section 9.66.3 on page 342).
\mtcPfont
\mtcSPfont 563 \def\mtcfont{\small\rmfamily\upshape\mdseries}
\mlffont 564 \def\mtcSfont{\small\rmfamily\upshape\bfseries}
\mlfSfont 565 \let\mtcSSfont\mtcfont
\mltfont 566 \let\mtcSSSfont\mtcfont
\mltSfont
\mtifont

```

```

567 \let\mtcPfont\mtcfont
568 \let\mtcSPfont\mtcfont
569 \let\mlffont\mtcfont
570 \let\mlfSfont\mtcfont
571 \let\mltfont\mtcfont
572 \let\mltSfont\mtcfont
573 \def\mtifont{\large\rmfamily\upshape\bfseries}

```

`\coffeeont` And `\coffeeont` is used for “coffee breaks ☕” in the `minutes` package [145].

```

574 \def\coffeeont{\small\rmfamily\slshape\mdseries}

```

9.33 Internal commands to position the mini-table titles

`\df@mtitc` 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 next page). An empty title needs a vertical correction (Frank MITTELBACH).

```

\c@mti Centering, flushleft, flushright or empty titles:
\l@mti
\r@mti 575 \def\c@mti#1{\null\hfill #1\hfill\hfill\hfill\hfill\hfill\hfill}
\e@mti 576 \def\l@mti#1{\null #1\hfill\hfill\hfill\hfill\hfill\hfill}
\n@mti 577 \def\r@mti#1{\null\hfill #1\hfill\hfill\hfill\hfill\hfill}
578 \def\e@mti#1{\vspace{-\baselineskip}}
579 \def\n@mti#1{\vspace{-\baselineskip}}

```

```

\l@mti Default: titles on left:
\do@mtitc
\df@mtitc 580 \let\do@mtitc\l@mti
\do@mtilf 581 \let\df@mtitc\l@mti
\df@mtilf 582 \let\do@mtilf\l@mti
\do@mtilt 583 \let\df@mtilf\l@mti
\df@mtilt 584 \let\do@mtilt\l@mti
585 \let\df@mtilt\l@mti

```

9.34 The `mtc@verse` environment

`mtc@verse` Each minitoc is placed inside a `mtc@verse` environment. This environment is analog to the standard `verse` environment and hence defined via two commands: `\mtc@verse` and `\endmtc@verse`. As it is a list environment, we first define (in a local way) `\`, then call `\list{}` and set some dimensions like `\itemsep`, `\itemindent`, `\listparindent`, `\topsep`. `\parsep` is set to zero if the `tight` option is active (to reduce the spacing of the lines). `\parskip` is set to zero if the `k-tight` option is active (to reduce the spacing of the lines). Both margins are set to `\mtcindent`. `\endmtc@verse` terminates the list and discourages a page break.

```

\iftightmtc
\ifktightmtc
  \
  \list
  \itemsep
  \itemindent
\listparindent
  \topsep
  \parsep
\mtcindent
586 \def\mtc@verse{\let\=\@centercr
587 \list{ }\itemsep\z@
588 \itemindent \z@
589 \listparindent \itemindent
590 \partopsep\z@
591 \iftightmtc \parsep\z@ \fi
592 \ifktightmtc \parskip\z@ \fi
593 \topsep=1ex
594 \leftmargin\mtcindent
595 \rightmargin\leftmargin}\item[]}
596 \def\endmtc@verse{\nopagebreak[4]\endlist}

```

9.35 The `\minitoc`, `\minilof`, and `\minilot` commands

These three commands are very similar, with only cosmetic differences.

9.35.1 The `\minitoc` command

`\minitoc` The `\minitoc` command must be used after `\chapter` if you need a minitoc (no `\chapter` automatic minitoc).

`\dominitoc` This command accepts an optional argument, whose default value has eventually been set earlier by a `\dominitoc` command. The letter “d” represents this default value. `\dominitoc` has itself an optional argument which sets the default value of the optional argument of `\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` So we define `\minitoc` with an optional argument and its (current) default value, and
`\minitoc@` call the true code in the `\minitoc@` macro (which has one delimited argument); we use
`\@ifnextchar` the `\@ifnextchar` trick to detect a left bracket for the optional argument:

```
597 \def\minitoc{\@ifnextchar[{ \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.

`\if@minitoc@used@` 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 9.78.2.2 on page 378).

```
598 \def\minitoc@[#1]{%
599 \global\@minitoc@used@true
```

`\@tocfile` The name of the file containing the minitoc is constructed from `\jobname` and a suf-
`\if@longextensions` fix `\@tocfile`, which is `mtc` (long extensions) or `M` (short extensions) followed by the absolute number of the minitoc.

```
600 \if@longextensions@
601 \def\@tocfile{mtc\The@mtc}%
602 \else
603 \def\@tocfile{M\The@mtc}%
604 \fi
```

`\mtc@CkFile` Then we test (via `\mtc@CkFile`) the emptiness of this file. A warning is given if the file
`\if@mtc@FE` is empty and a flag is set (a hint will signal that an empty minitoc has been requested).
`\if@mtc@empty@minitoc@`

```
605 \mtc@CkFile{\jobname.\@tocfile}
606 \if@mtc@FE
607 \mtcPackageInfo[<I0006>]{minitoc}%
608 {\jobname.\@tocfile\space is empty}
609 \@mtc@empty@minitoc@true
610 \else
```

`\thispageminicstyle` We call `\thispageminicstyle` 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.

```
611 \thispageminicstyle
612 %% \mtc@markboth{\MakeUppercase{\mtctitle}}{\MakeUppercase{\mtctitle}}%
```



```

\beforeminitoc We call \beforeminitoc, then begin a samepage environment (to try to discourage page
samepage breaks in a minitoc) and look at the position of the title. If the title is empty, the layout
\do@mtitc is corrected. We print the title with its font (\mtifont), then the top rule of the minitoc
\e@mti (if rules are present), using a tabular environment (to inhibit a page break between the
\n@mti title and the top rule). The font is set to \mtcfont.
\c@mti
\l@mti 613 \beforeminitoc
\r@mti 614 \relax\begin{samepage}%
\df@mtic 615 \if #1e\let\do@mtitc\e@mti
\mtc@CkStr 616 \else\if #1n\let\do@mtitc\n@mti
\mtctitle 617 \else\if #1c\let\do@mtitc\c@mti
\if@mtc@FE 618 \else\if #1l\let\do@mtitc\l@mti
\mtcfont 619 \else\if #1r\let\do@mtitc\r@mti
\mtifont 620 \else\if #1d\let\do@mtitc\df@mtitc
\mtc@rule 621 \fi\fi\fi\fi\fi\fi
\columnwidth 622 \mtc@CkStr{\mtctitle}\if@mtc@FE \let\do@mtitc\e@mti\relax\fi
tabular 623 \raggedright
624 \parskip=\z@%
625 \reset@font\mtcfont%
626 \parindent=\z@%
627 \noprogrambreak[4]%
628 \kern-0.8\baselineskip\noprogrambreak[4]%
629 \par\noindent %
630 \ifx\mtc@rule\relax
631 \begin{tabular}{@{}p{\columnwidth}@{}}
632 \reset@font\mtifont\do@mtitc{\mtc@v\mtctitle}\\
633 \end{tabular}%
634 \else
635 \begin{tabular}{@{}p{\columnwidth}@{}}
636 \reset@font\mtifont\do@mtitc{\mtc@v\mtctitle}\\ \hline
637 \end{tabular}%
638 \fi

\mtc@zrule We forbid a page break after the title and the top rule, then set some layout parameters
\mtc@BBR and begin an mtc@verse environment:
\mtcindent
mtc@verse 639 \noprogrambreak[4]\null\leavevmode\mtc@zrule\\\mtc@BBR
640 \leftmargin\mtcindent
641 \rightmargin\mtcindent
642 \itemindent=\z@\labelwidth=\z@%
643 \labelsep=\z@\listparindent=\z@%
644 \begin{mtc@verse}%

\c@tocdepth We force the effective depth of the mini-table (\c@tocdepth) to the required depth
\c@minitocdepth (\c@minitocdepth), so the printing is done inside the mtc@verse environment, where
\\ tocdepth has been forced to minitocdepth, to print only the entries whose level is low
\mtc@BBR enough. then inhibit a page break. The blank line is necessary to avoid a parasite negative
indentation.

```

```

645      \c@tocdepth=\c@minitocdepth
646      \leavevmode\\mtc@BBR\vskip -.5\baselineskip

```

\mtc@pgno We test the presence of leaders and page numbers, then print the minitoc by inputting the minitoc file. But before reading the minitoc file, we must call the hook macro (asked for by Donald ARSENEAU for his notoccite package [8]) \mtc@hook@beforeinputfile and the macro \mtc@setform which adjusts some layout parameters (defined by the user via some \mtcsetformat commands). We work in a group to keep local some macro redefinitions.

```

\mtc@strut
647 \begingroup
648   \makeatletter
649   \ifundefined{mtc@pgno}%
650   {\let\@dottedtocline\@undottedtocline}{}
651   \@filesfalse\mtc@hook@beforeinputfile
652   \mtc@setform%
653   \@input{\jobname.\@tocfile}
654   \vspace{-1ex} \vspace{-.5\baselineskip}
655   \leavevmode\mtc@strut
656   \global\@nobreakfalse\endgroup

```

mtc@verse We close the mtc@verse environment, add the bottomrule (while preventing a page break), then close the samepage environment, and call \afterminitoc. The blank line (\\) is essential.

```

\\
samepage
\afterminitoc
657      \end{mtc@verse}%
658      \kernafterminitoc
659      \nopagebreak[4]\mtc@bottom@rule\null\leavevmode\\%
660      \vskip-1.0\baselineskip\mtc@zrule
661      \end{samepage}%
662      \par\pagebreak[1]\vspace*{-1ex}\afterminitoc\fi}

```

\mtc@bottom@rule And we define the bottom rule for a minitoc, with some space under the minitoc:

```

\mtc@rule
\columnwidth
663 \def\mtc@bottom@rule{%
664   \ifx\mtc@rule\relax\relax\else
665     \vskip -2.5ex
666     \rule[2.4\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}\fi}

```

9.35.2 The \minilof command

\minilof The \minilof command is very similar to the \minitoc command.

`\minilof` The `\minilof` command must be used after `\chapter` if you need a minilof (no `\chapter` automatic minilof).

`\dominilof` This command accepts an optional argument, whose default value has eventually been
`\minilof` 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 “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 `\dominilof`, or locally via the optional argument of `\minilof`.

`\minilof` So we define `\minilof` with an optional argument and its (current) default value, and
`\minilof@` call the true code in the `\minilof@` macro (which has one delimited argument); we use
`\@ifnextchar` the `\@ifnextchar` trick to detect a left bracket for the optional argument:

```
667 \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.

`\if@minilof@used@` 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 9.78.2.2 on page 378).

```
668 \def\minilof@[#1]{%
669 \global\@minilof@used@true
```

`\@tocfile` The name of the file containing the minilof is constructed from `\jobname` and a suf-
`\if@longextensions@` fix `\@tocfile`, which is mlf (long extensions) or F (short extensions) followed by the absolute number of the minilof.

```
670 \if@longextensions@%
671 \def\@tocfile{mlf\The@mtc}%
672 \else
673 \def\@tocfile{F\The@mtc}%
674 \fi
```

`\mtc@CkFile` Then we test (via `\mtc@CkFile`) the emptiness of this file. A warning is given if the file
`\if@mtc@FE` is empty and a flag is set (a hint will signal that an empty minilof has been requested).
`\if@mtc@empty@minilof@`

```
675 \mtc@CkFile{\jobname.\@tocfile}
676 \if@mtc@FE
677 \mtcPackageInfo[<I0006>]{minitoc}%
678 {\jobname.\@tocfile\space is empty}
679 \@mtc@empty@minilof@true
680 \else
```

`\thispageminilofstyle` We call `\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.

```
681      \thispageminilofstyle
682 %%      \mtc@markboth{\MakeUppercase{\mlftitle}}{\MakeUppercase{\mlftitle}}%
```

`\beforeminilof` We call `\beforeminilof`, then begin a `samepage` environment (to try to discourage page breaks in a minilof) 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 minilof (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 `\mlffont`.

```
\l@mti 683      \beforeminilof
\r@mti 684      \relax\begin{samepage}%
\df@mtic 685      \if #1e\let\do@mtilf\e@mti
\mtc@CkStr 686      \else\if #1n\let\do@mtilf\n@mti
\mtctitle 687      \else\if #1c\let\do@mtilf\c@mti
\if@mtc@FE 688      \else\if #1l\let\do@mtilf\l@mti
\mlffont 689      \else\if #1r\let\do@mtilf\r@mti
\mtifont 690      \else\if #1d\let\do@mtilf\df@mtilf
\mlf@rule 691      \fi\fi\fi\fi\fi\fi
\columnwidth 692      \mtc@CkStr{\mlftitle}\if@mtc@FE \let\do@mtilf\e@mti\relax\fi
tabular 693      \raggedright
694      \parskip=\z@
695      \reset@font\mlffont
696      \parindent=\z@
697      \nopagebreak[4]%
698      \kern-0.8\baselineskip\nopagebreak[4]%
699      \par\noindent
700      \ifx\mlf@rule\relax
701      \begin{tabular}{@{}p{\columnwidth}@{}}
702      \reset@font\mtifont\do@mtilf{\mtc@v\mlftitle}\\
703      \end{tabular}%
704      \else
705      \begin{tabular}{@{}p{\columnwidth}@{}}
706      \reset@font\mtifont\do@mtilf{\mtc@v\mlftitle}\\\hline
707      \end{tabular}%
708      \fi
```

`\mtc@zrule` We forbid a page break after the title and the top rule, then set some layout parameters and begin an `mtc@verse` environment:

```
\mtc@BRR
\mtcindent
mtc@verse 709      \nopagebreak[4]\null\leavevmode\mtc@zrule\\\mtc@BRR
710      \leftmargin\mtcindent
711      \rightmargin\mtcindent
712      \itemindent=\z@\labelwidth=\z@%
713      \labelsep=\z@\listparindent=\z@%
714      \begin{mtc@verse}%
```

`\c@lofdepth` We force the effective depth of the mini-table (`\c@tocdepth`) to the required depth
`\c@minilofdepth` (`\c@minilofdepth`), so the printing is done inside the `mtc@verse` environment, where
`\` `tocdepth` has been forced to `minilofdepth`, to print only the entries whose level is low
`\mtc@BBR` enough. then inhibit a page break. The blank line is necessary to avoid a parasite negative
indentation.

```
715 \ifundefined{c@lofdepth}%
716 {}%
717 {\c@lofdepth=\c@minilofdepth
718 \ifnum\c@lofdepth<1\relax\c@lofdepth=1\fi}
719 \leavevmode\\mtc@BBR\vskip -.5\baselineskip
```

`\mtc@pgno` We test the presence of leaders and page numbers, then print the minilof by inputting the
`\@dottedtocline` minilof file. But before reading the minilof file, we must call the hook macro (asked for
`\@undottedtocline` by Donald ARSENEAU for his `notoccite` package [8]) `\mtc@hook@beforeinputfile`
`\mtc@hook@beforeinputfile` and the macro `\mlf@setform` which adjusts some layout parameters (defined by the user
`\mlf@setform` via some `\mtcsetformat` commands). We work in a group to keep local some macro
`\mtcsetformat` redefinitions.
`\mtc@strut`

```
720 \beginngroup
721 \makeatletter
722 \ifundefined{mlf@pgno}%
723 {\let\@dottedtocline\@undottedtocline}{}
724 \@filesfalse\mtc@hook@beforeinputfile
725 \mlf@setform
726 \@input{\jobname.\@tocfile}
727 \vspace{-1ex} \vspace{-.5\baselineskip}
728 \leavevmode\mtc@strut
729 \global\@nbreakfalse\endgroup
```

`mtc@verse` We close the `mtc@verse` environment, add the bottomrule (while preventing a page
`\mtc@bottom@rule` break), then close the `samepage` environment, and call `\afterminilof`. The blank line
`\` (`\`) is essential.

```
samepage
\afterminilof 730 \end{mtc@verse}%
731 \kernafterminilof
732 \nopagebreak[4]\mlf@rule\null\leavevmode\\%
733 \vskip-1.0\baselineskip\mtc@zrule\end{samepage}%
734 \par\pagebreak[1]\vspace*{-1ex}\afterminilof\fi%
```

9.35.3 The `\minilot` command

`\minilot` The `\minilot` command is absolutely similar to the `\minilof` command:

`\minilot` The `\minilot` command must be used after `\chapter` if you need a minilot (no `\chapter` automatic minilot).

`\dominilot` This command accepts an optional argument, whose default value has eventually been
`\minilot` 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 `\dominilot`, or locally via the optional argument of `\minilot`.

`\minilot` So we define `\minilot` with an optional argument and its (current) default value, and
`\minilot@` call the true code in the `\minilot@` macro (which has one delimited argument); we use
`\@ifnextchar` the `\@ifnextchar` trick to detect a left bracket for the optional argument:

```
735 \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.

`\if@minilot@used@` First, we set the global flag `\@minilot@used@true` to note that `\minilot` has been called (this will be used by a hint later, section 9.78.2.2 on page 378).

```
736 \def\minilot@[#1]{%
737 \global\@minilot@used@true
```

`\@tocfile` The name of the file containing the minilot is constructed from `\jobname` and a suf-
`\if@longextensions@` fix `\@tocfile`, which is mlt (long extensions) or T (short extensions) followed by the absolute number of the minilot.

```
738 \if@longextensions@%
739 \def\@tocfile{mlt\The@mtc}%
740 \else
741 \def\@tocfile{T\The@mtc}%
742 \fi
```

`\mtc@CkFile` Then we test (via `\mtc@CkFile`) the emptiness of this file. A warning is given if the file
`\if@mtc@FE` is empty and a flag is set (a hint will signal that an empty minilot has been requested).
`\if@mtc@empty@minilot@`

```
743 \mtc@CkFile{\jobname.\@tocfile}
744 \if@mtc@FE
745 \mtcPackageInfo[<I0006>]{minitoc}%
746 {\jobname.\@tocfile\space is empty}
747 \@mtc@empty@minilot@true
748 \else
```

`\thispageminilotstyle` 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.

```
749      \thispageminilotstyle
750 %%      \mtc@markboth{\MakeUppercase{\mltttitle}}{\MakeUppercase{\mltttitle}}%
```

`\beforeminilot` 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`.

```

\l@mti 751      \beforeminilot
\r@mti 752      \relax\begin{samepage}%
\df@mtic 753      \if #1e\let\do@mtilt\e@mti
\mtc@CkStr 754      \else\if #1n\let\do@mtilt\n@mti
\mtctitle 755      \else\if #1c\let\do@mtilt\c@mti
\if@mtc@FE 756      \else\if #1l\let\do@mtilt\l@mti
\mltfont 757      \else\if #1r\let\do@mtilt\r@mti
\mtifont 758      \else\if #1d\let\do@mtilt\df@mtilt
\mlt@rule 759      \fi\fi\fi\fi\fi\fi
\columnwidth 760      \mtc@CkStr{\mltttitle}\if@mtc@FE \let\do@mtilt\e@mti\relax\fi
tabular 761      \raggedright
762      \parskip=\z@%
763      \reset@font\mltfont%
764      \parindent=\z@%
765      \nopagebreak[4]%
766      \kern-0.8\baselineskip\nopagebreak[4]%
767      \par\noindent
768      \ifx\mlt@rule\relax
769      \begin{tabular}{@{}p{\columnwidth}@{}}
770      \reset@font\mtifont\do@mtilt{\mtc@v\mltttitle}\\
771      \end{tabular}%
772      \else
773      \begin{tabular}{@{}p{\columnwidth}@{}}
774      \reset@font\mtifont\do@mtilt{\mtc@v\mltttitle}\\\hline
775      \end{tabular}%
776      \fi
```

`\mtc@zrule` We forbid a page break after the title and the top rule, then set some layout parameters and begin an `mtc@verse` environment:

```

\mtcindent
mtc@verse 777      \nopagebreak[4]\null\leavevmode\mtc@zrule\\\mtc@BBR
778      \leftmargin\mtcindent
779      \rightmargin\mtcindent
780      \itemindent=\z@\labelwidth=\z@%
781      \labelsep=\z@\listparindent=\z@%
782      \begin{mtc@verse}%
```

`\c@lotdepth` We force the effective depth of the mini-table (`\c@lotdepth`) to the required depth (`\c@minilotdepth`), so the printing is done inside the `mtc@verse` environment, where `lotdepth` has been forced to `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.

```
783     \ifundefined{c@lotdepth}%
784     {}%
785     {\c@lotdepth=\c@minilotdepth
786     \ifnum\c@lotdepth<1\relax\c@lotdepth=1\fi}
787     \leavevmode\\mtc@BBR\vskip -.5\baselineskip
```

`\mtc@pgno` We test the presence of leaders and page numbers, then print the minilot by inputting the `\@dottedtocline` minilot file. But before reading the minilot file, we must call the hook macro (asked for by Donald ARSENEAU for his `notoccite` package [8]) `\mtc@hook@beforeinputfile` and the macro `\mlt@setform` which adjusts some layout parameters (defined by the user via some `\mtcsetformat` commands). We work in a group to keep local some macro redefinitions.

`\mtc@strut`

```
788 \begingroup
789   \makeatletter
790   \@ifundefined{mlt@pgno}%
791   {\let\@dottedtocline\@undottedtocline}{}
792   \@filesfalse\mtc@hook@beforeinputfile
793   \mlt@setform
794   \@input{\jobname.\@tocfile}
795   \vspace{-1ex} \vspace{-.5\baselineskip}
796   \leavevmode\mtc@strut
797   \global\@nobreakfalse\endgroup
```

`mtc@verse` 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 (`\\`) is essential.

```
\\
samepage
\afterminilot 798     \end{mtc@verse}%
799     \kernafterminilot
800     \nopagebreak[4]\mlt@rule\null\leavevmode\\%
801     \vskip-1.0\baselineskip\mtc@zrule\end{samepage}%
802     \par\pagebreak[1]\vspace*{-1ex}\afterminilot\fi}%
```

9.36 Patching the `\chapter` command, continued

`\l@xchapter` First, we define `\l@xchapter` which is like `\l@chapter`, but with a huge depth, to inhibit its printing (except if you cheat):

`\l@chapter`

`\xchapter`


```

803 \def\l@xchapter{\@dottedtocline{\@M}{1em}{2.3em}}
804 \def\xchapter{xchapter}

```

`\@chapter` Then we patch `\@chapter` (the non-starred branch of `\chapter`) to add pseudo-chapter entries in the LOF and the LOT (these entries will be used by the `\dominiXXX` commands to split the LOF and the LOT into slices).

`\sv@chapter`

`\addcontentsline`

`\ignorespaces`

```

805 \let\sv@chapter\@chapter
806 \def\@chapter[#1]#2{\sv@chapter[{#1}]{#2}\relax%
807   \addcontentsline{lof}{xchapter}{#1}%
808   \addcontentsline{lot}{xchapter}{#1}%
809   \ignorespaces}

```

`\mtc@schapter` 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.

`\@schapter`

`\addtocontents`

`\chapterbegin`

`\chapterend`

```

810 \let\mtc@schapter\@schapter
811 \def\@schapter{\addtocontents{toc}{\protect\chapterend}\mtc@schapter}
812 \def\@schapter{\addtocontents{@@@}{\protect\chapterbegin}\mtc@schapter}
813 \let\chapterbegin\relax
814 \let\chapterend\relax

```

9.37 The `\addstarred...` commands

`\addstarredsection` 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

`\addstarredchapter` command `\addstarredchapter`. If the command `\part` is defined, we define the

`\addstarredpart` command `\addstarredpart`. We use the utility command `\addst@rred` defined in

`\chapter` section 9.31 on page 260.

`\section`

`\part`

```

\addst@rred 815 \@ifundefined{chapter}%
816   {\@ifundefined{section}%
817     {}{\def\addstarredsection#1{\addst@rred{section}{#1}}}%
818   {\def\addstarredchapter#1{\addst@rred{chapter}{#1}}}
819 \@ifundefined{part}%
820   {}{\def\addstarredpart#1{\addst@rred{part}{#1}}}

```

9.38 TOC entries without leaders

`\@Undottedtocline` We define two internal macros to format TOC entries without leaders. The macro `\coffeeont` `\@Undottedtocline` prints no page number, but `\@Undottedtoclinep` prints it.

```

821 \def\@Undottedtocline#1#2#3#4#5{%
822   \ifnum #1>\c@tocdepth\relax \else
823     \vskip \z@ \@plus.2\p@
824     {\leftskip #2\relax \rightskip \@tocrmarg \parfillskip -\rightskip
825     \parindent #2\relax\@afterindenttrue
826     \interlinepenalty\@M
827     \leavevmode
828     \@tempdima #3\relax
829     \advance\leftskip \@tempdima \null\nobreak\hskip -\leftskip
830     {\coffeeont #4}\nobreak
831     \nobreak\null
832     \par}%
833   \fi}

```

`\@Undottedtoclinep` The same but with the page number:
`\coffeeont`

```

834 \def\@Undottedtoclinep#1#2#3#4#5{%
835   \ifnum #1>\c@tocdepth\relax \else
836     \vskip \z@ \@plus.2\p@
837     {\leftskip #2\relax \rightskip \@tocrmarg \parfillskip -\rightskip
838     \parindent #2\relax\@afterindenttrue
839     \interlinepenalty\@M
840     \leavevmode
841     \@tempdima #3\relax
842     \advance\leftskip \@tempdima \null\nobreak\hskip -\leftskip
843     {\#4}\nobreak
844     \hfill
845     \nobreak\null
846     \hb@xt@\@pnumwidth{\hfil\normalfont \normalcolor #5}%
847     \par}%
848   \fi}

```

9.39 Mini-tables with or without leaders

`\minitoc@` This code sets the flag to false, then patches each mini-table command (its internal
`\minilof@` part). We alter the commands `\minitoc@`, `\minilof@`, etc., to test the flag
`\minilot@` `\ifundottedmtc` and, if true, replace locally `\@dottedtocline` by its dotless version
`\@dottedtocline` `\@Undottedtoclinep`. Of course, we must also test the availability of the `\chapter`,
`\@Undottedtoclinep` `\part` and `\section` commands, to avoid to define many unnecessary commands.
`\sv@minitoc@`
`\sv@minilof@` 849 `\@ifundefined{chapter}{}{}%`
`\sv@minilot@`

```

850 \let\sv@minitoc@\minitoc@
851 \def\minitoc@[#1]{\ifundottedmtc\let\@dottedtocline\@Undottedtoclinep\fi
852   \sv@minitoc@[#1]}}%
853 \let\sv@minilof@\minilof@
854 \def\minilof@[#1]{\ifundottedmtc\let\@dottedtocline\@Undottedtoclinep\fi
855   \sv@minilof@[#1]}}%
856 \let\sv@minilot@\minilot@
857 \def\minilot@[#1]{\ifundottedmtc\let\@dottedtocline\@Undottedtoclinep\fi
858   \sv@minilot@[#1]}}

```

```

\sv@parttoc@ For the part level:
\sv@partlof@
\sv@partlot@ 859 \@ifundefined{part}{}{}%
\ifundottedmtc 860 \let\sv@parttoc@\parttoc@
\parttoc@ 861 \def\parttoc@[#1]{\ifundottedmtc\let\@dottedtocline\@Undottedtoclinep\fi
\partlof@ 862   \sv@parttoc@[#1]}}%
\partlot@ 863 \let\sv@partlof@\partlof@
864 \def\partlof@[#1]{\ifundottedmtc\let\@dottedtocline\@Undottedtoclinep\fi
865   \sv@partlof@[#1]}}%
866 \let\sv@partlot@\partlot@
867 \def\partlot@[#1]{\ifundottedmtc\let\@dottedtocline\@Undottedtoclinep\fi
868   \sv@partlot@[#1]}}

```

```

\sv@secttoc@ For the section level:
\sv@sectlof@
\sv@sectlot@ 869 \@ifundefined{chapter}{}{}%
\ifundottedmtc 870 \@ifundefined{section}{}{}%
\secttoc@ 871 \let\sv@secttoc@\secttoc@
\sectlof@ 872 \def\secttoc@[#1]{\ifundottedmtc\let\@dottedtocline\@Undottedtoclinep\fi
\sectlot@ 873   \sv@secttoc@[#1]}}%
874 \let\sv@sectlof@\sectlof@
875 \def\sectlof@[#1]{\ifundottedmtc\let\@dottedtocline\@Undottedtoclinep\fi
876   \sv@sectlof@[#1]}}%
877 \let\sv@sectlot@\sectlot@
878 \def\sectlot@[#1]{\ifundottedmtc\let\@dottedtocline\@Undottedtoclinep\fi
879   \sv@sectlot@[#1]}}{}{}

```

9.40 The `\dominitoc` command and its siblings

`\dominitoc` 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<N>` files (resp. the `\jobname.mlf<N>` and `\jobname.mlt<N>` files), using specific lines in the `\starchapter` `\jobname.toc` (resp. `\jobname.lof` and `\jobname.lot`) file. These lines are essentially chapter-level entry commands (like `\contentsline{chapter}...`,

`\contentsline{xchapter}...`, `\contentsline{starchapter}...`, `\chapbegin`) delimiting chapters in the TOC (or in the LOF or the LOT). Analog part-level lines delimit parts, hence also chapters.

`\dominitoc` As `\dominitoc` has an optional argument, whose default value is “1” (left), it calls `\dominitoc@` with a argument delimited by brackets.

`\dominitoc@`

`@@dominitoc`

`\ifdominitoc@used@` The macros are `\dominitoc` (user interface), which calls `\dominitoc@[1]` (or with the optional argument of `\dominitoc`). Then `\dominitoc@[1]` 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 `\MTC@next#1.toc` (where #1 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 9.78.2.2 on page 378. The code is similar for `\dominilof` and `\dominilot`.

`\dominitoc` The `\dominitoc` command extracts information from the .toc file and create the minitocs files, with the adequate extension.

`@dominitoc`

`\MTC@next`

```
880 \def\dominitoc#1{%
881   \makeatletter
882   \setcounter{mtc}{0}
883   \MTC@next#1.toc\relax\}\setcounter{mtc}{0}}
884 \def\dominitoc{\@ifnextchar[\dominitoc@]{\dominitoc@[1]}}
```

`\dominilof` The `\dominilof` command extracts information from the .lof file and create the minilofs files, with the adequate extension.

`@dominilof`

`\MLF@next`

```
885 \def\dominilof#1{%
886   \makeatletter
887   \setcounter{mtc}{0}
888   \MLF@next#1.lof\relax\}\setcounter{mtc}{0}}
889 \def\dominilof{\@ifnextchar[\dominilof@]{\dominilof@[1]}}
```

`\dominilot` The `\dominilot` command extracts information from the .lot file and create the minilots files, with the adequate extension.

`@dominilot`

`\MLT@next`

```
890 \def\dominilot#1{%
891   \makeatletter
892   \setcounter{mtc}{0}
893   \MLT@next#1.lot\relax\}\setcounter{mtc}{0}}
894 \def\dominilot{\@ifnextchar[\dominilot@]{\dominilot@[1]}}
```

```

\if@dominitoc@used@ Some code to flag the use of the command and manage the position of the minitoc title; a
\if@mtc@hints@ hint detects any spurious invocation.
\@mtc@hints@given@true
\df@mtitc 895 \def\dominitoc@[#1]{%
\@mti 896 \if@mtc@hints@
\n@mti 897 \if@dominitoc@used@
\c@mti 898 \mtcPackageInfo[<I0045>]{minitoc(hints)}%
\l@mti 899 {The \string\dominitoc \space command\MessageBreak
\r@mti 900 has been invoked more than once\MessageBreak}
\@dominitoc 901 \global\@mtc@hints@given@true
902 \fi
903 \fi
904 \global\@dominitoc@used@true
905 \if #1e\let\df@mtitc\@mti%
906 \else\if #1n\let\df@mtitc\n@mti%
907 \else\if #1c\let\df@mtitc\c@mti%
908 \else\if #1l\let\df@mtitc\l@mti%
909 \else\if #1r\let\df@mtitc\r@mti%
910 \fi\fi\fi\fi\fi%
911 \@dominitoc}

```

```

\if@dominilof@used@ Some code to flag the use of the command and manage the position of the minilof title; a
\if@mtc@hints@ hint detects any spurious invocation.
\@mtc@hints@given@true
\df@mtilf 912 \def\dominilof@[#1]{%
\@mti 913 \if@mtc@hints@
\n@mti 914 \if@dominilof@used@
\c@mti 915 \mtcPackageInfo[<I0045>]{minitoc(hints)}%
\l@mti 916 {The \string\dominilof \space command\MessageBreak
\r@mti 917 has been invoked more than once\MessageBreak}
\@dominilof 918 \global\@mtc@hints@given@true
919 \fi
920 \fi
921 \global\@dominilof@used@true
922 \if #1e\let\df@mtilf\@mti%
923 \else\if #1n\let\df@mtilf\n@mti%
924 \else\if #1c\let\df@mtilf\c@mti%
925 \else\if #1l\let\df@mtilf\l@mti%
926 \else\if #1r\let\df@mtilf\r@mti%
927 \fi\fi\fi\fi\fi%
928 \@dominilof}

```

```

\if@dominilot@used@ Some code to flag the use of the command and manage the position of the minilot title; a
\if@mtc@hints@ hint detects any spurious invocation.
\@mtc@hints@given@true
\df@mtilt 929 \def\dominilot@[#1]{%
\@mti 930 \if@mtc@hints@
\n@mti 931 \if@dominilot@used@
\c@mti 932 \mtcPackageInfo[<I0045>]{minitoc(hints)}%
\l@mti 933 {The \string\dominilot \space command\MessageBreak
\r@mti
\@dominilot

```

```

934         has been invoked more than once\MessageBreak}
935   \global\@mtc@hints@given@true
936   \fi
937 \fi
938 \global\@dominilot@used@true
939 \if #1e\let\df@mtilt\e@mti%
940 \else\if #1n\let\df@mtilt\n@mti%
941 \else\if #1c\let\df@mtilt\c@mti%
942 \else\if #1l\let\df@mtilt\l@mti%
943 \else\if #1r\let\df@mtilt\r@mti%
944 \fi\fi\fi\fi\fi%
945 \@@dominilot}

```

\@@dominitoc These macros invoke the \@domini... macros to create the mini-table file, then close
 \@@dominilof the file descriptor.

```

\@@dominilot
  \tf@mtc 946 \def\@@dominitoc{\@dominitoc{\jobname}\immediate\closeout\tf@mtc}
          947 \def\@@dominilof{\@dominilof{\jobname}\immediate\closeout\tf@mtc}
          948 \def\@@dominilot{\@dominilot{\jobname}\immediate\closeout\tf@mtc}

```

9.40.1 Analysis and splitting of the TOC file

This is done via a loop managed by the following macros ⁶:

```

\MTC@next Processes the next entry in the list and removes it from the head of the list:
\MTC@list
\MTC@loop 949 \def\MTC@next#1\relax#2\{\%
          950   \edef\MTC@list{#2}%
          951   \MTC@loop{#1}%
          952 }

\MTC@toc Check if the list is empty:
\MTC@list
\MTC@explist 953 \def\MTC@toc{%
          954   \ifx\MTC@list@empty\else\expandafter\MTC@explist\fi
          955 }

```

⁶ This code is derived from the `xr` package [79], by David CARLISLE, with his permission. The strings “HO”, “DV”, and “BJ” in the comments denote modifications made by Heiko OBERDIEK, Didier VERNA, and Bernd JAEHNE for the support of `hyperref`, essentially by adding an argument to some macros, to use the hyperlink argument in the contents lines.

```

\MTC@contentsline The macro \MTC@contentsline analyses the lines read from the TOC file and detects
\arabic interesting keywords. If \chapter is found, the mtc counter (which simulates the chapter
\chapter counter, but is absolute) is incremented and a new minitoc file is created.
\themtc
\tf@mtc 956 \def\MTC@contentsline#1#2#3#4{% %%H0/BJ: 4 instead of 3 parameters
957 \gdef\themtc{\arabic{mtc}}% %%H0: space removed
958 \expandafter\ifx\csname #1\endcsname\chapter
959 \stepcounter{mtc}%

\if@longextensions@ We test if long or short extensions are used, to build the name of the mini-table file, then
\themtc open it (after closing the file descriptor):
\mtcname
\tf@mtc 960 \if@longextensions@%
\closeout 961 \mtcPackageInfo[<I0033>]{minitoc}%
\openout 962 {Writing\space\jobname.mtc\themtc\@gobble}%
963 \def\mtcname{\jobname.mtc\themtc}%
964 \else
965 \mtcPackageInfo[<I0033>]{minitoc}%
966 {Writing\space\jobname.M\themtc\@gobble}%
967 \def\mtcname{\jobname.M\themtc}%
968 \fi
969 \immediate\closeout\tf@mtc
970 \immediate\openout\tf@mtc=\mtcname
971 \fi

\mtc@toks Now, we filter the relevant contents lines, the token register \mtc@toks is used as a
verbatim memory.

972 \mtc@toks{\noexpand\leavevmode #2}%

\MTC@WriteContentsline Each interesting contents line is copied, with a font command added before it. We begin
\section with the standard sectionning commands, below \chapter:
\subsection
\subsubsection 973 \expandafter\ifx\csname #1\endcsname\section
\paragraph 974 \MTC@WriteContentsline{#1}{mtcS}{#3}{#4}%
\subparagraph 975 \fi
976 \expandafter\ifx\csname #1\endcsname\subsection
977 \MTC@WriteContentsline{#1}{mtcSS}{#3}{#4}%
978 \fi
979 \expandafter\ifx\csname #1\endcsname\subsubsection
980 \MTC@WriteContentsline{#1}{mtcSSS}{#3}{#4}%
981 \fi
982 \expandafter\ifx\csname #1\endcsname\paragraph
983 \MTC@WriteContentsline{#1}{mtcP}{#3}{#4}%
984 \fi
985 \expandafter\ifx\csname #1\endcsname\subparagraph
986 \MTC@WriteContentsline{#1}{mtcSP}{#3}{#4}%
987 \fi

```

```

\coffee A coffee break contents line ☕ is written for \coffee:
\MTC@WriteCoffeeline
988 \expandafter\ifx\csname #1\endcsname\coffee
989 \MTC@WriteCoffeeline{#1}{#3}%
990 \fi

\starchapter If it is \starchapter (for a starred chapter), we increment the mtc counter, build a new
\stepcounter minitoc file name, close the file descriptor and open it with this new file.
\if@longextensions@
\mtcname 991 \expandafter\ifx\csname #1\endcsname\starchapter
\themtc 992 \stepcounter{mtc}%
\tf@mtc 993 \if@longextensions@
\closeout 994 \mtcPackageInfo[<I0033>]{minitoc}%
\openout 995 {Writing\space\jobname.mtc\themtc@gobble}%
996 \def\mtcname{\jobname.mtc\themtc}%
997 \else
998 \mtcPackageInfo[<I0033>]{minitoc}%
999 {Writing\space\jobname.M\themtc@gobble}%
1000 \def\mtcname{\jobname.M\themtc}%
1001 \fi
1002 \immediate\closeout\tf@mtc
1003 \immediate\openout\tf@mtc=\mtcname
1004 \fi

\starsection For starred sectioning commands lower than \chapter, a contents line is written into
\MTC@WriteContentsline the minitoc file, with a font command added:
\starsubsection
\starsubsubsection 1005 \expandafter\ifx\csname #1\endcsname\starsection
\starparagraph 1006 \MTC@WriteContentsline{#1}{mtcS}{#3}{#4}%
\starsubparagraph 1007 \fi
1008 \expandafter\ifx\csname #1\endcsname\starsubsection
1009 \MTC@WriteContentsline{#1}{mtcSS}{#3}{#4}%
1010 \fi
1011 \expandafter\ifx\csname #1\endcsname\starsubsubsection
1012 \MTC@WriteContentsline{#1}{mtcSSS}{#3}{#4}%
1013 \fi
1014 \expandafter\ifx\csname #1\endcsname\starparagraph
1015 \MTC@WriteContentsline{#1}{mtcP}{#3}{#4}%
1016 \fi
1017 \expandafter\ifx\csname #1\endcsname\starsubparagraph
1018 \MTC@WriteContentsline{#1}{mtcSP}{#3}{#4}%
1019 \fi
1020 }

\MTC@explist The loop to read the lines of the TOC file; it expands the list of entries and call \MTC@next
\MTC@next to process the first one:
\MTC@list
1021 \def\MTC@explist{\expandafter\MTC@next\MTC@list\}

```



```

\MTC@loop  If an entry is found, loop through line by line, looking for interesting entries. Otherwise,
\openin    process the next entry in the list.
\@inputcheck
\MTC@toc 1022 \def\MTC@loop#1{\openin\@inputcheck#1\relax
\MTC@read 1023 \ifeof\@inputcheck
1024 \mtcPackageWarning[<W0010>]{minitoc}%
1025 {No file #1.\MessageBreak MINITOCs NOT PREPARED}%
1026 \expandafter\MTC@toc
1027 \else
1028 \mtcPackageInfo[<I0024>]{minitoc}{PREPARING MINITOCs FROM #1}%
1029 \expandafter\MTC@read
1030 \fi
1031 }

```

```

\MTC@read  Read the next entry of the .toc file.
\read
\@inputcheck 1032 \def\MTC@read{%
\MTC@line 1033 \read\@inputcheck to\MTC@line

```

```

\MTC@test  The ..... make sure that \MTC@test has enough arguments:
\MTC@line
\MTC@ 1034 \expandafter\MTC@test\MTC@line.....\MTC% %%H0: . added
1035 }%

```

```

\MTC@test  The \MTC@test macro finds the “interesting” commands in the TOC file, mainly to
\contentsline delimit chapters:
\MTC@contentsline
\mtc@string 1036 %%H0/BJ: now patch \MTC@test,
\@input 1037 %%H0/BJ: call \MTC@contentsline with 4 instead of 3 parameters
\MTC@list
\chapterend Look at the first token of the line. If it is an interesting entry, process it. If it is \@input,
\closeout add the file to the list. Otherwise ignore. Go around the loop if not at end of file. Finally
\tf@mtc process the next file in the list.
\openout
\chapterbegin 1038 \long\def\MTC@test#1#2#3#4#5#6\MTC@{% %%H0: #6 added
\addtocounter 1039 \ifx#1\contentsline
\MTC@toc 1040 \let\mtc@string\string
\MTC@read 1041 \MTC@contentsline{#2}{#3}{#4}{#5}%
1042 %%H0/BJ: 4. parameter added by Tony Roberts
1043 \let\mtc@string\relax
1044 \else\ifx#1\@input
1045 \edef\MTC@list{\MTC@list#2\relax}%
1046 \else\ifx#1\chapterend
1047 \immediate\closeout\tf@mtc
1048 \immediate\openout\tf@mtc=\jobname.mtc
1049 \else\ifx#1\chapterbegin
1050 \addtocounter{mtc}{-1}%

```

```

1051 \fi\fi\fi\fi
1052 \ifeof\@inputcheck
1053 \expandafter\MTC@toc
1054 \else
1055 \expandafter\MTC@read
1056 \fi
1057 }%

```

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:

```

1058 \def\MLF@next#1\relax#2\{\%
1059 \edef\MLF@list{#2}%
1060 \MLF@loop{#1}}

```

`\MLF@lof` Checks if the list is empty:

`\MLF@list`

```

\MLF@explist 1061 \def\MLF@lof{%
1062 \ifx\MLF@list\@empty\else\expandafter\MLF@explist\fi}

```

`\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.

`\arabic`

`\xchapter`

```

1063 \def\MLF@contentsline#1#2#3#4{% %%H0: added #4
1064 \gdef\themtc{\arabic{mtc}} %%H0: space removed
1065 \expandafter\ifx\csname #1\endcsname\xchapter
1066 \stepcounter{mtc}%

```

`\if@longextensions@` The name of the minilof file is built from `\jobname` and a long or short extension:

```

\themtc
\mlfname 1067 \if@longextensions@%
\closeout 1068 \mtcPackageInfo[<I0033>]{minitoc}%
\tf@mtc 1069 {Writing\space\jobname.mlf\themtc@gobble}
\openout 1070 \def\mlfname{\jobname.mlf\themtc}%
1071 \else
1072 \mtcPackageInfo[<I0033>]{minitoc}%
1073 {Writing\space\jobname.F\themtc@gobble}
1074 \def\mlfname{\jobname.F\themtc}%
1075 \fi
1076 \immediate\closeout\tf@mtc
1077 \immediate\openout\tf@mtc=\mlfname
1078 \fi

```

`\figure` If we found a `\figure` entry, we copy it into the minilof file:

```

\subfigure
\mtc@toks 1079 \expandafter\ifx\csname #1\endcsname\figure
\MTC@WriteContentsline 1080 \mtc@toks{\noexpand\leavevmode#2}%
1081 \MTC@WriteContentsline{#1}{mlf}{#3}{#4}%
1082 \fi
1083 \expandafter\ifx\csname #1\endcsname\subfigure
1084 \mtc@toks{\noexpand\leavevmode#2}%
1085 \MTC@WriteContentsline{#1}{mlfS}{#3}{#4}%
1086 \fi
1087 }

```

`\MLF@explist` The loop to read the LOF file; it expands the list of entries and calls `\MLF@next` to process the first one:

```

\MLF@next
\MLF@list
1088 \def\MLF@explist{\expandafter\MLF@next\MLF@list\}

```

`\MLF@loop` And now, we scan the `.lof` file:

```

\openin
\@inputcheck 1089 \def\MLF@loop#1{\openin\@inputcheck#1\relax
\MLF@lof 1090 \ifeof\@inputcheck
\MLF@read 1091 \mtcPackageWarning[<W0008>]{minitoc}%
1092 {No file #1.\MessageBreak MINILOFS NOT PREPARED}%
1093 \expandafter\MLF@lof
1094 \else
1095 \mtcPackageInfo[<I0034>]{minitoc}%
1096 {PREPARING MINILOFS FROM #1}%
1097 \expandafter\MLF@read\fi}

```

```

\MLF@read  Read the next entry in the .lof file:
\read
\@inputcheck 1098 \def\MLF@read{%
1099   \read\@inputcheck to\MLF@line

```

```

\MLF@line  The ..... make sure that \MLF@test has enough arguments:
\MLF@test
  \MLF@ 1100   \expandafter\MLF@test\MLF@line.....\MLF@% %%H0: . added
1101   }%

```

\MLF@test The \MLF@test macro finds the “interesting” commands in the LOF file, mainly to delimit chapters.

```

\contentsline Look at the first token of the line. If it is an interesting entry, process it. If it is \@input,
\mtc@string add the file to the list. Otherwise ignore. Go around the loop if not at end of file. Finally
\MLF@contentsline process the next file in the list.
  \@input
    \MLF@list 1102 \long\def\MLF@test#1#2#3#4#5#6\MLF@{% %%H0: #6 added
\chapterend 1103   \ifx#1\contentsline
\closeout 1104     \let\mtc@string\string
  \tf@mtc 1105     \MLF@contentsline{#2}{#3}{#4}{#5}% %%H0: #4 added
\openout 1106     \let\mtc@string\relax
\chapterbegin 1107   \else\ifx#1\@input
\addtocounter 1108     \edef\MLF@list{\MLF@list#2\relax}%
  \MLF@lof 1109     \else\ifx#1\chapterend
  \MLF@read 1110       \immediate\closeout\tf@mtc
1111       \immediate\openout\tf@mtc=\jobname.mtc
1112       \else\ifx#1\chapterbegin
1113         \addtocounter{mtc}{-1}%
1114         \fi\fi\fi\fi
1115         \ifeof\@inputcheck\expandafter\MLF@lof
1116         \else\expandafter\MLF@read\fi}%

```

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

`\MLT@next` This is done via a loop managed by the following macros:
`\MLT@list`
`\MLT@loop` Processes the next entry in the list and removes it from the head of the list:

```
1117 \def\MLT@next#1\relax#2\{\%
1118   \edef\MLT@list{#2}%
1119   \MLT@loop{#1}}
```

`\MLT@lot` Checks if the list is empty:
`\MLT@list`
`\MLT@explist` 1120 \def\MLT@lot{%
 1121 \ifx\MLT@list\@empty\else\expandafter\MLT@explist\fi}

`\MLT@contentsline` The macro `\MLT@contentsline` analyses the lines read from the LOT file and detects
`\arabic` interesting keywords. If `\xchapter` is found, the mtc counter is incremented and a new
`\xchapter` minilot file is created.

```
1122 \def\MLT@contentsline#1#2#3#4{\% %%H0: added #4
1123   \gdef\themtc{\arabic{mtc}}\% %%H0: space removed
1124   \expandafter\ifx\csname #1\endcsname\xchapter
1125     \stepcounter{mtc}%
```

`\if@longextensions@` The name of the minilot file it build from `\jobname` and a long or short extension:

```
\themtc
\mltname 1126   \if@longextensions@
\closeout 1127     \mtcPackageInfo[<I0033>]{minitoc}%
\tf@mtc 1128       {Writing\space\jobname.mlt\themtc\@gobble}%
\openout 1129       \def\mltname{\jobname.mlt\themtc}%
1130       \else
1131       \mtcPackageInfo[<I0033>]{minitoc}%
1132       {Writing\space\jobname.T\themtc\@gobble}%
1133       \def\mltname{\jobname.T\themtc}%
1134       \fi
1135       \immediate\closeout\tf@mtc
1136       \immediate\openout\tf@mtc=\mltname
1137       \fi
```

`\table` If we found a `\table` entry, we copy it into the minilot file:
`\subtable`

```
\mtc@toks 1138   \expandafter\ifx\csname #1\endcsname\table
\MTC@WriteContentsline 1139     \mtc@toks{\noexpand\leavevmode#2}%
1140     \MTC@WriteContentsline{#1}{\mlt}{#3}{#4}%
1141     \fi
```

```

1142 \expandafter\ifx\csname #1\endcsname\subtable
1143 \mtc@toks{\noexpand\leavevmode#2}%
1144 \MTC@WriteContentsline{#1}{mltS}{#3}{#4}%
1145 \fi
1146 }

```

\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

```

1147 \def\MLT@explist{\expandafter\MLT@next\MLT@list\}

```

\MLT@loop And now, we scan the .lot file:

\openin

\@inputcheck 1148 \def\MLT@loop#1{\openin\@inputcheck#1\relax

\MLT@lot 1149 \ifeof\@inputcheck

\MLT@read 1150 \mtcPackageWarning[<W0009>]{minitoc}%

1151 {No file #1.\MessageBreak MINILOTS NOT PREPARED}%

1152 \expandafter\MLT@lot

1153 \else

1154 \mtcPackageInfo[<I0037>]{minitoc}%

1155 {PREPARING MINILOTS FROM #1}%

1156 \expandafter\MLT@read\fi}

\MLT@read Read the next entry in the .lot file:

\read

\@inputcheck 1157 \def\MLT@read{%

1158 \read\@inputcheck to\MLT@line

\MLT@line The make sure that \MLT@test has enough arguments:

\MLT@test

\MLT@ 1159 \expandafter\MLT@test\MLT@line.....\MLT@% %%H0: . added

1160 }%

\MLT@test The \MLT@test macro finds the “interesting” commands in the LOT file, mainly to delimit chapters.

```

\contentsline Look at the first token of the line. If it is an interesting entry, process it. If it is \@input,
\mtc@string add the file to the list. Otherwise ignore. Go around the loop if not at end of file. Finally
\MLT@contentsline process the next file in the list.
\@input
\MLT@list 1161 \long\def\MLT@test#1#2#3#4#5#6\MLT@{% %%H0: #6 added
\chapterend 1162 \ifx#1\contentsline
\closeout 1163 \let\mtc@string\string
\tf@mtc 1164 \MLT@contentsline{#2}{#3}{#4}{#5}% %%H0: #4 added
\openout 1165 \let\mtc@string\relax
\chapterbegin 1166 \else\ifx#1\@input
\addtocounter 1167 \edef\MLT@list{\MLT@list#2\relax}%
\MLT@lot 1168 \else\ifx#1\chapterend
\MLT@read 1169 \immediate\closeout\tf@mtc
1170 \immediate\openout\tf@mtc=\jobname.mtc
1171 \else\ifx#1\chapterbegin
1172 \addtocounter{mtc}{-1}%
1173 \fi\fi\fi\fi
1174 \ifeof\@inputcheck\expandafter\MLT@lot
1175 \else\expandafter\MLT@read\fi}%

```

Note that we terminate with a closing brace to end the chapter-level macros (end of the *else* branch of a `\@ifundefined{chapter}` alternative).

```
1176 }%
```

9.43 Macro to write a contents line

```

\mtc@dot The \MTC@WriteContentsline macro makes the definition of \MTC@contentsline
\MTC@WriteContentsline shorter. An extra \edef level is removed (Heiko OBERDIEK):
\mtc@param
\write The arguments of \MTC@WriteContentsline are:
\tf@mtc
\@resetfont
\mtc@string #1: the #1 argument of \MTC@contentsline;
\contentsline #2: font shorthand =: \csname #2font\endcsname;
\mtc@toks #3: the #3 argument of \MTC@contentsline;
#4: the #4 argument of \MTC@contentsline (hyperlink).

```

```

1177 \def\mtc@dot{.}
1178 \def\MTC@WriteContentsline#1#2#3#4{%
1179 \def\mtc@param{#4}%
1180 \immediate\write\tf@mtc{%
1181 {\string\reset@font
1182 \expandafter\string\csname #2font\endcsname
1183 \string\mtc@string
1184 \string\contentsline{#1}%

```

```

1185     {\the\mtc@toks}%
1186     {\string\reset@font
1187      \expandafter\string\csname #2font\endcsname
1188      \space #3%
1189     }%
1190     \ifx\mtc@dot\mtc@param
1191     \else
1192       {#4}% %%H0/BJ: #4 is hyperlink
1193     \fi
1194   }%
1195 }%
1196 }

```

`\MTC@WriteCoffeeline` And the same for a “coffee” ☕ line. The arguments of the macro `\MTC@WriteCoffeeline` are:

```

\write
\tf@mtc
\@resetfont
\coffeeFont
\mtc@string
\mtc@toks

```

#1: the #1 argument of `\MTC@contentsline`;
#2: the #3 argument of `\MTC@contentsline`.

```

1197 \def\MTC@WriteCoffeeline#1#2#3{%
1198   \immediate\write\tf@mtc{%
1199     {\string\reset@font \string\coffeeFont \string\mtc@string
1200     {\the\mtc@toks}%
1201     {\string\reset@font \string\coffeeFont \space #3%
1202     }%
1203   }%
1204 }%
1205 }

```

9.44 Depth counters for partlofs and partlots

`\AtBeginDocument` If the counters `lofdepth` and `lotdepth` are defined, we create the corresponding new counters: `partlofdepth` and `partlotdepth`. These counters are initialised to 2. This is done after the loading of the packages, in an `\AtBeginDocument` block:

```

\c@lofdepth
\c@lotdepth

```

```

1206 \AtBeginDocument{%
1207   \@ifundefined{c@lofdepth}{}%
1208   {\newcounter{partlofdepth}\setcounter{partlofdepth}{2}}%
1209   \@ifundefined{c@lotdepth}{}%
1210   {\newcounter{partlotdepth}\setcounter{partlotdepth}{2}}%
1211 }

```


9.45 Part level commands

```

\part      If \part is defined, we define some utility commands, a counter (ptc) for the
\theptc    parttocs and related commands (\theptc, \Thepart, \adjustptc, \decrementptc,
\Thepart   \incrementptc), the obsolete command \firstpartis, and the depth counter
\adjustptc parttocdepth.
\decrementptc
\incrementptc 1212 \@ifundefined{part}{}%
\firstpartis 1213 {%
\firstpartis 1214 \def\part{part}
\if@firstpartis@used@ 1215 \def\Thepart{\arabic{ptc}}
\newcounter 1216 \def\firstpartis#1%
\setcounter 1217 {\mtcPackageWarning[<W0004>]{minitoc}%
1218   {\string\firstpartis \space is an obsolete\MessageBreak command}%
1219   \@firstpartis@used@true}
1220 \newcounter{ptc}
1221 \setcounter{ptc}{0}
1222 \newcommand{\adjustptc}[1][1]{\addtocounter{ptc}{#1}}
1223 \def\decrementptc{\addtocounter{ptc}{-1}}
1224 \def\incrementptc{\addtocounter{ptc}{+1}}
1225 \def\theptc{\arabic{ptc}}
1226 \newcounter{parttocdepth}
1227 \setcounter{parttocdepth}{2}

```

\ptc@rule But, sometimes, we need to make a difference between book/report and article classes (is \columnwidth \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, partlofs, and partlots for books. You should redeclare \ptc@rule if you want these rules.

```

1228 \@ifundefined{chapter}%
1229   {\def\ptc@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}}%
1230   {\let\ptc@rule\relax}

```

\ptcindent And we declare the default indentation (both sides) of the parttocs:

```

1231 \newlength\ptcindent
1232 \@ifundefined{chapter}{\ptcindent=24\p@}{\ptcindent=\z@}

```

9.46 Fonts for the parttocs

```

\ptcfont
\ptcSfont
\ptcSSfont
\ptcSSSfont We define the fonts for the parttocs. Note that they are larger if \chapter is defined
\ptcPfont (book/report-like document classes) than when it is not (article-like document
\ptcSPfont classes):
\plffont
\plfSfont
\pltfont
\pltSfont
\ptifont

```

```

1233 \@ifundefined{chapter}{%
1234   \def\ptcfont{\small\rmfamily\upshape\mdseries} % the parttoc
1235   \def\ptcSfont{\small\rmfamily\upshape\bfseries}% (sections)
1236   \let\ptcSSfont\ptcfont      % (subsections)
1237   \let\ptcSSSfont\ptcfont     % (subsubsections)
1238   \let\ptcPfont\ptcfont       % (paragraphs)
1239   \let\ptcSPfont\ptcfont      % (subparagraphs)
1240   \let\plffont\ptcfont        % (figures)
1241   \let\plfSfont\ptcfont       % (subfigures)
1242   \let\pltfont\ptcfont        % (tables)
1243   \let\pltSfont\ptcfont       % (subtables)
1244   \def\ptifont{\Large\rmfamily\upshape\bfseries}% titles
1245 }%

\ptcfont  If \chapter is defined, the fonts are larger and \ptcCfont must be defined:
\ptcCfont
\ptcSfont 1246 {%
\ptcSSfont 1247 \def\ptcfont{\normalsize\rmfamily\upshape\mdseries} % the parttoc
\ptcSSSfont 1248 \def\ptcCfont{\normalsize\rmfamily\upshape\bfseries}% (chapters)
\ptcPfont 1249 \def\ptcSfont{\normalsize\rmfamily\upshape\mdseries}% (sections)
\ptcSPfont 1250 \let\ptcSSfont\ptcfont      % (subsections)
\plffont 1251 \let\ptcSSSfont\ptcfont     % (subsubsections)
\plfSfont 1252 \let\ptcPfont\ptcfont       % (paragraphs)
\pltfont 1253 \let\ptcSPfont\ptcfont      % (subparagraphs)
\pltSfont 1254 \let\plffont\ptcfont        % (figures)
\ptifont 1255 \let\plfSfont\ptcfont       % (subfigures)
1256 \let\pltfont\ptcfont        % (tables)
1257 \let\pltSfont\ptcfont       % (subtables)
1258 \def\ptifont{\LARGE\rmfamily\upshape\bfseries}% titles
1259 }

```

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-
\partlof tables (\parttoc, \partlof and \partlot). This formatting is different if \chapter is
\partlot defined or undefined.

```

\c@pti  If \chapter is undefined, the definitions are very simple, for centered, flushleft,
\l@pti  flushright or empty titles. Empty titles need a vertical correction (Frank MITTELBACH).
\r@pti
\c@pti 1260 \@ifundefined{chapter}{%
\l@pti 1261 \def\c@pti#1{\null\hfill #1\hfill\hfill\hfill}
1262 \def\l@pti#1{\null #1\hfill\hfill\hfill}
1263 \def\r@pti#1{\null\hfill #1\hfill\hfill}
1264 \def\c@pti#1{\vspace{-\baselineskip}}
1265 \def\l@pti#1{\vspace{-\baselineskip}}
1266 }%

```

`\e@pti` But, if `\chapter` is defined, we must simulate the formatting of a chapter head, which is
`\n@pti` more complex. Empty titles need a vertical correction (Frank MITTELBACH).

```
1267 {%
1268 \def\e@pti#1{\vspace{-\baselineskip}}
1269 \def\n@pti#1{\vspace{-\baselineskip}}
```

`\l@pti` For a title on the left, we must test if the main text is on two columns:

```
\if@twocolumn
\@topnewpage 1270 \def\l@pti#1{\if@twocolumn
\@afterheading 1271 \@topnewpage[\@makehead@l{#1}]\%
\ptifont 1272 \else
\@makehead@l 1273 \@makehead@l{#1}%
\mtcgapbeforeheads 1274 \@afterheading
\mtcgapafterheads 1275 \fi}
1276 \def\@makehead@l#1{%
1277 \vspace*{\mtcgapbeforeheads}%
1278 {\parindent \z@ \raggedright
1279 \ptifont
1280 #1\par
1281 \nobreak
1282 \vskip \mtcgapafterheads\hbox{}}
1283 }}
```

`\r@pti` For a title on the right, we must also test if the main text is on two columns:

```
\if@twocolumn
\@topnewpage 1284 \def\r@pti#1{\if@twocolumn
\@makehead@r 1285 \@topnewpage[\@makehead@r{#1}]\%
\@afterheading 1286 \else
\ptifont 1287 \@makehead@r{#1}%
\mtcgapbeforeheads 1288 \@afterheading
\mtcgapafterheads 1289 \fi}
1290 \def\@makehead@r#1{%
1291 \vspace*{\mtcgapbeforeheads}%
1292 {\parindent \z@ \raggedleft
1293 \ptifont
1294 #1\par
1295 \nobreak
1296 \vskip \mtcgapafterheads\hbox{}}
1297 }}
```

`\c@pti` For a centered title, we must also test if the main text is on two columns:

```
\if@twocolumn
\@topnewpage 1298 \def\c@pti#1{\if@twocolumn
\@makehead@c 1299 \@topnewpage[\@makehead@c{#1}]\%
\@afterheading 1300 \else
\ptifont 1301 \@makehead@c{#1}%
\mtcgapbeforeheads 1302 \@afterheading
\mtcgapafterheads
```

```

1303 \fi}
1304 \def\@makephead@c#1{%
1305 \vspace*{\mtcgapbeforeheads}%
1306 {\parindent \z@ \centering
1307 \ptifont
1308 #1\par
1309 \nobreak
1310 \vskip \mtcgapafterheads\hbox{}}
1311 }}%
1312 }

```

\l@pti By default, titles are on left:

```

\do@ptitc
\df@ptitc 1313 \let\do@ptitc\l@pti
\do@ptilf 1314 \let\df@ptitc\l@pti
\df@ptilf 1315 \let\do@ptilf\l@pti
\do@ptilt 1316 \let\df@ptilf\l@pti
\df@ptilt 1317 \let\do@ptilt\l@pti
1318 \let\df@ptilt\l@pti

```

9.48 The ptc@verse environment

ptc@verse Each parttoc is placed inside a ptc@verse environment. This environment is analog to the standard verse environment and hence defined via two commands: \ptc@verse and \endptc@verse. As it is a list environment, we first define (in a local way) \list, then call \list{} and set some dimensions like \itemsep, \itemindent, \listparindent, \itemindent, \partopsep, \topsep. \parsep is set to zero if the tight option is active (this reduces the spacing between the lines). \parskip is set to zero if the k-tight option is active (this reduces the spacing between the lines). Both margins are set to \ptcindent. \endptc@verse terminates the list and discourages a page break.

```

\listparindent
\topsep 1319 \def\ptc@verse{\let\list=\@centercr
\parsep 1320 \list{}\itemsep\z@
\parskip 1321 \itemindent \z@
\partopsep 1322 \listparindent \itemindent
\ptcindent 1323 \partopsep\z@
\iftightmtc 1324 \iftightmtc \parsep\z@ \fi
\ifktightmtc 1325 \ifktightmtc \parskip\z@ \fi
1326 \topsep=lex
1327 \leftmargin\ptcindent
1328 \rightmargin\leftmargin\item[]}
1329 \def\endptc@verse{\nopagebreak[4]\endlist}

```

9.49 The part level mini-tables: `\parttoc`, `\partlof`, and `\partlot`

`\parttoc` These commands are essentially similar to the `\minitoc` command, except that they
`\partlof` should be placed after a `\part` command to produce a `parttoc`, a `partlof` or a `partlot`,
`\partlot` and the formatting is different and depends of the availability of the `\chapter`
command (for the fonts and the horizontal rules). The code is very similar. The
`\partlof` and `\partlot` commands are siblings of the `\parttoc` command. Note
that `\parttoc`, `\partlof` and `\partlot` use page styles, because `\beforepart...`
and `\afterpart...` commands imply usually a `\clear[double]page` command,
and hence `\markboth{...}{...}` must be called.

9.49.1 The `\parttoc` command

`\parttoc` This command must be used after `\part` if you need a `parttoc` (no automatic `parttoc`).
`\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:

```
1330 \def\parttoc{\@ifnextchar{\{\parttoc@\}\parttoc@[d]}}
```

`\parttoc@` The `\parttoc@` macro does the real work. It first sets the flag `\if@parttoc@used@` (for
`\if@parttoc@used@` a coherence hint) and checks if long extensions are used or not (to create the name of the
`\if@longextensions@` `parttoc` file):

```
\@tocfile
\Thepart 1331 \def\parttoc@[#1]{%
1332 \global\@parttoc@used@true
1333 \if@longextensions@%
1334 \def\@tocfile{ptc\Thepart}%
1335 \else
1336 \def\@tocfile{P\Thepart}%
1337 \fi
```

`\mtc@CkFile` Then, we check the presence of the `parttoc` file and give a warning if it is not here:

```
\if@mtc@FE
\@tocfile 1338 \mtc@CkFile{\jobname.\@tocfile}
1339 \if@mtc@FE
1340 \mtc@PackageInfo[<I0006>]{minitoc}%
1341 {\jobname.\@tocfile\space is empty}
1342 \@mtc@empty@parttoc@true
1343 \else
```

`\beforeparttoc` If the parttoc file is present, we can insert it, but we must add some presentation code: first, `\beforeparttoc`, of course:

```
1344      \beforeparttoc
```

```

\mtc@markboth  If \chapter is defined, we just set the page marks with the parttoc title and set the page
\@mkboth      style:
\thispageparttocstyle
\MakeUppercase 1345      \@ifundefined{chapter}{}{}%
\ptctitle      1346      \global\let\mtc@markboth\markboth
                1347      \global\let\@mkboth\markboth
                1348      \thispageparttocstyle
                1349      \mtc@markboth{\MakeUppercase{\ptctitle}}{\MakeUppercase{\ptctitle}}%
                1350      }%
```

```

\do@ptitc      A samepage environment is begun, then the argument is treated to set the position of the
\@pti          parttoc title. If the title string is empty, this forces the positionning.
\@npti
\@c@pti 1351      \relax\begin{samepage}%
\@l@pti 1352      \if #1e\let\do@ptitc\@e@pti
\@r@pti 1353      \else\if #1n\let\do@ptitc\@n@pti
\@df@pti 1354      \else\if #1c\let\do@ptitc\@c@pti
\mtc@CkStr 1355      \else\if #1l\let\do@ptitc\@l@pti
\ptctitle 1356      \else\if #1r\let\do@ptitc\@r@pti
\if@mtc@FE 1357      \else\if #1d\let\do@ptitc\@df@ptitc
samepage     1358      \fi\fi\fi\fi\fi\fi
            1359      \mtc@CkStr{\ptctitle}\if@mtc@FE \let\do@ptitc\@e@pti\relax\fi
```

`\raggedright` We adjust some formatting parameters and avoid a page break between the title and the parttoc, then we set the font:

```

\parskip
\ptcfont
1360      \raggedright
1361      \parskip=\z@%
1362      \reset@font\ptcfont%
1363      \parindent=\z@%
1364      \nopagebreak[4]%
1365      \kern-0.8\baselineskip\nopagebreak[4]%
1366      \par\noindent
1367      \nopagebreak[4]%
```

`\ptc@rule` The parttoc 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 parttoc.

```

\columnwidth
\ptifont
\do@ptitc 1368      \ifx\ptc@rule\relax
\mtc@v
\ptctitle
\hline
```

```

1369     \begin{tabular}{@{}p{\columnwidth}@{}}
1370     \reset@font\ptifont\do@ptitc{\mtc@v\ptctitle}\\
1371     \end{tabular}%
1372     \else
1373     \begin{tabular}{@{}p{\columnwidth}@{}}
1374     \reset@font\ptifont\do@ptitc{\mtc@v\ptctitle}\\\hline
1375     \end{tabular}%
1376     \fi

```

\mtc@zrule Then, we adjust the position under the top rule and set the indentation and some formatting parameters:

\mtc@BBR

\ptcindent

```

1377     \nopagebreak[4]\null\leavevmode\mtc@zrule\\*[-\baselineskip]\mtc@BBR
1378     \leftmargin\ptcindent
1379     \rightmargin\ptcindent
1380     \itemindent=\z@\labelwidth=\z@%
1381     \labelsep=\z@\listparindent=\z@%

```

ptc@verse We enter in a ptc@verse environment to format the parttoc. The toc depth is forced (locally) to parttocdepth. A little trick is necessary to adjust the position. A blank line is necessary to avoid a negative indentation.

\mtc@BBR

```

1382     \begin{ptc@verse}\c@tocdepth=\c@parttocdepth%
1383     \leavevmode\\\mtc@BBR\vskip -.5\baselineskip

```

\ptc@pgno 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.

\@tocfile

\mtc@strut 1384 \begingroup

ptc@verse

```

1385     \makeatletter
1386     \@ifundefined{ptc@pgno}%
1387     {\let\@dottedtocline\@undottedtocline}{}
1388     \@ifundefined{ptc@pgno}%
1389     {\let\l@chapter@SVPN\l@chapter%
1390      \def\l@chapter##1##2{\l@chapter@SVPN{##1}{\hbox{}}}}{}
1391     \@filesfalse\mtc@hook@beforeinputfile
1392     \ptc@setform
1393     \@input{\jobname.\@tocfile}
1394     \vspace{-1ex} \vspace{-1\baselineskip}
1395     \leavevmode\mtc@strut
1396     \global\@nobreakfalse\endgroup
1397     \end{ptc@verse}%

```

`\ptc@rule` The final part is just to add the bottom rule, if necessary, a possible page break (if
`\mtc@zrule` `\chapter` is not defined), and `\afterparttoc`.
`samepage`
`\afterparttoc` 1398 `\kernafterparttoc`
1399 `\nopagebreak[4]\ptc@rule\null\leavevmode\\%`
1400 `\vskip-1.0\baselineskip\mtc@zrule\end{samepage}%`
1401 `\par\@ifundefined{chapter}{\pagebreak[1]\vspace*{-1ex}}%`
1402 `\afterparttoc\fi%`

9.49.2 The `\partlof` command

`\partlof` 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:

```
1403 \def\partlof{\@ifnextchar[{\partlof@}{\partlof@d}}
```

`\partlof@` The `\partlof@` macro does the real work. It first sets the flag `\if@partlof@used@` (for
`\if@partlof@used@` a coherence hint) and checks if long extensions are used or not (to create the name of the
`\if@longextensions@` `partlof` file):
`\@tocfile`
`\Thepart` 1404 `\def\partlof@[#1]{%`
1405 `\global\@partlof@used@true`
1406 `\if@longextensions@%`
1407 `\def\@tocfile{plf\Thepart}%`
1408 `\else`
1409 `\def\@tocfile{G\Thepart}%`
1410 `\fi`

`\mtc@CkFile` Then, we check the presence of the `partlof` file and give a warning if it is not here:

```
\if@mtc@FE
\@tocfile 1411 \mtc@CkFile{\jobname.\@tocfile}
1412 \if@mtc@FE
1413 \mtcPackageInfo[<I0006>]{minitoc}%
1414 {\jobname.\@tocfile\space is empty}
1415 \@mtc@empty@partlof@true
1416 \else
```

`\beforepartlof` If the `partlof` file is present, we can insert it, but we must add some presentation code: first, `\beforepartlof`, of course:

```
1417 \beforepartlof
```



```

\mtc@markboth  If \chapter is defined, we just set the page marks with the partlof title and set the page
\@mkboth      style:
\thispagepartlofsty
\MakeUppercase 1418      \@ifundefined{chapter}{}{}%
\plftitle      1419      \global\let\mtc@markboth\markboth
                1420      \global\let\@mkboth\markboth
                1421      \thispagepartlofsty
                1422      \mtc@markboth{\MakeUppercase{\plftitle}}{\MakeUppercase{\plftitle}}%
                1423      }%

\do@ptilf      A samepage environment is begun, then the argument is treated to set the position of the
\@epti         partlof title. If the title string is empty, this forces the positionning.
\@npti
\@cpti 1424      \relax\begin{samepage}%
\@lpti 1425      \if #1e\let\do@ptilf\@epti
\@rpti 1426      \else\if #1n\let\do@ptilf\@npti
\@dfpti 1427      \else\if #1c\let\do@ptilf\@cpti
\mtc@CkStr 1428      \else\if #1l\let\do@ptilf\@lpti
\plftitle 1429      \else\if #1r\let\do@ptilf\@rpti
\if@mtc@FE 1430      \else\if #1d\let\do@ptilf\@dfptilf
samepage      1431      \fi\fi\fi\fi\fi\fi
                1432      \mtc@CkStr{\plftitle}\if@mtc@FE \let\do@ptilf\@epti\relax\fi

\raggedright   We adjust some formatting parameters and avoid a page break between the title and the
\parskip       parttoc, then we set the font:
\plffont
                1433      \raggedright
                1434      \parskip=\z@%
                1435      \reset@font\plffont%

\plf@rule      The parttoc title is set in a tabular environment (to inhibit a page break between the title
tabular        and the top rule), with a rule at its bottom if necessary. This rule is an \hline. It is the
\columnwidth   top rule of the partlof.
\ptifont
\do@ptilf 1436      \parindent=\z@%
\mtc@v 1437      \nopagebreak[4]%
\plftitle 1438      \kern-0.8\baselineskip\nopagebreak[4]%
\hline 1439      \par\noindent
                1440      \ifx\plf@rule\relax
                1441      \begin{tabular}{@{}p{\columnwidth}@{}}
                1442      \reset@font\ptifont\do@ptilf{\mtc@v\plftitle}\\
                1443      \end{tabular}%
                1444      \else
                1445      \begin{tabular}{@{}p{\columnwidth}@{}}
                1446      \reset@font\ptifont\do@ptilf{\mtc@v\plftitle}\\ \hline
                1447      \mtc@hstrut\\
                1448      \end{tabular}%
                1449      \fi

```

`\mtc@zrule` Then, we adjust the position under the top rule and set the indentation and some
`\mtc@BBR` formatting parameters:

```
\ptcindent
1450      \nopagebreak[4]\null\leavevmode\mtc@zrule\\*[-\baselineskip]\mtc@BBR
1451      \leftmargin\ptcindent
1452      \rightmargin\ptcindent
1453      \itemindent=\z@\labelwidth=\z@%
1454      \labelsep=\z@\listparindent=\z@%
```

`ptc@verse` We enter in a `ptc@verse` environment to format the partlof. If necessary, the toc depth
`\mtc@BBR` is forced (locally) to `partlofdepth`. A little trick is necessary to adjust the position. A
blank line is necessary to avoid a negative indentation.

```
1455      \begin{ptc@verse}%
1456      \@ifundefined{c@lofdepth}%
1457      {}%
1458      {\c@lofdepth=\c@partlofdepth
1459      \ifnum\c@lofdepth<1\relax\c@lofdepth=1\fi}
1460      \leavevmode\\ \mtc@BBR\vskip -.5\baselineskip
```

`\plf@pgno` If the contents lines must have no numbers, we replace the macro `\@dottedtocline`
`\@dottedtocline` with its undotted version. A hook is added, and the formatting settings coming from
`\@undottedtocline` `\mtcsetformat` are activated via `\plf@setform`. Then the partlof file is inserted,
`\mtc@hook@beforeinputfile` followed by a strut, and the `ptc@verse` environment is terminated.

```
\plf@setform
\@tocfile 1461 \begingroup
\mtc@strut 1462 \makeatletter
ptc@verse 1463 \@ifundefined{plf@pgno}%
1464 {\let\@dottedtocline\@undottedtocline}{}
1465 \@filesfalse\mtc@hook@beforeinputfile
1466 \plf@setform
1467 \@input{\jobname.\@tocfile}
1468 \vspace{-1ex} \vspace{-1\baselineskip}
1469 \leavevmode\mtc@strut
1470 \global\@nbreakfalse\endgroup
1471 \end{ptc@verse}%
```

`\plf@rule` The final part is just to add the bottom rule, if necessary, a possible page break (if
`\mtc@zrule` `\chapter` is not defined), and `\afterpartlof`. The blank line (`\\`) is essential.
`samepage`

```
\afterpartlof 1472 \kernafterpartlof
1473 \nopagebreak[4]\plf@rule\null\leavevmode\\%
1474 \vskip-1.0\baselineskip\mtc@zrule\end{samepage}%
1475 \par\@ifundefined{chapter}{\pagebreak[1]\vspace*{-1ex}}%
1476 \afterpartlof\fi}
```

9.49.3 The `\partlot` command

`\partlot` This command must be used after `\part` if you need a partlot (no automatic partlot). First, `\partlot@` `\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:

```
1477 \def\partlot{\ifnextchar{\\partlot@}\partlot@d}}
```

`\partlot@` The `\partlot@` macro does the real work. It first sets the flag `\if@partlot@used@` (for a coherence hint) and checks if long extensions are used or not (to create the name of the partlot file):

```
\if@longextensions@
  \@tocfile
  \Thepart 1478 %
            1479 \def\partlot@[#1]{%
            1480 \global\@partlot@used@true
            1481 \if@longextensions@%
            1482 \def\@tocfile{plt\Thepart}%
            1483 \else
            1484 \def\@tocfile{U\Thepart}%
            1485 \fi
```

`\mtc@CkFile` Then, we check the presence of the partlot file and give a warning if it is not here:

```
\if@mtc@FE
  \@tocfile 1486 \mtc@CkFile{\jobname.\@tocfile}
            1487 \if@mtc@FE
            1488 \mtcPackageInfo[<I0006>]{minitoc}%
            1489 {\jobname.\@tocfile\space is empty}
            1490 \@mtc@empty@partlot@true
            1491 \else
```

`\beforepartlot` If the partlot file is present, we can insert it, but we must add some presentation code: first, `\beforepartlot`, of course:

```
1492 \beforepartlot
```

`\mtc@markboth` If `\chapter` is defined, we just set the page marks with the partlot title and set the page style:

```
\@mkboth
\thispagepartlotstyle
\MakeUppercase 1493 \@ifundefined{chapter}{}{%
  \plttitle 1494 \global\let\mtc@markboth\markboth
            1495 \global\let\@mkboth\markboth
            1496 \thispagepartlotstyle
            1497 \mtc@markboth{\MakeUppercase{\plttitle}}{\MakeUppercase{\plttitle}}%
            1498 }%
```

```

\do@ptilt A samepage environment is begun, then the argument is treated to set the position of the
\e@pti partlof title. If the title string is empty, this forces the positionning.
\n@pti
\c@pti 1499 \relax\begin{samepage}%
\l@pti 1500 \if #1e\let\do@ptilt\e@pti
\r@pti 1501 \else\if #1n\let\do@ptilt\n@pti
\df@pti 1502 \else\if #1c\let\do@ptilt\c@pti
\mtc@CkStr 1503 \else\if #1l\let\do@ptilt\l@pti
\plttitle 1504 \else\if #1r\let\do@ptilt\r@pti
\if@mtc@FE 1505 \else\if #1d\let\do@ptilt\df@ptilt
samepage 1506 \fi\fi\fi\fi\fi\fi
1507 \mtc@CkStr{\plttitle}\if@mtc@FE \let\do@ptilt\e@pti\relax\fi

\raggedright We adjust some formatting parameters and avoid a page break between the title and the
\parskip partlot, then we set the font:
\pltfont
1508 \raggedright
1509 \parskip=\z@%
1510 \reset@font\pltfont%
1511 \parindent=\z@%
1512 \nopagebreak[4]%
1513 \kern-0.8\baselineskip\nopagebreak[4]%
1514 \par\noindent

\plt@rule The partlot title is set in a tabular environment (to inhibit a page break between the title
tabular and the top rule), with a rule at its bottom if necessary. This rule is an \hline. It is the
\columnwidth top rule of the partlot.
\ptifont
\do@ptilt 1515 \ifx\plt@rule\relax
\mtc@v 1516 \begin{tabular}{@{}p{\columnwidth}@{}}
\plttitle 1517 \reset@font\ptifont\do@ptilt{\mtc@v\plttitle}\\
\hline 1518 \end{tabular}%
1519 \else
1520 \begin{tabular}{@{}p{\columnwidth}@{}}
1521 \reset@font\ptifont\do@ptilt{\mtc@v\plttitle}\\ \hline
1522 \mtc@hstrut\\
1523 \end{tabular}%
1524 \fi

\mtc@zrule Then, we adjust the position under the top rule and set the indentation and some
\mtc@BBR formatting parameters:
\ptcindent
1525 \nopagebreak[4]\null\leavevmode\mtc@zrule\\*[-\baselineskip]\mtc@BBR
1526 \leftmargin\ptcindent
1527 \rightmargin\ptcindent
1528 \itemindent=\z@\labelwidth=\z@%
1529 \labelsep=\z@\listparindent=\z@%
```

`ptc@verse` We enter in a `ptc@verse` environment to format the parttoc. If necessary, the toc depth
`\mtc@BBR` is forced (locally) to `partlotdepth`. A little trick is necessary to adjust the position.

```

1530      \begin{ptc@verse}%
1531      \@ifundefined{c@lotdepth}%
1532      {}%
1533      {\c@lotdepth=\c@partlotdepth
1534      \ifnum\c@lotdepth<1\relax\c@lotdepth=1\fi}
1535      \leavevmode\\mtc@BBR\vskip -.5\baselineskip

```

`\plt@pgno` If the contents lines must have no numbers, we replace the macro `\@dottedtocline`
`\@dottedtocline` with its undotted version. A hook is added, and the formatting settings coming from
`\@undottedtocline` `\mtcsetformat` are activated via `\plt@setform`. Then the partlot file is inserted,
`\mtc@hook@beforeinputfile` followed by a strut, and the `ptc@verse` environment is terminated.

```

\plt@setform
\@tocfile 1536 \begingroup
\mtc@strut 1537 \makeatletter
ptc@verse 1538 \@ifundefined{plt@pgno}%
1539      {\let\@dottedtocline\@undottedtocline}{}
1540      \@filesfalse\mtc@hook@beforeinputfile
1541      \plt@setform
1542      \@input{\jobname.\@tocfile}
1543      \vspace{-1ex} \vspace{-1\baselineskip}
1544      \leavevmode\mtc@strut
1545      \global\@nobreakfalse\endgroup
1546      \end{ptc@verse}%

```

`\ptc@rule` The final part is just to add the bottom rule, if necessary, a possible page break (if
`\mtc@zrule` `\chapter` is not defined), and `\afterpartlot`. The blank line (\\) is essential.
`samepage`
`\afterpartlot` 1547 `\kernafterpartlot`
1548 `\nopagebreak[4]\plt@rule\null\leavevmode\\%`
1549 `\vskip-1.0\baselineskip\mtc@zrule\end{samepage}%`
1550 `\par\@ifundefined{chapter}{\pagebreak[1]\vspace*{-1ex}}%`
1551 `\afterpartlot\fi}`

9.50 Auxiliary commands for printing parttocs

`\@dottedtocline` The following auxiliary commands are used in the printing of parttocs. Note that
`\l@xpart` `\l@xpart` uses a huge depth to inhibit the printing of its contents line (except if you
`\l@pchapter` cheat). These command are similar to `\l@subsection`, only the arguments have been
`\l@psect` altered:
`\pchapter`
`\psect` 1552 `\def\l@xpart{\@dottedtocline{\@M}{1.0em}{2.3em}}`
1553 `\def\l@pchapter{\@dottedtocline{1}{1.0em}{2.3em}}`

```

1554 \def\l@psect{\@dottedtocline{2}{1.0em}{2.3em}}
1555 \def\pchapter{pchapter}
1556 \def\psect{psect}

```

9.51 Patching the \part command, continued

```

\sv@part We patch both branches of the \part command: \@part (unstarred \part) and \@spart
\mtc@svpart (\part*). We add the incrementation of the ptc counter to both branches. In the
\@part unstarred branch, we add xpart entries in the TOC, the LOF and the LOT. In the starred
\addtocontents branch, we add a \partbegin line in the TOC. This command is just a marker and does
\sv@spart nothing real (\relax).
\ptc@spart
\@spart 1557 \let\sv@part\mtc@svpart
\stepcounter 1558 \def\@part[#1]#2{\sv@part[{#1}]{#2}\relax
\partbegin 1559 \addcontentsline{lof}{xpart}{#1}%
\partend 1560 \addcontentsline{lot}{xpart}{#1}%
1561 \addcontentsline{toc}{xpart}{#1}%
1562 \stepcounter{ptc}}
1563 \let\sv@spart\@spart
1564 \def\@spart{\stepcounter{ptc}\sv@spart}
1565 \let\ptc@spart\@spart
1566 \def\@spart{\addtocontents{toc}{\protect\partend}\ptc@spart}
1567 \def\@spart{\addtocontents{toc}{\protect\partbegin}\ptc@spart}
1568 \let\partend\relax
1569 \let\partbegin\relax

```

9.52 The \doparttoc command and its siblings

```

\doparttoc The \doparttoc command works like the \dominitoc command, \dopartlof like
\dopartlof \dominilof and \dopartlot like \dominilot.
\dopartlot

```

```

\@doparttoc The \doparttoc command extracts information from the .toc file and creates the
\PTC@next .ptc(N) files (.ptc becomes .P on MS-DOS).
\setcounter
1570 \def\@doparttoc#1{%
1571 \makeatletter
1572 \setcounter{ptc}{0}%
1573 \PTC@next#1.toc\relax\\}\setcounter{ptc}{0}}%

```

`\@dopartlof` The `\dopartlof` command extracts information from the `.lof` file and creates the
`\PLF@next` `.plf(N)` files (`.plf` becomes `.G` on MS-DOS).
`\setcounter`

```
1574 \def\@dopartlof#1{%
1575   \makeatletter
1576   \setcounter{ptc}{0}%
1577   \PLF@next#1.lof\relax\}\setcounter{ptc}{0}}%
```

`\@dopartlot` The `\dopartlot` command extracts information from the `.lot` file and creates the
`\PLT@next` `.plt(N)` files (`.plt` becomes `.U` on MS-DOS).
`\setcounter`

```
1578 \def\@dopartlot#1{%
1579   \makeatletter
1580   \setcounter{ptc}{0}%
1581   \PLT@next#1.lot\relax\}\setcounter{ptc}{0}}%
```

`\doparttoc` We define the user macros, who detect the optional argument:

`\dopartlof`

```
\dopartlot 1582 \def\doparttoc{\@ifnextchar[{\doparttoc@}{\doparttoc@[1]}}
1583 \def\dopartlof{\@ifnextchar[{\dopartlof@}{\dopartlof@[1]}}
1584 \def\dopartlot{\@ifnextchar[{\dopartlot@}{\dopartlot@[1]}}
```

`\doparttoc@` We treat the optional argument of `\doparttoc` (it becomes the default position for titles
`\if@mtc@hints@` of parttocs) and flag this macro as used; a hint detects any spurious invocation.

`\if@doparttoc@used@`

```
\@mtc@hints@given@true 1585 \def\doparttoc@[#1]{%
\df@ptitc 1586 \if@mtc@hints@
\@pti 1587 \if@doparttoc@used@
\n@pti 1588 \mtcPackageInfo[<I0045>]{minitoc(hints)}%
\c@pti 1589 {The \string\doparttoc \space command\MessageBreak
\l@pti 1590 has been invoked more than once\MessageBreak}
\r@pti 1591 \global\@mtc@hints@given@true
1592 \fi
1593 \fi
1594 \global\@doparttoc@used@true
1595 \if #1e\let\df@ptitc\@pti%
1596 \else\if #1n\let\df@ptitc\n@pti%
1597 \else\if #1c\let\df@ptitc\c@pti%
1598 \else\if #1l\let\df@ptitc\l@pti%
1599 \else\if #1r\let\df@ptitc\r@pti%
1600 \fi\fi\fi\fi\fi%
1601 \@doparttoc}
```

```

\dopartlof@ We treat the optional argument of \dopartlof (it becomes the default position for titles
\if@dopartlof@used@ of partlofs) and flag this macro as used, a hint detects any spurious invocation.
\if@mtc@hints@
\@mtc@hints@given@true 1602 \def\dopartlof@[#1]{%
\df@ptilf 1603 \if@mtc@hints@
\@e@pti 1604 \if@dopartlof@used@
\@n@pti 1605 \mtcPackageInfo[<I0045>]{minitoc(hints)}%
\@c@pti 1606 {The \string\dopartlof \space command\MessageBreak
\@l@pti 1607 has been invoked more than once\MessageBreak}
\@r@pti 1608 \global\@mtc@hints@given@true
1609 \fi
1610 \fi
1611 \global\@dopartlof@used@true
1612 \if #1e\let\df@ptilf\@e@pti%
1613 \else\if #1n\let\df@ptilf\@n@pti%
1614 \else\if #1c\let\df@ptilf\@c@pti%
1615 \else\if #1l\let\df@ptilf\@l@pti%
1616 \else\if #1r\let\df@ptilf\@r@pti%
1617 \fi\fi\fi\fi\fi%
1618 \@dopartlof}

```

```

\dopartlot@ We treat the optional argument of \dopartlot (it becomes the default position for titles
\if@mtc@hints@ of partlofs) and flag this macro as used; a hint detects any spurious invocation.
\@mtc@hints@given@true
\if@dopartlot@used@ 1619 \def\dopartlot@[#1]{%
\df@ptilt 1620 \if@mtc@hints@
\@e@pti 1621 \if@dopartlot@used@
\@n@pti 1622 \mtcPackageInfo[<I0045>]{minitoc(hints)}%
\@c@pti 1623 {The \string\dopartlot \space command\MessageBreak
\@l@pti 1624 has been invoked more than once\MessageBreak}
\@r@pti 1625 \global\@mtc@hints@given@true
1626 \fi
1627 \fi
1628 \global\@dopartlot@used@true
1629 \if #1e\let\df@ptilt\@e@pti%
1630 \else\if #1n\let\df@ptilt\@n@pti%
1631 \else\if #1c\let\df@ptilt\@c@pti%
1632 \else\if #1l\let\df@ptilt\@l@pti%
1633 \else\if #1r\let\df@ptilt\@r@pti%
1634 \fi\fi\fi\fi\fi%
1635 \@dopartlot}

```

\@@doparttoc These macros invoke the \@dopart... commands to create the mini-table file, then close the file descriptor.

\@@dopartlof

\@@dopartlot

```

\tf@mtc 1636 \def\@@doparttoc{\@doparttoc{\jobname}\immediate\closeout\tf@mtc}
1637 \def\@@dopartlof{\@dopartlof{\jobname}\immediate\closeout\tf@mtc}
1638 \def\@@dopartlot{\@dopartlot{\jobname}\immediate\closeout\tf@mtc}

```


9.52.1 Processing macros for the parttocs

`\PTC@next` Processing the next entry in the list and remove it from the head of the list:

`\PTC@list`

```
\PTC@loop 1639 \def\PTC@next#1\relax#2\{\%
1640 \edef\PTC@list{#2}%
1641 \PTC@loop{#1}}
```

`\PTC@toc` Check if the list is empty:

`\PTC@list`

```
\PTC@explist 1642 \def\PTC@toc{%
1643 \ifx\PTC@list\@empty\else\expandafter\PTC@explist\fi}
```

`\PTC@contentsline` 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.

`\tf@mtc`

```
\ptcname 1644 \def\PTC@contentsline#1#2#3#4{% %%HO/DV: 4 instead of 3 parameters
\MTC@WriteContentsLine 1645 \expandafter\ifx\csname #1\endcsname\part
1646 \stepcounter{ptc}%
1647 \if@longextensions@%
1648 \mtcPackageInfo[<I0033>]{minitoc}%
1649 {Writing\space\jobname.ptc\theptc\@gobble}%
1650 \def\ptcname{\jobname.ptc\theptc}%
1651 \else
1652 \mtcPackageInfo[<I0033>]{minitoc}%
1653 {Writing\space\jobname.P\theptc\@gobble}%
1654 \def\ptcname{\jobname.P\theptc}%
1655 \fi
1656 \immediate\closeout\tf@mtc
1657 \immediate\openout\tf@mtc=\ptcname
1658 \fi
1659 \expandafter\ifx\csname #1\endcsname\starpart\relax
1660 \stepcounter{ptc}%
```

`\if@longextensions@` We test if long or short extensions are used, to build the name of the mini-table file, then open it:

`\ptcname`

```
1661 \if@longextensions@%
1662 \mtcPackageInfo[<I0033>]{minitoc}%
1663 {Writing\space\jobname.ptc\theptc}%
1664 \def\ptcname{\jobname.ptc\theptc}%
1665 \else
1666 \mtcPackageInfo[<I0033>]{minitoc}%
1667 {Writing\space\jobname.P\theptc}%
1668 \def\ptcname{\jobname.P\theptc}%
1669 \fi
```

```

1670 \immediate\closeout\tf@mtc
1671 \immediate\openout\tf@mtc=\ptcname
1672 \fi

```

\mtc@toks The token register \mtc@toks is used to pass the entry to \MTC@WriteContentsline:

```

1673 \mtc@toks{\noexpand\leavevmode #2}%

```

\MTC@WriteContentsline Now, we filter the relevant contents lines; this code extracts and writes info for chapters, sections, etc.:

```

\chapter
\pchapter
\section 1674 \expandafter\ifx\csname #1\endcsname\chapter
\coffee 1675 \MTC@WriteContentsline{#1}{ptcC}{#3}{#4}%
\subsection 1676 \fi
\subsubsection 1677 \expandafter\ifx\csname #1\endcsname\pchapter
\paragraph 1678 \MTC@WriteContentsline{#1}{ptcC}{#3}{#4}%
\subparagraph 1679 \fi
1680 \expandafter\ifx\csname #1\endcsname\section
1681 \MTC@WriteContentsline{#1}{ptcS}{#3}{#4}%
1682 \fi
1683 \expandafter\ifx\csname #1\endcsname\coffee
1684 \MTC@WriteCoffeeline{#1}{#3}%
1685 \fi
1686 \expandafter\ifx\csname #1\endcsname\subsection
1687 \MTC@WriteContentsline{#1}{ptcSS}{#3}{#4}%
1688 \fi
1689 \expandafter\ifx\csname #1\endcsname\subsubsection
1690 \MTC@WriteContentsline{#1}{ptcSSS}{#3}{#4}%
1691 \fi
1692 \expandafter\ifx\csname #1\endcsname\paragraph
1693 \MTC@WriteContentsline{#1}{ptcP}{#3}{#4}%
1694 \fi
1695 \expandafter\ifx\csname #1\endcsname\subparagraph
1696 \MTC@WriteContentsline{#1}{ptcSP}{#3}{#4}%
1697 \fi

```

\MTC@WriteContentsline And for the starred sectionning commands:

```

\starchapter
\starsection 1698 \expandafter\ifx\csname #1\endcsname\starchapter
\starsubsection 1699 %%HO: the following line should be disabled: \stepcounter{ptc}%
\starsubsubsection 1700 \MTC@WriteContentsline{#1}{ptcC}{#3}{#4}%
\starparagraph 1701 \fi
\starsubparagraph 1702 \expandafter\ifx\csname #1\endcsname\starsection
1703 \MTC@WriteContentsline{#1}{ptcS}{#3}{#4}%
1704 \fi
1705 \expandafter\ifx\csname #1\endcsname\starsubsection
1706 \MTC@WriteContentsline{#1}{ptcSS}{#3}{#4}%
1707 \fi

```

```

1708 \expandafter\ifx\csname #1\endcsname\starsubsubsection
1709 \MTC@WriteContentsline{#1}{ptcSSS}{#3}{#4}%
1710 \fi
1711 \expandafter\ifx\csname #1\endcsname\starparagraph
1712 \MTC@WriteContentsline{#1}{ptcP}{#3}{#4}%
1713 \fi
1714 \expandafter\ifx\csname #1\endcsname\starsubparagraph
1715 \MTC@WriteContentsline{#1}{ptcSP}{#3}{#4}%
1716 \fi
1717 }

```

\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@next
 \PTC@list

```

1718 \def\PTC@explist{\expandafter\PTC@next\PTC@list\}

```

\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@toc
 \PTC@read

```

1719 \def\PTC@loop#1{\openin\@inputcheck#1\relax
1720 \ifeof\@inputcheck
1721 \mtcPackageWarning[<W0013>]{minitoc}%
1722 {No file #1\MessageBreak PARTTOCS NOT PREPARED}%
1723 \expandafter\PTC@toc
1724 \else
1725 \mtcPackageInfo[<I0025>]{minitoc}%
1726 {PREPARING PARTTOCS FROM #1}%
1727 \expandafter\PTC@read\fi}

```

\PTC@read Read the next entry of the .toc file.

\PTC@line

```

1728 \def\PTC@read{%
1729 \read\@inputcheck to\PTC@line

```

\PTC@test The make sure that \PTC@test has enough arguments:

\PTC@line

```

1730 \expandafter\PTC@test\PTC@line.....\PTC@% %%H0: . added
1731 }%

```

\PTC@test The \PTC@test macro finds the “interesting” commands in the TOC file, mainly to delimit parts:

```

1732 %%H0/BJ: now patch \PTC@test,
1733 %%H0/BJ: call \PTC@contentsline with 4 instead of 3 parameters

```

```

\PTC@contentsline Look at the first token of the line. If it is an interesting entry, process it. If it is \@input,
\mtc@string add the file to the list. Otherwise ignore. Go around the loop if not at end of file. Finally
\PTC@list process the next file in the list.
\PTC@toc
\PTC@read 1734 \long\def\PTC@test#1#2#3#4#5#6\PTC@{% %%HO: #6 added
\partend 1735 \ifx#1\contentsline
1736 \let\mtc@string\string
1737 \PTC@contentsline{#2}{#3}{#4}{#5}%
1738 %%HO/DV: 4 instead of 3 parameters
1739 \let\mtc@string\relax
1740 \else\ifx#1\@input
1741 \edef\PTC@list{\PTC@list#2\relax}%
1742 \else\ifx#1\partend
1743 \immediate\closeout\tf@mtc
1744 \immediate\openout\tf@mtc=\jobname.mtc
1745 \else\ifx#1\partbegin
1746 \addtocounter{ptc}{-1}%
1747 \fi\fi\fi\fi
1748 \ifeof\@inputcheck\expandafter\PTC@toc
1749 \else\expandafter\PTC@read\fi}%

```

9.52.2 Processing macros for the partlofs

```

\PLF@next Processing the next entry in the list and remove it from the head of the list:
\PLF@list
\PLF@loop 1750 \def\PLF@next#1\relax#2\{\%
1751 \edef\PLF@list{#2}%
1752 \PLF@loop{#1}}

```

```

\PLF@lof Check if the list is empty:
\PLF@list
\PLF@explist 1753 \def\PLF@lof{%
1754 \ifx\PLF@list\@empty\else\expandafter\PLF@explist\fi}

```

```

\PLF@contentsline The macro \PLF@contentsline analyses the lines read from the LOF file and detects
\part interesting keywords. If \part is found, the ptc counter is incremented and a new partlof
\theptc file is created.
\tf@mtc
\plfname 1755 \def\PLF@contentsline#1#2#3#4{% %%HO: #4 added
\PLF@WriteContentsLine 1756 \expandafter\ifx\csname #1\endcsname\xpart
1757 \stepcounter{ptc}%

```

`\if@longextensions@` We test if long or short extensions are used, to build the name of the mini-table file, then
`\plfname` open it:

```

1758 \if@longextensions@%
1759 \mtcPackageInfo[<I0033>]{minitoc}%
1760 {Writing\space\jobname.plf\theptc}%
1761 \def\plfname{\jobname.plf\theptc}%
1762 \else
1763 \mtcPackageInfo[<I0033>]{minitoc}%
1764 {Writing\space\jobname.G\theptc}%
1765 \def\plfname{\jobname.G\theptc}%
1766 \fi
1767 \immediate\closeout\tf@mtc
1768 \immediate\openout\tf@mtc=\plfname
1769 \fi

```

`\mtc@toks` The token register `\mtc@toks` is used to pass the entry to `\MTC@WriteContentsline`:

`\figure` Now, we filter the relevant contents lines:

```

\subfigure
\mtc@toks 1770 \expandafter\ifx\csname #1\endcsname\figure
\MTC@WriteContentsline 1771 \mtc@toks{\noexpand\leavevmode#2}%
1772 \MTC@WriteContentsline{#1}{plf}{#3}{#4}%
1773 \fi
1774 \expandafter\ifx\csname #1\endcsname\subfigure
1775 \mtc@toks{\noexpand\leavevmode#2}%
1776 \MTC@WriteContentsline{#1}{plfS}{#3}{#4}%
1777 \fi
1778 }

```

`\PLF@explist` The loop to read the lines of the LOF file; expands the list of entries and call `\PLF@next`

`\PLF@next` to process the first one:

`\PLF@list`

```

1779 \def\PLF@explist{\expandafter\PLF@next\PLF@list\}

```

`\PLF@loop` If an entry is found, loop through line by line, looking for interesting entries. Otherwise,
`\PLF@lof` process the next entry in the list.

`\PLF@read`

```

1780 \def\PLF@loop#1{\openin\@inputcheck#1\relax
1781 \ifeof\@inputcheck
1782 \mtcPackageWarning[<W0011>]{minitoc}%
1783 {No file #1\MessageBreak PARTLOFS NOT PREPARED}%
1784 \expandafter\PLF@lof
1785 \else
1786 \mtcPackageInfo[<I0035>]{minitoc}%
1787 {PREPARING PARTLOFS FROM #1}%
1788 \expandafter\PLF@read\fi}

```

\PLF@read Read the next entry of the .lof file.

```
\PLF@line
1789 \def\PLF@read{%
1790   \read\inputcheck to\PLF@line
```

\PLF@test The make sure that \PLF@test has enough arguments:

```
\PLF@line
1791   \expandafter\PLF@test\PLF@line....\PLF@% %%H0: . added
1792 }%
```

\PLF@test The \PLF@test macro finds the “interesting” commands in the LOF file, mainly to delimit parts:

```
1793 %%H0/BJ: now patch \PLF@test,
1794 %%H0/BJ: call \PLF@contentsline with 4 instead of 3 parameters
```

\PLF@contentsline 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

\mtc@string process the next file in the list.

\PLF@list

\PLF@lof

```
\PLF@read 1795 \long\def\PLF@test#1#2#3#4#5#6\PLF@{% %%H0: #6 added
\partend 1796   \ifx#1\contentsline
1797     \let\mtc@string\string
1798     \PLF@contentsline{#2}{#3}{#4}{#5}% %%H0: #4 added
1799     \let\mtc@string\relax
1800   \else\ifx#1\@input
1801     \edef\PLF@list{\PLF@list#2\relax}%
1802   \else\ifx#1\partend
1803     \immediate\closeout\tf@mtc
1804     \immediate\openout\tf@mtc=\jobname.mtc
1805   \else\ifx#1\partbegin
1806     \addtocounter{ptc}{-1}%
1807   \fi\fi\fi\fi
1808   \ifeof\@inputcheck\expandafter\PLF@lof
1809   \else\expandafter\PLF@read\fi}%
```

9.52.3 Processing macros for the partlots

\PLT@next Processing the next entry in the list and remove it from the head of the list:

\PLT@list

```
\PLT@loop 1810 \def\PLT@next#1\relax#2\{\%
1811   \edef\PLT@list{#2}%
1812   \PLT@loop{#1}}
```

```

\PLT@lot    Check if the list is empty:
\PLT@list
\PLT@explist 1813 \def\PLT@lot{%
              1814   \ifx\PLT@list\@empty\else\expandafter\PLT@explist\fi}

```

```

\PLT@contentsline The macro \PLT@contentsline analyses the lines read from the LOT file and detects
                  \part interesting keywords. If \part is found, the ptc counter is incremented and a new partlot
                  \theptc file is created.
                  \tf@mtc
                  \plname 1815 \def\PLT@contentsline#1#2#3#4{% %%HO: #4 added
\PLT@WriteContentsLine 1816   \expandafter\ifx\csname #1\endcsname\xpart
                  1817     \stepcounter{ptc}%

```

```

\if@longextensions@ We test if long or short extensions are used, to build the name of the mini-table file, then
\plname open it:

```

```

1818   \if@longextensions@%
1819     \mtcPackageInfo[<I0033>]{minitoc}%
1820     {Writing\space\jobname.plt\theptc}%
1821     \def\plname{\jobname.plt\theptc}%
1822   \else
1823     \mtcPackageInfo[<I0033>]{minitoc}%
1824     {Writing\space\jobname.U\theptc}%
1825     \def\plname{\jobname.U\theptc}%
1826   \fi
1827   \immediate\closeout\tf@mtc
1828   \immediate\openout\tf@mtc=\plname
1829   \fi

```

\mtc@toks The token register \mtc@toks is used to pass the entry to \MTC@WriteContentsline:

```

\table Now, we filter the relevant contents lines:
\subtable
\mtc@toks 1830 \expandafter\ifx\csname #1\endcsname\table
\MTC@WriteContentsline 1831 \mtc@toks{\noexpand\leavevmode#2}%
1832 \MTC@WriteContentsline{#1}{plt}{#3}{#4}%
1833 \fi
1834 \expandafter\ifx\csname #1\endcsname\subtable
1835 \mtc@toks{\noexpand\leavevmode#2}%
1836 \MTC@WriteContentsline{#1}{pltS}{#3}{#4}%
1837 \fi
1838 }

```

\PLT@explist The loop to read the lines of the LOT file; expands the list of entries and call \PLT@next to process the first one:
 \PLT@next
 \PLT@list

```
1839 \def\PLT@explist{\expandafter\PLT@next\PLT@list\}
```

\PLT@loop If an entry is found, loop through line by line, looking for interesting entries. Otherwise, process the next entry in the list.
 \PLT@lot
 \PLT@read

```
1840 \def\PLT@loop#1{\openin\@inputcheck#1\relax
1841   \ifeof\@inputcheck
1842     \mtcPackageWarning[<W0012>]{minitoc}%
1843     {No file #1\MessageBreak PARTLOTS NOT PREPARED}%
1844     \expandafter\PLT@lot
1845   \else
1846     \mtcPackageInfo[<I0038>]{minitoc}%
1847     {PREPARING PARTLOTS FROM #1}%
1848     \expandafter\PLT@read\fi}
```

\PLT@read Read the next entry of the .lot file.
 \PLT@line

```
1849 \def\PLT@read{%
1850   \read\@inputcheck to\PLT@line
```

\PLT@test The make sure that \PLT@test has enough arguments:
 \PLT@line

```
1851   \expandafter\PLT@test\PLT@line.....\PLT@% %%H0: . added
1852   }%
```

\PLT@test The \PLT@test macro finds the “interesting” commands in the LOT file, mainly to delimit parts:

```
1853 %%H0/BJ: now patch \PLT@test,
1854 %%H0/BJ: call \PLT@contentsline with 4 instead of 3 parameters
```

\PLT@contentsline 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@string
 \PLT@list
 \PLT@lot

```
\PLT@read 1855 \long\def\PLT@test#1#2#3#4#5#6\PLT@{% %%H0: #6 added
\partend 1856   \ifx#1\contentsline
1857     \let\mtc@string\string
1858     \PLT@contentsline{#2}{#3}{#4}{#5}% %%H0: #4 added
1859     \let\mtc@string\relax
```



```

1860 \else\ifx#1\@input
1861     \edef\PLT@list{\PLT@list#2\relax}%
1862 \else\ifx#1\partend
1863     \immediate\closeout\tf@mtc
1864     \immediate\openout\tf@mtc=\jobname.mtc
1865 \else\ifx#1\partbegin
1866     \addtocounter{ptc}{-1}%
1867 \fi\fi\fi\fi
1868 \ifeof\@inputcheck\expandafter\PLT@lot
1869 \else\expandafter\PLT@read\fi}%

```

End of the part level stuff (begun in section 9.45 on page 289):

```

1870 }%

```

9.53 Depth counters for sectlofs and sectlots

<pre> \AtBeginDocument \c@lofdepth \c@lotdepth \newcounter \setcounter </pre>	<p>If the counters <code>lofdepth</code> and <code>lotdepth</code> are defined, we create new counters for the depths of the corresponding mini-tables: <code>sectlofdepth</code> and <code>sectlotdepth</code>. These counters are initialised to 2. This is done after the loading of the packages, in an <code>\AtBeginDocument</code> block:</p> <pre> 1871 \AtBeginDocument{% 1872 \@ifundefined{c@lofdepth}{}% 1873 {\newcounter{sectlofdepth}\setcounter{sectlofdepth}{2}}% 1874 \@ifundefined{c@lotdepth}{}% 1875 {\newcounter{sectlotdepth}\setcounter{sectlotdepth}{2}}% 1876 } </pre>
---	--

9.54 Section-level commands

<pre> \if@mtc@chapter@undef@ \if@mtc@section@def@ </pre>	<p>The section-level commands are defined only if <code>\chapter</code> is <i>not</i> defined, hence in article-like document classes, and only if <code>\section</code> is defined:</p>
--	--

```

1877 \if@mtc@chapter@undef@
1878 \if@mtc@section@def@

```

<pre> \firstsectionis \adjuststc \decrementstc \incrementstc \stc@rule \stcindent \columnwidth </pre>	<p>We define the obsolete command <code>\firstsectionis</code> (with its harmless warning), the counter <code>stc</code> of <code>secttocs</code>, the <code>\adjuststc</code>, <code>\decrementstc</code> and <code>\incrementstc</code> commands, the depth counter <code>secttocdepth</code> and its default value 2 (to include at least the subsections), the horizontal rule <code>\stc@rule</code> (rule before/after <code>secttoc</code>/<code>sectlof</code>/<code>sectlot</code>), the indentation (both sides) <code>\stcindent</code> for the <code>secttocs</code> (with its default values).</p>
---	---

```

1879 \def\firstsectionis#1%
1880   {\mtcPackageWarning[<W0005>]{minitoc}%
1881     {\string\firstsectionis \space is an obsolete\MessageBreak command}%
1882     \@firstsectionis@used@true}
1883 \newcounter{stc}\setcounter{stc}{0}%
1884 \newcommand{\adjuststc}[1][1]{\addtocounter{stc}{#1}}%
1885 \def\decrementstc{\addtocounter{stc}{-1}}%
1886 \def\incrementstc{\addtocounter{stc}{+1}}%
1887 \newcounter{secttocdepth}\setcounter{secttocdepth}{2}%
1888 \def\stc@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}%
1889 \newlength\stcindent \stcindent=24\p@

```

9.55 Fonts commands for secttocs and co.

```

\stcfont  We define the fonts commands for the secttocs, sectlofs and sectlots and their titles:
\stcSSfont
\stcSSSfont 1890 \def\stcfont{\small\rmfamily\upshape\mdseries} % secttoc
\stcPfont 1891 \def\stcSSfont{\small\rmfamily\upshape\bfseries} % (subsections)
\stcSPfont 1892 \let\stcSSfont\stcfont % (subsubsections)
\slffont 1893 \let\stcPfont\stcfont % (paragraphs)
\slfSfont 1894 \let\stcSPfont\stcfont % (subparagraphs)
\sltfont 1895 \let\slffont\stcfont % sectlof (figures)
\sltSfont 1896 \let\slfSfont\stcfont % sectlof (subfigures)
\stifont 1897 \let\sltfont\stcfont % sectlot (tables)
1898 \let\sltSfont\stcfont % sectlot (subtables)
1899 \def\stifont{\large\rmfamily\upshape\bfseries} % titles

```

9.56 Internal macros for title positionning

Some internal macros for title positionning, from the optional arguments of `\dosecttoc` and `\secttoc` commands (and siblings):

```

\l@sti Centering, flushleft, flushright or empty titles (with a vertical correction for empty titles,
\c@sti from Frank MITTELBACH):
\r@sti
\e@sti 1900 \def\c@sti#1{\null\hfill #1\hfill\null}
\n@sti 1901 \def\l@sti#1{\null #1\hfill\null}
1902 \def\r@sti#1{\null\hfill #1\null}
1903 \def\e@sti#1{\vspace{-\baselineskip}}
1904 \def\n@sti#1{\vspace{-\baselineskip}}

```

```

\do@stitle By default, titles are flushleft.
\df@stitle
\do@stilf 1905 \let\do@stitle\l@sti
\df@stilf 1906 \let\df@stitle\l@sti
\do@stilt 1907 \let\do@stilf\l@sti
\df@stilt 1908 \let\df@stilf\l@sti
\l@sti 1909 \let\do@stilt\l@sti
1910 \let\df@stilt\l@sti

```

9.57 The stc@verse environment

`stc@verse` The `stc@verse` environment is a very simple list environment, analog to the standard `verse` environment. Some formatting parameters are adjusted. The `tight/loose` and `ifktightmtc` `k-tight/k-loose` package options are honored.

```

1911 \def\stc@verse{\let\=\@centercr
1912 \list{ }\itemsep\z@
1913 \itemindent \z@
1914 \listparindent \itemindent
1915 \partopsep \z@
1916 \iftightmtc \parsep \z@ \fi
1917 \ifktightmtc \parskip \z@ \fi
1918 \topsep 1ex %% 20060731 was \z@
1919 \leftmargin\stcindent
1920 \rightmargin\leftmargin\item{ }
1921 \def\endstc@verse{\nopagebreak[4]\endlist}

```

9.58 The \secttoc, \sectlof, and \sectlot commands

These three commands are very similar.

9.58.1 The \secttoc command

`\secttoc` The `\secttoc` command must be used after `\section` if you need a `secttoc` (no automatic `secttoc`). Its code is similar to the code of `\minitoc` (but simpler). First, `\secttoc` detects the presence of its optional argument, and uses its default value, `d`, if it is missing. Then, `\secttoc@` is called with the effective position as argument:

```

1922 \def\secttoc{\@ifnextchar[{\secttoc@}{\secttoc@d}}

```

```

\secttoc@ The \secttoc@ macro does the real work. It first sets the flag \if@secttoc@used@ (for
\if@secttoc@used@ a coherence hint) and checks if long extensions are used or not (to create the name of the
\if@longextensions@ secttoc file):
\@tocfile
\thestc 1923 \def\secttoc@[#1]{%
1924 \global\@secttoc@used@true
1925 \if@longextensions@%
1926 \def\@tocfile{stc\thestc}%
1927 \else
1928 \def\@tocfile{S\thestc}%
1929 \fi

\mtc@CkFile Then, we check the presence and the emptiness of the secttoc file and give a warning if it
\if@mtc@FE is not here or is empty:
\@tocfile
1930 \mtc@CkFile{\jobname.\@tocfile}
1931 \if@mtc@FE
1932 \mtcPackageInfo[<I0006>]{minitoc}%
1933 {\jobname.\@tocfile\space is empty}
1934 \@mtc@empty@secttoc@true
1935 \else

\beforesecttoc If the secttoc file is present and not empty, we can insert it, but we must add some
\thispagesecttocstyle presentation code: first, \beforesecttoc, of course, and the page style feature:

1936 \beforesecttoc
1937 \thispagesecttocstyle

\do@stitle We begin a samepage environment, then treat the positionning argument. If the title is
\@stitle empty, we simulate the “e” positionning.
\n@stitle
\c@stitle 1938 %% \mtc@markboth{\MakeUppercase{\stctitle}}{\MakeUppercase{\stctitle}}%
\l@stitle 1939 \relax\begin{samepage}%
\r@stitle 1940 \if #1e\let\do@stitle\@stitle
\df@stitle 1941 \else\if #1n\let\do@stitle\n@stitle
\mtc@CkStr 1942 \else\if #1c\let\do@stitle\c@stitle
\stctitle 1943 \else\if #1l\let\do@stitle\l@stitle
\if@mtc@FE 1944 \else\if #1r\let\do@stitle\r@stitle
samepage 1945 \else\if #1d\let\do@stitle\df@stitle
1946 \fi\fi\fi\fi\fi
1947 \mtc@CkStr{\stctitle}\if@mtc@FE \let\do@stitle\@stitle\relax\fi

\raggedright We adjust some formatting parameters and avoid a page break between the title and the
\parskip secttoc, then we set the font:
\stcfont

```

```

1948      \raggedright
1949      \parskip=\z@%
1950      \reset@font\stcfont%
1951      \parindent=\z@%
1952      \nopagebreak[4]%

```

`\stc@rule` The secttoc 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 secttoc.

```

\stifont
\do@sttc 1953      \kern-0.8\baselineskip\nopagebreak[4]%
      \mtc@v 1954      \par\noindent
\stctitle 1955      \nopagebreak[4]%
      \hline 1956      \ifx\stc@rule\relax
1957      \begin{tabular}{@{}p{\columnwidth}@{}}
1958      \reset@font\stifont\do@sttc{\mtc@v\stctitle}\\
1959      \end{tabular}%
1960      \else
1961      \begin{tabular}{@{}p{\columnwidth}@{}}
1962      \reset@font\stifont\do@sttc{\mtc@v\stctitle}\\\hline
1963      \end{tabular}%
1964      \fi

```

`\mtc@zrule` Then, we adjust the position under the top rule and set the indentation and some formatting parameters:

```

\mtc@BBR
\stcindent
1965      \nopagebreak[4]\null\leavevmode\mtc@zrule\\\mtc@BBR
1966      \leftmargin\stcindent
1967      \rightmargin\stcindent
1968      \itemindent=\z@\labelwidth=\z@%
1969      \labelsep=\z@\listparindent=\z@%

```

`stc@verse` We enter in a `stc@verse` environment to format the secttoc. The toc depth is forced (locally) to `secttocdepth`. A little trick is necessary to adjust the position.

```

\c@tocdepth
\c@secttocdepth
      \mtc@BBR 1970      \begin{stc@verse}\c@tocdepth=\c@secttocdepth%
1971      \leavevmode\\\mtc@BBR\vskip -.5\baselineskip

```

`\stc@pgno` 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 `\@undottedtocline` `\mtcsetformat` are activated via `\stc@setform`. Then the secttoc file is inserted, followed by a strut, and the `stc@verse` environment is terminated.

```

\stc@setform
      \@tocfile 1972 \begingroup
      \mtc@strut 1973 \makeatletter
      stc@verse 1974 \stc@setform%

```

```

1975 \@ifundefined{stc@pgno}%
1976 {\let\@dottedtocline\@undottedtocline}{}
1977 \@fileswfalse\mtc@hook@beforeinputfile
1978 \stc@setform%
1979 \@input{\jobname.\@tocfile}
1980 \vspace{-1ex} \vspace{-\baselineskip}
1981 \leavevmode\mtc@strut
1982 \global\@nobreakfalse\endgroup
1983 \end{stc@verse}%

```

```

\stc@rule The final part is just to add the bottom rule, if necessary, a possible page break and
\mtc@zrule \aftersecttoc.
samepage
\aftersecttoc 1984 \kernaftersecttoc
1985 \nopagebreak[4]\stc@rule\null\leavevmode\\%
1986 \vskip-1.0\baselineskip\mtc@zrule\end{samepage}%
1987 \par\pagebreak[1]\vspace*{-1ex}\aftersecttoc\fi}%

```

9.58.2 The \sectlof command

\sectlof The \sectlof command must be used after \section if you need a sectlof (no automatic
 \sectlof@ sectlof). Its code is similar to the code of \minilof (but simpler). First, \sectlof
 detects the presence of its optional argument, and uses its default value, d, if it is missing.
 Then, \sectlof@ is called with the effective position as argument:

```

1988 \def\sectlof{\@ifnextchar[{\sectlof@}{\sectlof@d}}

```

\sectlof@ The \sectlof@ macro does the real work. It first sets the flag \if@sectlof@used@ (for
 \if@sectlof@used@ a coherence hint) and checks if long extensions are used or not (to create the name of the
 \if@longextensions@ sectlof file):

```

\@tocfile
\thestc 1989 \def\sectlof@[#1]{%
1990 \global\@sectlof@used@true
1991 \if@longextensions@
1992 \def\@tocfile{slf\thestc}%
1993 \else
1994 \def\@tocfile{H\thestc}%
1995 \fi

```

\mtc@CkFile Then, we check the presence and the emptiness of the sectlof file and give a warning if it
 \if@mtc@FE is not here or is empty:

```

\@tocfile
1996 \mtc@CkFile{\jobname.\@tocfile}
1997 \if@mtc@FE

```

```

1998      \mtcPackageInfo[<I0006>]{minitoc}%
1999          {\jobname.\@tocfile\space is empty}
2000      \@mtc@empty@sectlof@true
2001      \else

```

`\beforesectlof` 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:

```

2002      \thispagesectlofstyle
2003 %%      \mtc@markboth{\MakeUppercase{\slftitle}}{\MakeUppercase{\slftitle}}%
2004      \beforesectlof

```

`\do@stilf` We begin a `samepage` environment, then treat the positioning argument. If the title is empty, we simulate the “e” positioning.

```

\do@stilf
\@e@sti
\@n@sti
\@c@sti 2005      \relax\begin{samepage}%
\@l@sti 2006      \if #1e\let\do@stilf\@e@sti
\@r@sti 2007      \else\if #1n\let\do@stilf\@n@sti
\@df@sti 2008      \else\if #1c\let\do@stilf\@c@sti
\mtc@CkStr 2009      \else\if #1l\let\do@stilf\@l@sti
\slftitle 2010      \else\if #1r\let\do@stilf\@r@sti
\if@mtc@FE 2011      \else\if #1d\let\do@stilf\@df@stilf
2012      \fi\fi\fi\fi\fi
samepage 2013      \mtc@CkStr{\slftitle}\if@mtc@FE \let\do@stilf\@e@sti\relax\fi

```

`\raggedright` We adjust some formatting parameters and avoid a page break between the title and the sectlof, then we set the font:

```

\parskip
\slffont
2014      \raggedright
2015      \parskip=\z@%
2016      \reset@font\slffont%
2017      \parindent=\z@%
2018      \nopagebreak[4]%

```

`\slf@rule` The sectlof 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 sectlof.

```

\columnwidth
\do@stilf 2019      \kern-0.8\baselineskip\nopagebreak[4]%
\mtc@v 2020      \par\noindent
\slftitle 2021      \ifx\slf@rule\relax
\hline 2022      \begin{tabular}{@{}p{\columnwidth}@{}}
2023      \reset@font\stifont\do@stilf{\mtc@v\slftitle}\\
2024      \end{tabular}%
2025      \else
2026      \begin{tabular}{@{}p{\columnwidth}@{}}

```

```

2027      \mtc@hstrut
2028      \reset@font\stifont\do@stilf{\mtc@v\slftitle}\hline
2029      \end{tabular}%
2030      \fi

```

\mtc@zrule Then, we adjust the position under the top rule and set the indentation and some
 \mtc@BBR formatting parameters:

```

\stcindent
2031      \nopagebreak[4]\null\leavevmode\mtc@zrule\\\mtc@BBR
2032      \leftmargin\stcindent
2033      \rightmargin\stcindent
2034      \itemindent=\z@\labelwidth=\z@%
2035      \labelsep=\z@\listparindent=\z@%

```

stc@verse We enter in a stc@verse environment to format the sectlof. The toc depth is forced
 \c@tocdepth (locally) to sectlofdepth. A little trick is necessary to adjust the position.
 \c@sectlofdepth

```

\mtc@BBR 2036      \begin{stc@verse}%
2037      \ifundefined{c@lofdepth}%
2038      {}%
2039      {\c@lofdepth=\c@sectlofdepth
2040      \ifnum\c@lofdepth<1\relax\c@lofdepth=1\fi}
2041      \leavevmode\\\mtc@BBR\vskip -.5\baselineskip

```

\slf@pgno If the contents lines must have no numbers, we replace the macro \@dottedtocline
 \@dottedtocline with its undotted version. A hook is added, and the formatting settings coming from
 \@undottedtocline \mtcsetformat are activated via \slf@setform. Then the sectlof file is inserted,
 \mtc@hook@beforeinputfile followed by a strut, and the stc@verse environment is terminated.

```

\slf@setform
  \@tocfile 2042 \begingroup
\mtc@strut 2043 \makeatletter
stc@verse 2044 \ifundefined{slf@pgno}%
2045 {\let\@dottedtocline\@undottedtocline}{}
2046 \@filesfalse\mtc@hook@beforeinputfile
2047 \slf@setform%
2048 \@input{\jobname.\@tocfile}
2049 \global\@nbreakfalse\endgroup
2050 \end{stc@verse}%

```

\stc@rule The final part is just to add the bottom rule, if necessary, a possible page break and
 \mtc@zrule \aftersectlof. The blank line (\) is essential.

```

samepage
\aftersectlof 2051 \kernaftersectlof
2052 \nopagebreak[4]\slf@rule\null\leavevmode\\%
2053 \vskip-1.0\baselineskip\mtc@zrule\end{samepage}%
2054 \par\pagebreak[1]\vspace*{-1ex}\aftersectlof\fi}%

```


9.58.3 The `\sectlot` command

`\sectlot` 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:

```
2055 \def\sectlot{\@ifnextchar[{\sectlot@}{\sectlot@d}}
```

`\sectlot@` The `\sectlot@` macro does the real work. It first sets the flag `\if@sectlot@used@` (for a coherence hint) and checks if long extensions are used or not (to create the name of the sectlot file):

```
\@tocfile
\thetc 2056 \def\sectlot@[#1]{%
2057 \global\@sectlot@used@true
2058 \if@longextensions@%
2059 \def\@tocfile{slt\thetc}%
2060 \else
2061 \def\@tocfile{I\thetc}%
2062 \fi
```

`\mtc@CkFile` Then, we check the presence and the emptiness of the sectlot file and give a warning if it is not here or is empty:

```
\@tocfile
2063 \mtc@CkFile{\jobname.\@tocfile}
2064 \if@mtc@FE
2065 \mtcPackageInfo[<I0006>]{minitoc}%
2066 {\jobname.\@tocfile\space is empty}
2067 \@mtc@empty@sectlot@true
2068 \else
```

`\beforesectlot` 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:

```
2069 \thispagesectlotstyle
2070 %% \mtc@markboth{\MakeUppercase{\slttitle}}{\MakeUppercase{\slttitle}}%
2071 \beforesectlot
```

```

\do@stilt We begin a samepage environment, then treat the positionning argument. If the title is
  \e@sti empty, we simulate the “e” positionning.
  \n@sti
  \c@sti 2072      \relax\begin{samepage}%
  \l@sti 2073      \if #1e\let\do@stilt\e@sti
  \r@sti 2074      \else\if #1n\let\do@stilt\n@sti
  \df@sti 2075      \else\if #1c\let\do@stilt\c@sti
\mtc@CkStr 2076      \else\if #1l\let\do@stilt\l@sti
\sltttitle 2077      \else\if #1r\let\do@stilt\r@sti
\if@mtc@FE 2078      \else\if #1d\let\do@stilt\df@stilt
  samepage 2079      \fi\fi\fi\fi\fi\fi
  2080      \mtc@CkStr{\sltttitle}\if@mtc@FE \let\do@stilt\e@sti\relax\fi

```

```

\raggedright We adjust some formatting parameters and avoid a page break between the title and the
  \parskip sectlot, then we set the font:
  \sltfont
  2081      \raggedright
  2082      \parskip=\z@%
  2083      \reset@font\sltfont%
  2084      \parindent=\z@%
  2085      \nepagebreak[4]%

```

```

\stc@rule The sectlot title is set in a tabular environment (to inhibit a page break between the title
  tabular and the top rule), with a rule at its bottom if necessary. This rule is an \hline. It is the
  \stifont top rule of the sectlot.
\columnrule
\do@stilt 2086      \kern-0.8\baselineskip\nepagebreak[4]%
  \mtc@v 2087      \par\noindent
\sltttitle 2088      \ifx\slt@rule\relax
  \hline 2089      \begin{tabular}{@{}p{\columnwidth}@{}}
  2090      \reset@font\stifont\do@stilt{\mtc@v\sltttitle}\\
  2091      \end{tabular}%
  2092      \else
  2093      \begin{tabular}{@{}p{\columnwidth}@{}}
  2094      \mtc@hstrut
  2095      \reset@font\stifont\do@stilt{\mtc@v\sltttitle}\\\hline
  2096      \end{tabular}%
  2097      \fi

```

```

\mtc@zrule Then, we adjust the position under the top rule and set the indentation and some
  \mtc@BBR formatting parameters:
\stcindent
  2098      \nepagebreak[4]\null\leavevmode\mtc@zrule\\\mtc@BBR
  2099      \leftmargin\stcindent
  2100      \rightmargin\stcindent
  2101      \itemindent=\z@\labelwidth=\z@%
  2102      \labelsep=\z@\listparindent=\z@%

```

`stc@verse` We enter in a `stc@verse` environment to format the `sectlot`. The `toc` depth is forced
`\c@tocdepth` (locally) to `sectlotdepth`. A little trick is necessary to adjust the position.
`\c@sectlotdepth`

```

\mtc@BBR 2103 \begin{stc@verse}%
          2104 \ifundefined{c@lotdepth}%
          2105 {}%
          2106 {\c@lotdepth=\c@sectlotdepth
          2107 \ifnum\c@lotdepth<1\relax\c@lotdepth=1\fi}
          2108 \leavevmode\\ \mtc@BBR\vskip -.5\baselineskip

```

`\slt@pgno` If the contents lines must have no numbers, we replace the macro `\@dottedtocline`
`\@dottedtocline` with its undotted version. A hook is added, and the formatting settings coming from
`\@undottedtocline` `\mtcsetformat` are activated via `\slt@setform`. Then the `sectlot` file is inserted,
`\mtc@hook@beforeinputfile` followed by a strut, and the `stc@verse` environment is terminated.

```

\slt@setform
\@tocfile 2109 \begingroup
\mtc@strut 2110 \makeatletter
stc@verse 2111 \ifundefined{slt@pgno}%
          2112 {\let\@dottedtocline\@undottedtocline}{}
          2113 \gdef\thestc{\arabic{stc}}
          2114 \@filesfalse\mtc@hook@beforeinputfile
          2115 \slt@setform%
          2116 \input{\jobname.\@tocfile}
          2117 \global\@nobeckfalse\endgroup
          2118 \end{stc@verse}%

```

`\stc@rule` The final part is just to add the bottom rule, if necessary, a possible page break and
`\mtc@zrule` `\aftersectlot`.
`samepage`

```

\aftersectlot 2119 \kernaftersectlof
          2120 \nopagebreak[4]\slt@rule\null\leavevmode\\%
          2121 \vskip-1.0\baselineskip\mtc@zrule\end{samepage}%
          2122 \par\pagebreak[1]\vspace*{-1ex}\aftersectlot\fi}%

```

9.59 Auxiliary internal commands, section level

`\l@xsect` We define auxiliary commands, used for the mini-tables and as delimiters in the TOC
`\l@xsection` file (and LOF and LOT files). The depth of `xsect` is huge to inhibit the printing of its
`\@dottedtocline` contents line (except if you cheat).

```

\l@schapter
\l@xsect 2123 \def\l@xsect{\@dottedtocline{\@M}{1.0em}{2.3em}}
\l@xsection 2124 \def\l@xsection{\@dottedtocline{\@M}{1.0em}{2.3em}}
\l@schapter 2125 \def\l@schapter{\@dottedtocline{1}{1.0em}{2.3em}}
          2126 \def\xsect{xsect}
          2127 \def\xsection{xsection}
          2128 \def\schapter{schapter}

```

9.60 Patching the \section command (continued)

\@sect 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).

```
2129 \let\sv@sect\@sect
2130 \gdef\@sect#1#2#3#4#5#6[#7]#8{%
2131   \ifnum #2=1\relax
2132     \addcontentsline{lof}{xsect}{#7}%
2133     \addcontentsline{lot}{xsect}{#7}%
2134   \fi
2135   \sv@sect{#1}{#2}{#3}{#4}{#5}{#6}[#7]{#8}}
```

\section If it is a section (unstarred or starred via \starsection), we add a xsect entry in the
\starsection LOF and in the LOT.
\addcontentsline

```
2136 \def\@sect#1#2#3#4#5#6[#7]#8{
2137   \expandafter
2138   \ifx\csname #1\endcsname\section\relax
2139     \addcontentsline{lof}{xsect}{#7}%
2140     \addcontentsline{lot}{xsect}{#7}%
2141   \fi
2142   \ifx\csname #1\endcsname\starsection\relax
2143     \addcontentsline{lof}{xsect}{#7}%
2144     \addcontentsline{lot}{xsect}{#7}%
2145   \fi
```

\@svsec And the remainder of the section header formatting:
\refstepcounter
\@tempskipa 2146 \ifnum #2>\c@secnumdepth\relax
\@hangfrom 2147 \let\@svsec\@empty
\addcontentsline 2148 \else
\numberline 2149 \refstepcounter{#1}%
\@svsechd 2150 \edef\@svsec{\csname the#1\endcsname\hskip 1em}%
\@xsect 2151 \fi
2152 \@tempskipa #5\relax
2153 \ifdim \@tempskipa>z@
2154 \begingroup #6\relax
2155 \@hangfrom{\hskip #3\relax\@svsec}%
2156 {\interlinepenalty \@M #8\par}%
2157 \endgroup
2158 \csname #1mark\endcsname{#7}\addcontentsline
2159 {toc}{#1}{\ifnum #2>\c@secnumdepth\relax
2160 \else
2161 \protect\numberline{\csname the#1\endcsname}%

```

2162          \fi
2163          #7}%
2164 \else
2165   \def\@svsechd{#6\hskip #3\relax
2166   \@svsec #8\csname #1mark\endcsname
2167   {#7}\addcontentsline
2168   {toc}{#1}{\ifnum #2>\c@secnumdepth\relax
2169           \else
2170           \protect\numberline{\csname the#1\endcsname}
2171           \fi
2172   #7}}%
2173 \fi
2174 \@xsect{#5}}

```

\@sect 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.

```

\stc@sect
\addtocontents 2175 \let\stc@sect\@sect
2176 \def\@sect#1#2#3#4#5#6[#7]#8{%
2177   \ifnum #2<1 \relax
2178   \addtocontents{toc}{\protect\sectbegin}
2179   \fi
2180   \stc@sect{#1}{#2}{#3}{#4}{#5}{#6}[#7]{#8}}
2181 \let\sectend\relax
2182 \let\sectbegin\relax

```

9.61 The \dosecttoc command and siblings

The \dosecttoc command is very similar to \dominitoc.

\dosecttoc The \dosecttoc command extracts information from the .toc file and creates the
 \@dosecttoc .stc(N) files (.stc becomes .S on MS-DOS).
 \STC@next

```

2183 \def\@dosecttoc#1{%
2184   \makeatletter
2185   \setcounter{stc}{0}
2186   \STC@next#1.toc\relax\\}\setcounter{stc}{0}}

```

\dosectlof The \dosectlof command extracts information from the .lof file and creates the
 \@dosectlof .slf(N) files (.slf becomes .H on MS-DOS).
 \SLF@next

```

2187 \def\@dosectlof#1{%
2188   \makeatletter
2189   \setcounter{stc}{0}
2190   \SLF@next#1.lof\relax\\}\setcounter{stc}{0}}

```

`\dosectlot` The `\dosectlot` command extracts information from the `.lot` file and creates the
`\@dosectlot` `.slt(N)` files (`.slt` becomes `.V` on MS-DOS).
`\PLT@next`

```
2191 \def\dosectlot#1{%
2192   \makeatletter
2193   \setcounter{stc}{0}
2194   \SLT@next#1.lot\relax\\}\setcounter{stc}{0}}
```

`\dosecttoc` We define the user-level macros, who detect the optional argument:
`\dosectlof`

```
\dosectlot 2195 \def\dosecttoc{\@ifnextchar[{\dosecttoc@}{\dosecttoc@[1]}}
2196 \def\dosectlof{\@ifnextchar[{\dosectlof@}{\dosectlof@[1]}}
2197 \def\dosectlot{\@ifnextchar[{\dosectlot@}{\dosectlot@[1]}}
```

`\dosecttoc@` 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.

`\@mtc@hints@given@true`

```
\if@dosecttoc@used@ 2198 \def\dosecttoc@[#1]{%
\df@stitle 2199 \if@mtc@hints@
\@sti 2200   \if@dosecttoc@used@
\@sti 2201   \mtcPackageInfo[<I0045>]{minitoc(hints)}%
\@sti 2202   {The \string\dosecttoc \space command\MessageBreak
\@sti 2203   has been invoked more than once\MessageBreak}
\@sti 2204   \global\@mtc@hints@given@true
\@sti 2205   \fi
2206 \fi
2207 \global\@dosecttoc@used@true
2208 \if #1e\let\df@stitle\@sti%
2209 \else\if #1n\let\df@stitle\@sti%
2210 \else\if #1c\let\df@stitle\@sti%
2211 \else\if #1l\let\df@stitle\@sti%
2212 \else\if #1r\let\df@stitle\@sti%
2213 \fi\fi\fi\fi\fi%
2214 \@dosecttoc}
```

`\dosectlof@` 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.

`\@mtc@hints@given@true`

```
\if@dosectlof@used@ 2215 \def\dosectlof@[#1]{%
\df@stilf 2216 \if@mtc@hints@
\@sti 2217   \if@dosectlof@used@
\@sti 2218   \mtcPackageInfo[<I0045>]{minitoc(hints)}%
\@sti 2219   {The \string\dosectlof \space command\MessageBreak
\@sti 2220   has been invoked more than once\MessageBreak}
\@sti 2221   \global\@mtc@hints@given@true
\@sti 2222   \fi
2223 \fi
2224 \global\@dosectlof@used@true
```

```

2225 \if #1e\let\df@stilf\e@sti%
2226 \else\if #1n\let\df@stilf\n@sti%
2227 \else\if #1c\let\df@stilf\c@sti%
2228 \else\if #1l\let\df@stilf\l@sti%
2229 \else\if #1r\let\df@stilf\r@sti%
2230 \fi\fi\fi\fi\fi%
2231 \@@dosectlof}

```

\dosectlot@ We treat the optional argument of \dosectlot (it becomes the default position for titles
 \if@mtc@hints@ of sectlofs) and flag this macro as used; a hint detects any spurious invocation.

\@mtc@hints@given@true

```

\if@dosectlot@used@ 2232 \def\dosectlot@[#1]{%
  \df@stilt 2233 \if@mtc@hints@
    \e@sti 2234 \if@dosectlot@used@
    \n@sti 2235 \mtcPackageInfo[<I0045>]{minitoc(hints)}%
    \c@sti 2236 {The \string\dosectlot \space command\MessageBreak
    \l@sti 2237 has been invoked more than once\MessageBreak}
    \r@sti 2238 \global\@mtc@hints@given@true
  2239 \fi
  2240 \fi
  2241 \global\@dosectlot@used@true
  2242 \if #1e\let\df@stilt\e@sti%
  2243 \else\if #1n\let\df@stilt\n@sti%
  2244 \else\if #1c\let\df@stilt\c@sti%
  2245 \else\if #1l\let\df@stilt\l@sti%
  2246 \else\if #1r\let\df@stilt\r@sti%
  2247 \fi\fi\fi\fi\fi%
  2248 \@@dosectlot}

```

\@@dosecttoc These macros invoke the \@dosect... commands to create the mini-table file, then close
 \@@dosectlof the file descriptor.

\@@dosectlot

```

\tf@mtc 2249 \def\@@dosecttoc{\@dosecttoc{\jobname}\immediate\closeout\tf@mtc}
2250 \def\@@dosectlof{\@dosectlof{\jobname}\immediate\closeout\tf@mtc}
2251 \def\@@dosectlot{\@dosectlot{\jobname}\immediate\closeout\tf@mtc}

```

\STC@next Processing the next entry in the list and remove it from the head of the list:

\STC@list

```

\STC@loop 2252 \def\STC@next#1\relax#2\{\%
2253 \edef\STC@list{#2}%
2254 \STC@loop{#1}}

```

\STC@toc Check if the list is empty:

\STC@list

```

\STC@explist 2255 \def\STC@toc{%
2256 \ifx\STC@list\@empty\else\expandafter\STC@explist\fi}

```

```

\STC@contentsline The macro \STC@contentsline analyses the lines read from the TOC file. If \section
\section is found, the stc counter is incremented and a new secttoc file is created.
\thestc
\tf@mtc 2257 \def\STC@contentsline#1#2#3#4{% %%H0: #4 added
\stcname 2258 \gdef\thestc{\arabic{stc}}% %%H0: space removed
\MTC@WriteContentsLine 2259 \expandafter\ifx\csname #1\endcsname\section
2260 \stepcounter{stc}%
2261 %% \def\thestc{\arabic{stc}} %% H0: removed

```

```

\if@longextensions@ We test if long or short extensions are used, to build the name of the mini-table file, then
\stcname open it:

```

```

2262 \if@longextensions@%
2263 \mtcPackageInfo[<I0033>]{minitoc}%
2264 {Writing\space\jobname.stc\thestc}%
2265 \def\stcname{\jobname.stc\thestc}%
2266 \else
2267 \mtcPackageInfo[<I0033>]{minitoc}%
2268 {Writing\space\jobname.S\thestc}%
2269 \def\stcname{\jobname.S\thestc}%
2270 \fi
2271 \immediate\closeout\tf@mtc
2272 \immediate\openout\tf@mtc=\stcname
2273 \fi

```

```

\mtc@toks Now, we filter the relevant contents lines:
\MTC@WriteCoffeeline
\MTC@WriteContentsline 2274 \mtc@toks{\noexpand\leavevmode #2}%
\coffee 2275 \expandafter\ifx\csname #1\endcsname\coffee
\subsection 2276 \MTC@WriteCoffeeline{#1}{#3}%
\subsubsection 2277 \fi
\paragraph 2278 \expandafter\ifx\csname #1\endcsname\subsection
\subparagraph 2279 \MTC@WriteContentsline{#1}{stcSS}{#3}{#4}%
2280 \fi
2281 \expandafter\ifx\csname #1\endcsname\subsubsection
2282 \MTC@WriteContentsline{#1}{stcSSS}{#3}{#4}%
2283 \fi
2284 \expandafter\ifx\csname #1\endcsname\paragraph
2285 \MTC@WriteContentsline{#1}{stcP}{#3}{#4}%
2286 \fi
2287 \expandafter\ifx\csname #1\endcsname\subparagraph
2288 \MTC@WriteContentsline{#1}{stcSP}{#3}{#4}%
2289 \fi

```

```

\starsection A starred section terminates the current section and creates a new secttoc file:
\stepcounter
\thestc 2290 \ifx\csname #1\endcsname\starsection
\arabic 2291 \stepcounter{stc}%
\if@longextensions@
\stcname
\closeout
\openout

```



```

2292 \gdef\thetc{\arabic{stc}}
2293 \if@longextensions@%
2294 \mtcPackageInfo[<I0033>]{minitoc}%
2295 {Writing\space\jobname.stc\thetc}%
2296 \def\stcname{\jobname.stc\thetc}%
2297 \else
2298 \mtcPackageInfo[<I0033>]{minitoc}%
2299 {Writing\space\jobname.S\thetc}%
2300 \def\stcname{\jobname.S\thetc}%
2301 \fi
2302 \immediate\closeout\tf@mtc
2303 \immediate\openout\tf@mtc=\stcname
2304 \fi

```

\MTC@WriteContentsline We process the entries for starred sectionning commands:

```

\starsubsection
\starsubsubsection 2305 \expandafter\ifx\csname #1\endcsname\starsubsection
\starparagraph 2306 \MTC@WriteContentsline{#1}{stcSS}{#3}{#4}%
\starsubparagraph 2307 \fi
2308 \expandafter\ifx\csname #1\endcsname\starsubsubsection
2309 \MTC@WriteContentsline{#1}{stcSSS}{#3}{#4}%
2310 \fi
2311 \expandafter\ifx\csname #1\endcsname\starparagraph
2312 \MTC@WriteContentsline{#1}{stcP}{#3}{#4}%
2313 \fi
2314 \expandafter\ifx\csname #1\endcsname\starsubparagraph
2315 \MTC@WriteContentsline{#1}{stcSP}{#3}{#4}%
2316 \fi
2317 }

```

\STC@explist The loop to read the lines of the TOC file; expands the list of entries and call \STC@next to process the first one.

```

\STC@next
\STC@list
2318 \def\STC@explist{\expandafter\STC@next\STC@list\}

```

\STC@loop If an entry is found, loop through line by line, looking for interesting entries. Otherwise, process the next entry in the list.

```

\STC@read
2319 \def\STC@loop#1{\openin\@inputcheck#1\relax
2320 \ifeof\@inputcheck
2321 \mtcPackageWarning[<W0016>]{minitoc}%
2322 {No file #1\MessageBreak SECTOCS NOT PREPARED}%
2323 \expandafter\STC@toc
2324 \else
2325 \mtcPackageInfo[<I0026>]{minitoc}%
2326 {PREPARING SECTOCS FROM #1}%
2327 \expandafter\STC@read\fi}

```

\STC@read Read the next entry of the .toc file.

```
\STC@line
2328 \def\STC@read{%
2329   \read\@inputcheck to\STC@line
```

\STC@test The make sure that \STC@test has enough arguments:

```
\STC@line
2330   \expandafter\STC@test\STC@line.....\STC@% %%H0: . added
2331 }%
```

\STC@test The \STC@test macro finds the “interesting” commands in the TOC file, mainly to
 \STC@contentsline delimit sections;

```
\mtc@string
\STC@list 2332 \long\def\STC@test#1#2#3#4#5#6\STC@{% %%H0: #6 added
\STC@toc 2333   \ifx#1\contentsline
\STC@read 2334     \let\mtc@string\string
\sectend 2335     \STC@contentsline{#2}{#3}{#4}{#5}% %%H0: #4 added
\sectbegin 2336     \let\mtc@string\relax
2337   \else\ifx#1\@input
2338     \edef\STC@list{\STC@list#2\relax}%
2339   \else\ifx#1\sectend
2340     \immediate\closeout\tf@mtc
2341     \immediate\openout\tf@mtc=\jobname.mtc
2342   \else\ifx#1\sectbegin
2343     \addtocounter{stc}{-1}%
2344   \fi\fi\fi\fi
2345   \ifeof\@inputcheck\expandafter\STC@toc
2346   \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 2347 \def\SLF@next#1\relax#2\{\%
2348   \edef\SLF@list{#2}%
2349   \SLF@loop{#1}}
```

\SLF@lof Check if the list is empty:

```
\SLF@list
\SLF@explist 2350 \def\SLF@lof{%
2351   \ifx\SLF@list\@empty\else\expandafter\SLF@explist\fi}
```

\SLF@contentsline The macro \SLF@contentsline analyses the lines read from the LOF file. If \section
 \section is found, the stc counter is incremented and a new sectlof file is created.

```
\thestc
\tf@mtc 2352 \def\SLF@contentsline#1#2#3#4{% %%H0: #4 added
\slfname
```

\MTC@WriteContentsLine

```

2353 \gdef\thestc{\arabic{stc}}% %%H0: space removed
2354 \expandafter\ifx\csname #1\endcsname\xsect
2355 \stepcounter{stc}%
2356 %% \gdef\thestc{\arabic{stc}} %%H0: removed

```

`\if@longextensions@` We test if long or short extensions are used, to build the name of the mini-table file, then
`\slfname` open it:

```

2357 \if@longextensions@%
2358 \mtcPackageInfo[<I0033>]{minitoc}%
2359 {Writing\space\jobname.slf\thestc}%
2360 \def\slfname{\jobname.slf\thestc}%
2361 \else
2362 \mtcPackageInfo[<I0033>]{minitoc}%
2363 {Writing\space\jobname.H\thestc}%
2364 \def\slfname{\jobname.H\thestc}%
2365 \fi
2366 \immediate\closeout\tf@mtc
2367 \immediate\openout\tf@mtc=\slfname
2368 \fi

```

`\mtc@toks` Now, we filter the relevant contents lines:

```

\MTc@WriteContentsline
\figure 2369 \mtc@toks{\noexpand\leavevmode #2}%
\subfigure 2370 \expandafter\ifx\csname #1\endcsname\figure
2371 \MTc@WriteContentsline{#1}{slf}{#3}{#4}%
2372 \fi
2373 \expandafter\ifx\csname #1\endcsname\subfigure
2374 \MTc@WriteContentsline{#1}{slfS}{#3}{#4}%
2375 \fi
2376 }

```

`\SLF@explist` The loop to read the lines of the LOF file; expands the list of entries and call `\SLF@next`
`\SLF@next` to process the first one.

```

\SLF@list
2377 \def\SLF@explist{\expandafter\SLF@next\SLF@list\}

```

`\SLF@loop` If an entry is found, loop through line by line, looking for interesting entries. Otherwise,
`\SLF@lof` process the next entry in the list.

```

\SLF@read
2378 \def\SLF@loop#1{\openin\@inputcheck#1\relax
2379 \ifeof\@inputcheck
2380 \mtcPackageWarning[<W0014>]{minitoc}%
2381 {No file #1\MessageBreak SECTLOFS NOT PREPARED}%
2382 \expandafter\SLF@lof
2383 \else

```

```

2384 \mtcPackageInfo[<I0036>]{minitoc}%
2385 {PREPARING SECTLOFS FROM #1}%
2386 \expandafter\SLF@read\fi}

```

Read the next entry of the .lof file.

\SLF@read The make sure that \SLF@test has enough arguments:

\SLF@test

```

\SLF@line 2387 \def\SLF@read{%
2388 \read\@inputcheck to\SLF@line
2389 \expandafter\SLF@test\SLF@line. . . . \SLF@% %%H0: . added
2390 }%

```

\SLF@test The \SLF@test macro finds the “interesting” commands in the LOF file, mainly to
\SLF@contentsline delimit sections;

```

\mtc@string
\SLF@list 2391 \long\def\SLF@test#1#2#3#4#5#6\SLF@{% %%H0: #6 added
\SLF@lof 2392 \ifx#1\contentsline
\SLF@read 2393 \let\mtc@string\string
\sectend 2394 \SLF@contentsline{#2}{#3}{#4}{#5}% %%H0: #4 added
\sectbegin 2395 \let\mtc@string\relax
2396 \else\ifx#1\@input
2397 \edef\SLF@list{\SLF@list#2\relax}%
2398 \else\ifx#1\sectend
2399 \immediate\closeout\tf@mtc
2400 \immediate\openout\tf@mtc=\jobname.mtc
2401 \else\ifx#1\sectbegin
2402 \addtocounter{stc}{-1}%
2403 \fi\fi\fi\fi
2404 \ifeof\@inputcheck\expandafter\SLF@lof
2405 \else\expandafter\SLF@read\fi}%

```

\SLT@next Processing the next entry in the list and remove it from the head of the list:

\SLT@list

```

\SLT@loop 2406 \def\SLT@next#1\relax#2\{\%
2407 \edef\SLT@list{#2}%
2408 \SLT@loop{#1}}

```

\SLT@lot Check if the list is empty:

\SLT@list

```

\SLT@explist 2409 \def\SLT@lot{%
2410 \ifx\SLT@list\@empty\else\expandafter\SLT@explist\fi}

```

```

\SLT@contentsline The macro \SLT@contentsline analyses the lines read from the LOT file. If \section
\section is found, the stc counter is incremented and a new sectlot file is created.
\thestc
\tf@mtc 2411 \def\SLT@contentsline#1#2#3#4{% %%H0: #4 added
\sltname 2412 \gdef\thestc{\arabic{stc}} %%H0: space removed
\MTC@WriteContentsLine 2413 \expandafter\ifx\csname #1\endcsname\xsect
2414 \stepcounter{stc}%
2415 %% \gdef\thestc{\arabic{stc}} %%H0: removed

```

```

\if@longextensions@ We test if long or short extensions are used, to build the name of the mini-table file, then
\sltname open it:

```

```

2416 \if@longextensions@%
2417 \mtcPackageInfo[<I0033>]{minitoc}%
2418 {Writing\space\jobname.slt\thestc}%
2419 \def\sltname{\jobname.slt\thestc}%
2420 \else
2421 \mtcPackageInfo[<I0033>]{minitoc}%
2422 {Writing\space\jobname.V\thestc}%
2423 \def\sltname{\jobname.V\thestc}%
2424 \fi
2425 \immediate\closeout\tf@mtc
2426 \immediate\openout\tf@mtc=\sltname
2427 \fi

```

```

\mtc@toks Now, we filter the relevant contents lines:
\MTC@WriteContentsline
\table 2428 \mtc@toks{\noexpand\leavevmode #2}%
\subtable 2429 \expandafter\ifx\csname #1\endcsname\table
2430 \MTC@WriteContentsline{#1}{slt}{#3}{#4}%
2431 \fi
2432 \expandafter\ifx\csname #1\endcsname\subtable
2433 \MTC@WriteContentsline{#1}{sltS}{#3}{#4}%
2434 \fi
2435 }

```

```

\SLT@explist The loop to read the lines of the LOT file; expands the list of entries and call \SLT@next
\SLT@next to process the first one.
\SLT@list
2436 \def\SLT@explist{\expandafter\SLT@next\SLT@list\}

```

```

\SLT@loop If an entry is found, loop through line by line, looking for interesting entries. Otherwise,
\SLT@lot process the next entry in the list.
\SLT@read
2437 \def\SLT@loop#1{\openin\@inputcheck#1\relax
2438 \ifeof\@inputcheck

```

```

2439 \mtcPackageWarning[<W0015>]{minitoc}%
2440 {No file #1\MessageBreak SECTLOTS NOT PREPARED}%
2441 \expandafter\SLT@lot
2442 \else
2443 \mtcPackageInfo[<I0039>]{minitoc}%
2444 {PREPARING SECTLOTS FROM #1}%
2445 \expandafter\SLT@read\fi}

```

Read the next entry of the .lot file.

\SLT@read The make sure that \SLT@test has enough arguments:

\SLT@test

```

\SLT@line 2446 \def\SLT@read{%
2447 \read\@inputcheck to\SLT@line
2448 \expandafter\SLT@test\SLT@line. . . . \SLT@ %%H0: . added
2449 }%

```

\SLT@test The \SLT@test macro finds the “interesting” commands in the LOT file, mainly to
\SLT@contentsline delimit sections;

```

\mtc@string
\SLT@list 2450 \long\def\SLT@test#1#2#3#4#5#6\SLT@{% %%H0: #6 added
\SLT@lot 2451 \ifx#1\contentsline
\SLT@read 2452 \let\mtc@string\string
\sectend 2453 \SLT@contentsline{#2}{#3}{#4}{#5}% %%H0: #4 added
\sectbegin 2454 \let\mtc@string\relax
2455 \else\ifx#1\@input
2456 \edef\SLT@list{\SLT@list#2\relax}%
2457 \else\ifx#1\sectend
2458 \immediate\closeout\tf@mtc
2459 \immediate\openout\tf@mtc=\jobname.mtc
2460 \else\ifx#1\sectbegin
2461 \addtocounter{stc}{-1}%
2462 \fi\fi\fi\fi
2463 \ifeof\@inputcheck\expandafter\SLT@lot
2464 \else\expandafter\SLT@read\fi}%

```

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 the test \@ifundefined{chapter}:

```

2465 \fi% end of \if@mtc@section@def@
2466 \fi% end of \if@mtc@chapter@undef@

```

9.63 The \mtcprepare command

\mtcprepare This command tests the availability of the \do... minitoc preparation commands and of
 \@ifnextchar the contents files, then calls as much as possible of these preparation commands. A hint
 \mtcprepare@ is given.
 \@ifundefined
 \IfFileExists 2467 \def\mtcprepare{\@ifnextchar[{\mtcprepare@}{\mtcprepare@[1]}}%
 \jobname 2468 \def\mtcprepare@[#1]{%
 \doparttoc 2469 \@ifundefined{part}{}{%
 \dopartlof 2470 \IfFileExists{\jobname.toc}{\doparttoc[#1]}{}%
 \dopartlot 2471 \IfFileExists{\jobname.lof}{\dopartlof[#1]}{}%
 \dominitoc 2472 \IfFileExists{\jobname.lot}{\dopartlot[#1]}{}%
 \dominilof 2473 }%
 \dominilot 2474 \@ifundefined{chapter}{%
 \dosecttoc 2475 \@ifundefined{section}{}{%
 \dosectlof 2476 {%
 \dosectlot 2477 \IfFileExists{\jobname.toc}{\dosecttoc[#1]}{}%
 \dosectlof 2478 \IfFileExists{\jobname.lof}{\dosectlof[#1]}{}%
 \mtcPackageInfo 2479 \IfFileExists{\jobname.lot}{\dosectlot[#1]}{}%
 2480 }%
 2481 }{}%
 2482 \IfFileExists{\jobname.toc}{\dominitoc[#1]}{}%
 2483 \IfFileExists{\jobname.lof}{\dominilof[#1]}{}%
 2484 \IfFileExists{\jobname.lot}{\dominilot[#1]}{}%
 2485 }%
 2486 \if@mtc@hints@
 2487 \@mtc@hints@given@true
 2488 \mtcPackageInfo[<I0048>]{minitoc(hints)}%
 2489 {Using \string\mtcprepare\space may induce some\MessageBreak
 2490 hints about the preparation commands,\MessageBreak
 2491 because it invokes ALL the preparation\MessageBreak
 2492 commands allowed by the document class,\MessageBreak
 2493 without any previous check\@gobble}%
 2494 \fi
 2495 }

9.64 Necessary \l@... commands

\l@listof Some \l@... commands (analog to \l@section or \l@paragraph) are required to
 \l@startpart format some entries in the mini-tables, for starred sectionning commands essentially:
 \l@starchapter
 \l@starsection 2496 \@ifundefined{section}{}{\let\l@listof\l@section}
 \l@starsubsection 2497 \@ifundefined{chapter}{}{\let\l@listof\l@chapter}
 \l@starsubsubsection 2498 \@ifundefined{part}{}{\let\l@startpart\l@part}
 \l@starparagraph 2499 \@ifundefined{chapter}{}{\let\l@starchapter\l@chapter}
 \l@starsubparagraph 2500 \@ifundefined{section}{}{\let\l@starsection\l@section}
 2501 \@ifundefined{subsection}{}{\let\l@starsubsection\l@subsection}
 2502 \@ifundefined{subsubsection}{}{\let\l@starsubsubsection\l@subsubsection}

```

2503 \@ifundefined{paragraph}{}{\let\l@starparagraph\l@paragraph}
2504 \@ifundefined{subparagraph}{}{\let\l@starsubparagraph\l@subparagraph}

```

9.65 The horizontal rules and their default values

```

\noptcrule We define here the various commands to activate ou inhibit the horizontal rules in
\nomtcrule the various kinds of mini-tables. Each such command is an indirect definition of the
\nostcrule corresponding horizontal rule. The rules are .4 pt high horizontal rules.
\ptcrule
\mtcrule 2505 \def\noptcrule{\let\ptc@rule\relax}
\stcrule 2506 \def\nomtcrule{\let\mtc@rule\relax}
\noplfrule 2507 \def\nostcrule{\let\stc@rule\relax}
\nomlfrule 2508 \def\ptc@rule{\def\ptc@rule{\kern-3\p@ \hrule width \columnwidth \kern2.6\p@}}
\noslfrule 2509 \def\mtc@rule{\def\mtc@rule{\kern-3\p@ \hrule width \columnwidth \kern2.6\p@}}
\plfrule 2510 \def\stc@rule{\def\stc@rule{\kern-3\p@ \hrule width \columnwidth \kern2.6\p@}}
\mlfrule 2511 \def\ptc@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}
\slfrule 2512 \def\mtc@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}
2513 \def\stc@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}
\nopltrule
\nomltrule
\nosltrule 2514 \def\noplfrule{\let\plf@rule\relax}
2515 \def\nomlfrule{\let\mlf@rule\relax}
\pltrule 2516 \def\noslfrule{\let\slf@rule\relax}
\mltrule 2517 \def\plfrule{\def\plf@rule{\kern-3\p@ \hrule width \columnwidth \kern2.6\p@}}
\sltrule 2518 \def\mlfrule{\def\mlf@rule{\kern-3\p@ \hrule width \columnwidth \kern2.6\p@}}
\columnwidth 2519 \def\slfrule{\def\slf@rule{\kern-3\p@ \hrule width \columnwidth \kern2.6\p@}}
2520 \def\plf@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}
2521 \def\mlf@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}
2522 \def\slf@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}

2523 \def\nopltrule{\let\plt@rule\relax}
2524 \def\nomltrule{\let\mlt@rule\relax}
2525 \def\nosltrule{\let\slt@rule\relax}
2526 \def\pltrule{\def\plt@rule{\kern-3\p@ \hrule width \columnwidth \kern2.6\p@}}
2527 \def\mltrule{\def\mlt@rule{\kern-3\p@ \hrule width \columnwidth \kern2.6\p@}}
2528 \def\sltrule{\def\slt@rule{\kern-3\p@ \hrule width \columnwidth \kern2.6\p@}}
2529 \def\plt@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}
2530 \def\mlt@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}
2531 \def\slt@rule{\rule[3\p@]{\columnwidth}{.4\p@}\vspace*{2.6\p@}}

```

9.66 The \mtcset... commands

These commands ⁷ have been introduced to build a nicer user interface, and each of them replaces numerous user commands, offering a rather unified and logical syntax.

⁷ The general concept of the \mtcset... commands was proposed by Benjamin BAYART.

9.66.1 Keywords for the `\mtcset...` commands

`\@namedef` We define some common keywords for the `\mtcset...` commands. A keyword is
`\@nameuse` 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:

```
2532 \@namedef{mtc@typetable@parttoc}{ptc}\def\mtc@typetable@parttoc{ptc}
2533 \@namedef{mtc@typetable@partlof}{plf}\def\mtc@typetable@partlof{plf}
2534 \@namedef{mtc@typetable@partlot}{plt}\def\mtc@typetable@partlot{plt}
2535 \@namedef{mtc@typetable@minitoc}{mtc}\def\mtc@typetable@minitoc{mtc}
2536 \@namedef{mtc@typetable@minilof}{mlf}\def\mtc@typetable@minilof{mlf}
2537 \@namedef{mtc@typetable@minilot}{mlt}\def\mtc@typetable@minilot{mlt}
2538 \@namedef{mtc@typetable@secttoc}{stc}\def\mtc@typetable@secttoc{stc}
2539 \@namedef{mtc@typetable@sectlof}{slf}\def\mtc@typetable@sectlof{slf}
2540 \@namedef{mtc@typetable@sectlot}{slt}\def\mtc@typetable@sectlot{slt}
```

`\@namedef` Then another family (typetitle) for the titles of the mini-tables:

```
2541 \@namedef{mtc@typetitle@parttoc}{pti}\def\mtc@typetitle@parttoc{pti}
2542 \@namedef{mtc@typetitle@partlof}{pti}\def\mtc@typetitle@partlof{pti}
2543 \@namedef{mtc@typetitle@partlot}{pti}\def\mtc@typetitle@partlot{pti}
2544 \@namedef{mtc@typetitle@minitoc}{mti}\def\mtc@typetitle@minitoc{mti}
2545 \@namedef{mtc@typetitle@minilof}{mti}\def\mtc@typetitle@minilof{mti}
2546 \@namedef{mtc@typetitle@minilot}{mti}\def\mtc@typetitle@minilot{mti}
2547 \@namedef{mtc@typetitle@secttoc}{sti}\def\mtc@typetitle@secttoc{sti}
2548 \@namedef{mtc@typetitle@sectlof}{sti}\def\mtc@typetitle@sectlof{sti}
2549 \@namedef{mtc@typetitle@sectlot}{sti}\def\mtc@typetitle@sectlot{sti}
```

`\@namedef` We define a family (YN) of keywords to recognize the keywords “off” and “on”, with their many synonyms⁸ and meaning false or true:

```
2550 \@namedef{mtc@YN@off}{no}\def\mtc@YN@off{no}
2551 \@namedef{mtc@YN@OFF}{no}\def\mtc@YN@OFF{no}
```

⁸ 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 `\expandafter \csname ... \endcsname`.

```

2552 \@namedef{mtc@YN@no}{no}\def\mtc@YN@no{no}
2553 \@namedef{mtc@YN@NO}{no}\def\mtc@YN@NO{no}
2554 \@namedef{mtc@YN@n}{no}\def\mtc@YN@n{no}
2555 \@namedef{mtc@YN@N}{no}\def\mtc@YN@N{no}
2556 \@namedef{mtc@YN@false}{no}\def\mtc@YN@false{no}
2557 \@namedef{mtc@YN@FALSE}{no}\def\mtc@YN@FALSE{no}
2558 \@namedef{mtc@YN@faux}{no}\def\mtc@YN@faux{no}
2559 \@namedef{mtc@YN@FAUX}{no}\def\mtc@YN@FAUX{no}
2560 \@namedef{mtc@YN@f}{no}\def\mtc@YN@f{no}
2561 \@namedef{mtc@YN@F}{no}\def\mtc@YN@F{no}
2562 \@namedef{mtc@YN@NON}{no}\def\mtc@YN@NON{no}
2563 \@namedef{mtc@YN@non}{no}\def\mtc@YN@non{no}
2564 \@namedef{mtc@YN@0}{no}\expandafter\def\csname mtc@YN@0\endcsname{no}
2565 \@namedef{mtc@YN@on}{}}\def\mtc@YN@on{}}
2566 \@namedef{mtc@YN@ON}{}}\def\mtc@YN@ON{}}
2567 \@namedef{mtc@YN@yes}{}}\def\mtc@YN@yes{}}
2568 \@namedef{mtc@YN@YES}{}}\def\mtc@YN@YES{}}
2569 \@namedef{mtc@YN@y}{}}\def\mtc@YN@y{}}
2570 \@namedef{mtc@YN@Y}{}}\def\mtc@YN@Y{}}
2571 \@namedef{mtc@YN@true}{}}\def\mtc@YN@true{}}
2572 \@namedef{mtc@YN@TRUE}{}}\def\mtc@YN@TRUE{}}
2573 \@namedef{mtc@YN@t}{}}\def\mtc@YN@t{}}
2574 \@namedef{mtc@YN@T}{}}\def\mtc@YN@T{}}
2575 \@namedef{mtc@YN@vrai}{}}\def\mtc@YN@vrai{}}
2576 \@namedef{mtc@YN@VRAI}{}}\def\mtc@YN@VRAI{}}
2577 \@namedef{mtc@YN@v}{}}\def\mtc@YN@v{}}
2578 \@namedef{mtc@YN@V}{}}\def\mtc@YN@TRUEV
2579 \@namedef{mtc@YN@OUI}{}}\def\mtc@YN@OUI{}}
2580 \@namedef{mtc@YN@oui}{}}\def\mtc@YN@oui{}}
2581 \@namedef{mtc@YN@O}{}}\def\mtc@YN@O{}}
2582 \@namedef{mtc@YN@o}{}}\def\mtc@YN@o{}}
2583 \@namedef{mtc@YN@1}{}}\expandafter\def\csname mtc@YN@1\endcsname{}}

```

9.66.2 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.

```

2584 \@namedef{mtc@sectlevel@chapter}{C}\def\mtc@sectlevel@chapter{C}
2585 \@namedef{mtc@sectlevel@section}{S}\def\mtc@sectlevel@section{S}
2586 \@namedef{mtc@sectlevel@subsection}{SS}\def\mtc@sectlevel@subsection{SS}
2587 \@namedef{mtc@sectlevel@subsubsection}{SSS}\def\mtc@sectlevel@subsubsection{SSS}
2588 \@namedef{mtc@sectlevel@paragraph}{P}\def\mtc@sectlevel@paragraph{P}
2589 \@namedef{mtc@sectlevel@subparagraph}{SP}\def\mtc@sectlevel@subparagraph{SP}
2590 \@namedef{mtc@sectlevel@*}{}}\expandafter\def\csname mtc@sectlevel@*\endcsname{}}
2591 \@namedef{mtc@sectlevel@figure}{}}\def\mtc@sectlevel@figure{}}
2592 \@namedef{mtc@sectlevel@table}{}}\def\mtc@sectlevel@table{}}
2593 \@namedef{mtc@sectlevel@subfigure}{S}\def\mtc@sectlevel@subfigure{S}
2594 \@namedef{mtc@sectlevel@subtable}{S}\def\mtc@sectlevel@subtable{S}

```

`\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 sectioning level like `subsection` (no backslash). The *font commands* are a font specification, using NFSS [141] basic commands usually.

`\if@mtc@setfont@` First, we declare a flag, set true:

```
2595 \newif\if@mtc@setfont@\@mtc@setfont@true
```

`\mtcsetfont` Then, we begin the command, which has three arguments:

```
2596 \newcommand{\mtcsetfont}[3]{%
```

`\mtc@mta@abbrev` 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.
`\if@mtc@setfont@`
`\@nameuse` The result is stored in `\mtc@mta@abbrev`. Example: if #1 is `minitoc`, we get `mtc`.

```
2597 \def\mtc@mta@abbrev{X}
2598 \@mtc@setfont@true
2599 \expandafter\ifx\csname mtc@typetable@#1\endcsname\relax
2600   \@mtc@setfont@false
2601   \def\mtc@mta@abbrev{X}
2602   \mtcPackageError[<E0013>]{minitoc}%
2603     {\string\mtcsetfont \space has a wrong first argument
2604     \MessageBreak
2605     (#1).
2606     \MessageBreak
2607     It should be a mini-table type
2608     \MessageBreak
2609     (parttoc...sectlot)}%
2610     {Correct the source code.\MessageBreak
2611     Type <return> and rerun LaTeX}
2612 \else
2613   \edef\mtc@mta@abbrev{\@nameuse{mtc@typetable@#1}}
2614 \fi
```

`\mtc@level@abbrev` The second argument, a keyword from the family `sectlevel`, is processed the same way
`\if@mtc@setfont@` and the result is stored into `\mtc@level@abbrev`. Example: if #2 is the `subparagraph`
`\@nameuse` keyword, we get `SP`.

```
2615 \def\mtc@level@abbrev{X}
2616 \expandafter\ifx\csname mtc@sectlevel@#2\endcsname\relax
```

```

2617 \@mtc@setfont@false
2618 \def\mtc@level@abbrev{X}
2619 \mtcPackageError[<E0014>]{minitoc}%
2620 {\string\mtcsetfont \space has a wrong second argument
2621 \MessageBreak
2622 (#2).
2623 \MessageBreak
2624 It should be a sectionning level
2625 \MessageBreak
2626 (part...subparagraph) or * }%
2627 {Correct the source code.\MessageBreak
2628 Type <return> and rerun LaTeX}
2629 \else
2630 \edef\mtc@level@abbrev{\@nameuse{mtc@sectlevel@#2}}
2631 \fi

```

```

\mtc@tmp@name Then, we construct the effective macro to be applied:
\mtc@mta@abbrev
\mtc@level@abbrev 2632 \def\mtc@tmp@name{\mtc@mta@abbrev\mtc@level@abbrev font}

```

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).

```

\if@mtc@setfont@ But all combinaisons are not legal (the level of the entry must be lower than the level of
the mini-table), so we must test. Special care must be taken for testing via internal defined
commands (quarks) with @mtcqk at the end of their names).

```

```

2633 \def\parttoc@mtcqk{parttoc@mtcqk}
2634 \def\minitoc@mtcqk{minitoc@mtcqk}
2635 \def\secttoc@mtcqk{secttoc@mtcqk}
2636 \def\partlof@mtcqk{partlof@mtcqk}
2637 \def\minilof@mtcqk{minilof@mtcqk}
2638 \def\sectlof@mtcqk{sectlof@mtcqk}
2639 \def\partlot@mtcqk{partlot@mtcqk}
2640 \def\minilot@mtcqk{minilot@mtcqk}
2641 \def\sectlot@mtcqk{sectlot@mtcqk}
2642 \def\part@mtcqk{part@mtcqk}
2643 \def\chapter@mtcqk{chapter@mtcqk}
2644 \def\section@mtcqk{section@mtcqk}
2645 \def\subsection@mtcqk{subsection@mtcqk}
2646 \def\subsubsection@mtcqk{subsubsection@mtcqk}
2647 \def\paragraph@mtcqk{paragraph@mtcqk}
2648 \def\subparagraph@mtcqk{subparagraph@mtcqk}
2649 \def\figure@mtcqk{figure@mtcqk}
2650 \def\table@mtcqk{table@mtcqk}
2651 \def\subfigure@mtcqk{subfigure@mtcqk}
2652 \def\subtable@mtcqk{subtable@mtcqk}
2653 \@mtc@setfont@true
2654 \expandafter\ifx\csname #1@mtcqk\endcsname\parttoc@mtcqk\relax

```

```

2655 \expandafter\ifx\csname #2@mtcck\endcsname\figure@mtcck\relax\@mtc@setfont@false\fi
2656 \expandafter\ifx\csname #2@mtcck\endcsname\subfigure@mtcck\relax\@mtc@setfont@false\fi
2657 \expandafter\ifx\csname #2@mtcck\endcsname\table@mtcck\relax\@mtc@setfont@false\fi
2658 \expandafter\ifx\csname #2@mtcck\endcsname\subtable@mtcck\relax\@mtc@setfont@false\fi
2659 \fi
2660 \expandafter\ifx\csname #1@mtcck\endcsname\partlof@mtcck\relax
2661 \expandafter\ifx\csname #2@mtcck\endcsname\table@mtcck\relax\@mtc@setfont@false\fi
2662 \expandafter\ifx\csname #2@mtcck\endcsname\subtable@mtcck\relax\@mtc@setfont@false\fi
2663 \fi
2664 \expandafter\ifx\csname #1@mtcck\endcsname\partlot@mtcck\relax
2665 \expandafter\ifx\csname #2@mtcck\endcsname\figure@mtcck\relax\@mtc@setfont@false\fi
2666 \expandafter\ifx\csname #2@mtcck\endcsname\subfigure@mtcck\relax\@mtc@setfont@false\fi
2667 \fi
2668 \expandafter\ifx\csname #1@mtcck\endcsname\minitoc@mtcck\relax
2669 \expandafter\ifx\csname #2@mtcck\endcsname\part@mtcck\relax\@mtc@setfont@false\fi
2670 \expandafter\ifx\csname #2@mtcck\endcsname\chapter@mtcck\relax\@mtc@setfont@false\fi
2671 \expandafter\ifx\csname #2@mtcck\endcsname\figure@mtcck\relax\@mtc@setfont@false\fi
2672 \expandafter\ifx\csname #2@mtcck\endcsname\subfigure@mtcck\relax\@mtc@setfont@false\fi
2673 \expandafter\ifx\csname #2@mtcck\endcsname\table@mtcck\relax\@mtc@setfont@false\fi
2674 \expandafter\ifx\csname #2@mtcck\endcsname\subtable@mtcck\relax\@mtc@setfont@false\fi
2675 \fi
2676 \expandafter\ifx\csname #1@mtcck\endcsname\minilof@mtcck\relax
2677 \expandafter\ifx\csname #2@mtcck\endcsname\part@mtcck\relax\@mtc@setfont@false\fi
2678 \expandafter\ifx\csname #2@mtcck\endcsname\chapter@mtcck\relax\@mtc@setfont@false\fi
2679 \expandafter\ifx\csname #2@mtcck\endcsname\table@mtcck\relax\@mtc@setfont@false\fi
2680 \expandafter\ifx\csname #2@mtcck\endcsname\subtable@mtcck\relax\@mtc@setfont@false\fi
2681 \fi
2682 \expandafter\ifx\csname #1@mtcck\endcsname\minilot@mtcck\relax
2683 \expandafter\ifx\csname #2@mtcck\endcsname\part@mtcck\relax\@mtc@setfont@false\fi
2684 \expandafter\ifx\csname #2@mtcck\endcsname\chapter@mtcck\relax\@mtc@setfont@false\fi
2685 \expandafter\ifx\csname #2@mtcck\endcsname\figure@mtcck\relax\@mtc@setfont@false\fi
2686 \expandafter\ifx\csname #2@mtcck\endcsname\subfigure@mtcck\relax\@mtc@setfont@false\fi
2687 \fi
2688 \expandafter\ifx\csname #1@mtcck\endcsname\secttoc@mtcck\relax
2689 \expandafter\ifx\csname #2@mtcck\endcsname\part@mtcck\relax\@mtc@setfont@false\fi
2690 \expandafter\ifx\csname #2@mtcck\endcsname\chapter@mtcck\relax\@mtc@setfont@false\fi
2691 \expandafter\ifx\csname #2@mtcck\endcsname\section@mtcck\relax\@mtc@setfont@false\fi
2692 \expandafter\ifx\csname #2@mtcck\endcsname\figure@mtcck\relax\@mtc@setfont@false\fi
2693 \expandafter\ifx\csname #2@mtcck\endcsname\subfigure@mtcck\relax\@mtc@setfont@false\fi
2694 \expandafter\ifx\csname #2@mtcck\endcsname\table@mtcck\relax\@mtc@setfont@false\fi
2695 \expandafter\ifx\csname #2@mtcck\endcsname\subtable@mtcck\relax\@mtc@setfont@false\fi
2696 \fi
2697 \expandafter\ifx\csname #1@mtcck\endcsname\sectlof@mtcck\relax
2698 \expandafter\ifx\csname #2@mtcck\endcsname\part@mtcck\relax\@mtc@setfont@false\fi
2699 \expandafter\ifx\csname #2@mtcck\endcsname\chapter@mtcck\relax\@mtc@setfont@false\fi
2700 \expandafter\ifx\csname #2@mtcck\endcsname\section@mtcck\relax\@mtc@setfont@false\fi
2701 \expandafter\ifx\csname #2@mtcck\endcsname\table@mtcck\relax\@mtc@setfont@false\fi
2702 \expandafter\ifx\csname #2@mtcck\endcsname\subtable@mtcck\relax\@mtc@setfont@false\fi
2703 \fi
2704 \expandafter\ifx\csname #1@mtcck\endcsname\sectlot@mtcck\relax
2705 \expandafter\ifx\csname #2@mtcck\endcsname\part@mtcck\relax\@mtc@setfont@false\fi
2706 \expandafter\ifx\csname #2@mtcck\endcsname\chapter@mtcck\relax\@mtc@setfont@false\fi
2707 \expandafter\ifx\csname #2@mtcck\endcsname\section@mtcck\relax\@mtc@setfont@false\fi
2708 \expandafter\ifx\csname #2@mtcck\endcsname\figure@mtcck\relax\@mtc@setfont@false\fi

```

```

2709 \expandafter\ifx\csname #2@mtcqk\endcsname\subfigure@mtcqk\relax\@mtc@setfont@false\fi
2710 \fi

```

`\if@mtc@setfont@` If the combinaison 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 can print it *verbatim*).

`\mtc@tmp@name`

`\mtc@mta@abbrev`

`\mtc@level@abbrev`

`\mtc@toks` If the combinaison is not legal, an error message is displayed.

```

2711 \if@mtc@setfont@
2712 \def\mtc@tmp@name{\mtc@mta@abbrev\mtc@level@abbrev font}
2713 \mtc@toks{#3}
2714 \mtcPackageInfo[<I0015>]{minitoc}%
2715 {\string\mtcsetfont\space redefines the macro\MessageBreak
2716 "\mtc@tmp@name" as "\the\mtc@toks"}
2717 \expandafter\def\csname\mtc@tmp@name\endcsname{#3}
2718 \else
2719 \mtcPackageError[<E0024>]{minitoc}%
2720 {The macro \string\mtcsetfont\space has incompatible\MessageBreak
2721 first (#1) and second (#2) arguments}%
2722 {Correct the source code.\MessageBreak
2723 Type <return> and rerun LaTeX}
2724 \fi
2725 }

```

9.66.3 The `\mtcsetttitlefont` command

`\mtcsetttitlefont` This command is very similar to the `\mtcsetfont` command. Its syntax is almost identical:

```
\mtcsetttitlefont{mini-table}{font commands}
```

`\if@mtc@setttitlefont@` The *mini-table* type is a keyword like `minitoc`. The *font commands* are a font specification, using NFSS [141] basic commands usually. The difference is the absence of the second keyword argument, because the *font commands* will be applied to the title of each mini-table of the given kind.

First, we declare a flag, set true:

```
2726 \newif\if@mtc@setttitlefont@\@mtc@setttitlefont@true
```

`\mtcsettitlefont` And we begin the definition of the `\mtcsettitlefont` command, which has two arguments:

```
2727 \newcommand{\mtcsettitlefont}[2]{%
```

`\mtc@mtatf@abbrev` We process the first argument, a keyword of the `typetitle` family, then the result is stored into `\mtc@mtatf@abbrev`:
`\if@mtc@settitlefont@`
`\@nameuse`

```
2728 \def\mtc@mtatf@abbrev{X}
2729 \@mtc@settitlefont@true
2730 \expandafter\ifx\csname mtc@typetitle@#1\endcsname\relax
2731   \mtc@settitlefont@false
2732   \def\mtc@mtatf@abbrev{X}
2733   \mtcPackageError[<E0022>]{minitoc}%
2734     {\string\mtcsettitlefont \space has a wrong first argument
2735     \MessageBreak
2736     (#1).
2737     \MessageBreak
2738     It should be a mini-table type
2739     \MessageBreak
2740     (parttoc...sectlot))}%
2741   {Correct the source code.\MessageBreak
2742   Type <return> and rerun LaTeX}
2743 \else
2744   \edef\mtc@mtatf@abbrev{\@nameuse{mtc@typetitle@#1}}
2745 \fi
```

`\if@mtc@settitlefont@` Then we build the name of the effective command and apply this command:

```
\mtc@tmptf@name
\mtc@mtatf@abbrev 2746 \if@mtc@settitlefont@
\mtc@toks 2747   \def\mtc@tmptf@name{\mtc@mtatf@abbrev font}
2748   \mtc@toks{#2}
2749   \mtcPackageInfo[<I0018>]{minitoc}%
2750   {\string\mtcsettitlefont\space redefines the macro
2751   \MessageBreak "\mtc@tmptf@name" as
2752   \MessageBreak "\the\mtc@toks"}
2753   \expandafter\def\csname\mtc@tmptf@name\endcsname{#2}
2754 \else
2755   \mtcPackageError[<E0034>]{minitoc}%
2756   {The macro \string\mtcsettitlefont\space uses\MessageBreak
2757   an illegal type of table (#1)}%
2758   {Correct the source code.\MessageBreak
2759   Type <return> and rerun LaTeX}{\relax}
2760 \fi
2761 }
```

9.66.4 The `\mtcsettitle` command

`\mtcsettitle` This command is very similar to the `\mtcsettitlefont` command. Its syntax is almost identical:

```
\mtcsettitle{mini-table}{text}
```

The *mini-table* type is a keyword like `minitoc`. The *text* is the text for a mini-table title.

`\if@mtc@settitle@` First, we declare a flag, set true:

```
2762 \newif\if@mtc@settitle@\@mtc@settitle@true
```

`\mtcsettitle` Then we define the `\mtcsettitle` command, which has two arguments:

```
2763 \newcommand{\mtcsettitle}[2]{%
```

`\mtc@mtati@abbrev` We process the first argument, a keyword of the `typetable` family. The result is stored in `\mtc@mtati@abbrev`:

```
\if@mtc@settitle@
  \@nameuse
2764 \def\mtc@mtati@abbrev{X}
2765 \@mtc@settitle@true
2766 \expandafter\ifx\csname mtc@typetable@#1\endcsname\relax
2767   \@mtc@settitle@false
2768   \def\mtc@mtati@abbrev{X}
2769   \mtcPackageError[<E0021>]{minitoc}%
2770     {\string\mtcsettitle \space has a wrong first argument
2771     \MessageBreak
2772     (#1).
2773     \MessageBreak
2774     It should be a mini-table type
2775     \MessageBreak
2776     (parttoc...sectlot)}%
2777   {Correct the source code.\MessageBreak
2778   Type <return> and rerun LaTeX}
2779 \else
2780   \edef\mtc@mtati@abbrev{\@nameuse{mtc@typetable@#1}}
2781 \fi
```

`\if@mtc@settitle@` And we construct the name of the effective macro and apply it:

```
\mtc@tmpti@name
\mtc@mtati@abbrev 2782 \if@mtc@settitle@
\mtc@toks 2783   \def\mtc@tmpti@name{\mtc@mtati@abbrev title}
2784   \mtc@toks{#2}
2785   \mtcPackageInfo[<I0017>]{minitoc}%
```



```

2786     {\string\mtcsettitle\space redefines the macro\MessageBreak
2787      "\mtc@tmpti@name" as
2788       \MessageBreak "\the\mtc@toks"}
2789   \expandafter\def\csname\mtc@tmpti@name\endcsname{#2}
2790 \else
2791   \mtcPackageError[<E0033>]{minitoc}%
2792     {The macro \string\mtcsettitle\space uses\MessageBreak
2793      an illegal type of table (#1)}%
2794     {Correct the source code.\MessageBreak
2795      Type <return> and rerun LaTeX}{\relax}
2796 \fi
2797 }

```

9.66.5 The \mtcsetformat command

\@namedef We define first the keywords (family formatparam) for the three formatting parameters that this command can alter:

```

2798 \@namedef{mtc@formatparam@dotinterval}{dotsep}%
2799 \def\mtc@arg@dotinterval{dotsep}
2800 \@namedef{mtc@formatparam@tocrightmargin}{tocrmarg}%
2801 \def\mtc@arg@tocrightmargin{tocrightmargin}
2802 \@namedef{mtc@formatparam@pagenumwidth}{pnumwidth}%
2803 \def\mtc@arg@pagenumwidth{\mtc@arg@pagenumwidth}
2804 %% \@namedef{mtc@arg@numwidth}{numwidth} %not yet available
2805 %% \def\mtc@arg@numwidth{\mtc@arg@numwidth} %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:

```

2806 \AtBeginDocument{%

```

\@pnumwidth We take, if possible, the default value of \@pnumwidth for each type of mini-tables:

```

\ptcpnumwidth
\mtcpnumwidth 2807 \@ifundefined{ptcpnumwidth}{\let\ptcpnumwidth\@pnumwidth}{}%
\stcpnumwidth 2808 \@ifundefined{stcpnumwidth}{\let\stcpnumwidth\@pnumwidth}{}%
\plfpnumwidth 2809 \@ifundefined{mtcpnumwidth}{\let\mtcpnumwidth\@pnumwidth}{}%
\mlfpnumwidth 2810 \@ifundefined{plfpnumwidth}{\let\plfpnumwidth\@pnumwidth}{}%
\slfpnumwidth 2811 \@ifundefined{mlfpnumwidth}{\let\mlfpnumwidth\@pnumwidth}{}%
\pltpnumwidth 2812 \@ifundefined{slfpnumwidth}{\let\slfpnumwidth\@pnumwidth}{}%
\mltpnumwidth 2813 \@ifundefined{pltpnumwidth}{\let\pltpnumwidth\@pnumwidth}{}%
\sltpnumwidth 2814 \@ifundefined{mltpnumwidth}{\let\mltpnumwidth\@pnumwidth}{}%
2815 \@ifundefined{sltpnumwidth}{\let\sltpnumwidth\@pnumwidth}{}%

```

```

\@tocrmarg We take, if possible, the default value of \@tocrmarg for each type of mini-tables:
\ptctocrmarg
\mtclofrmarg 2816 \@ifundefined{ptctocrmarg}{\let\ptctocrmarg\@tocrmarg}{}%
\stclotrarg 2817 \@ifundefined{mtctocrmarg}{\let\mtctocrmarg\@tocrmarg}{}%
\plftocrmarg 2818 \@ifundefined{stctocrmarg}{\let\stctocrmarg\@tocrmarg}{}%
\mlflofrmarg 2819 \@ifundefined{plftocrmarg}{\let\plftocrmarg\@tocrmarg}{}%
\slflotrarg 2820 \@ifundefined{mlftocrmarg}{\let\mlftocrmarg\@tocrmarg}{}%
\plttocrmarg 2821 \@ifundefined{slftocrmarg}{\let\slftocrmarg\@tocrmarg}{}%
\mltlofrmarg 2822 \@ifundefined{plttocrmarg}{\let\plttocrmarg\@tocrmarg}{}%
\sltlotrarg 2823 \@ifundefined{mlttocrmarg}{\let\mlttocrmarg\@tocrmarg}{}%
2824 \@ifundefined{slttocrmarg}{\let\slttocrmarg\@tocrmarg}{}%

```

```

\@dotsep We take, if possible, the default value of \@dotsep for each type of mini-tables:
\ptcdotsep
\mtcdotsep 2825 \@ifundefined{ptcdotsep}{\let\ptcdotsep\@dotsep}{}%
\stcdotsep 2826 \@ifundefined{mtcdotsep}{\let\mtcdotsep\@dotsep}{}%
\plfdotsep 2827 \@ifundefined{stcdotsep}{\let\stcdotsep\@dotsep}{}%
\mlfdotsep 2828 \@ifundefined{plfdotsep}{\let\plfdotsep\@dotsep}{}%
\slfdotsep 2829 \@ifundefined{mlfdotsep}{\let\mlfdotsep\@dotsep}{}%
\pltdotsep 2830 \@ifundefined{slfdotsep}{\let\slfdotsep\@dotsep}{}%
\mltdotsep 2831 \@ifundefined{pltdotsep}{\let\pltdotsep\@dotsep}{}%
\sltdotsep 2832 \@ifundefined{mltdotsep}{\let\mltdotsep\@dotsep}{}%
2833 \@ifundefined{sltdotsep}{\let\sltdotsep\@dotsep}{}%

```

And we terminate the \AtBeginDocument block:

```
2834 }
```

\mtcsetformat 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.

\ptc@setform The \ptc@setform macro is invoked in ptc@verse to set format parameters:

```

ptc@verse
2835 \def\ptc@setform{%
2836 \let\@pnumwidth\ptcpnumwidth\relax
2837 \let\@tocrmarg\ptctocrmarg\relax
2838 \let\@dotsep\ptcdotsep\relax
2839 }

```

\mtc@setform The \mtc@setform macro is invoked in mtc@verse to set format parameters:

```

mtc@verse
2840 \def\mtc@setform{%
2841 \let\@pnumwidth\mtcpnumwidth\relax
2842 \let\@tocrmarg\mtctocrmarg\relax
2843 \let\@dotsep\mtcdotsep\relax
2844 }

```

`\stc@setform` The `\stc@setform` macro is invoked in `stc@verse` to set format parameters:
`stc@verse`

```
2845 \def\stc@setform{%
2846 \let\@pnumwidth\stcpnumwidth\relax
2847 \let\@tocrmarg\stctocrmarg\relax
2848 \let\@dotsep\stcdotsep\relax
2849 }
```

`\plf@setform` The `\plf@setform` macro is invoked in `ptc@verse` to set format parameters:
`ptc@verse`

```
2850 \def\plf@setform{%
2851 \let\@pnumwidth\plfpnumwidth\relax
2852 \let\@tocrmarg\plftocrmarg\relax
2853 \let\@dotsep\plfdotsep\relax
2854 }
```

`\mlf@setform` The `\mlf@setform` macro is invoked in `mtc@verse` to set format parameters:
`mtc@verse`

```
2855 \def\mlf@setform{%
2856 \let\@pnumwidth\mlfpnumwidth\relax
2857 \let\@tocrmarg\mlftocrmarg\relax
2858 \let\@dotsep\mlfdotsep\relax
2859 }
```

`\slf@setform` The `\slf@setform` macro is invoked in `stc@verse` to set format parameters:
`stc@verse`

```
2860 \def\slf@setform{%
2861 \let\@pnumwidth\slfpnumwidth\relax
2862 \let\@tocrmarg\slftocrmarg\relax
2863 \let\@dotsep\slfdotsep\relax
2864 }
```

`\plt@setform` The `\plt@setform` macro is invoked in `ptc@verse` to set format parameters:
`ptc@verse`

```
2865 \def\plt@setform{%
2866 \let\@pnumwidth\pltpnumwidth\relax
2867 \let\@tocrmarg\plttocrmarg\relax
2868 \let\@dotsep\pltdotsep\relax
2869 }
```

`\mlt@setform` The `\mlt@setform` macro is invoked in `mtc@verse` to set format parameters:
`mtc@verse`

```
2870 \def\mlt@setform{%
2871 \let\@pnumwidth\plfpnumwidth\relax
```

```

2872 \let\@tocrmarg\plftocrmarg\relax
2873 \let\@dotsep\plfdotsep\relax
2874 }

```

`\slt@setform` The `\slt@setform` macro is invoked in `stc@verse` to set format parameters:

```

stc@verse
2875 \def\slt@setform{%
2876 \let\@pnumwidth\plfpnumwidth\relax
2877 \let\@tocrmarg\plftocrmarg\relax
2878 \let\@dotsep\plfdotsep\relax
2879 }

```

`\if@mtc@setformat@` We now define a flag and the `\mtcsetformat` command, which has the following syntax:

```
\mtcsetformat{mini-table}{parameter-name}{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.

```

2880 \newif\if@mtc@setformat@\@mtc@setformat@true
2881 \newcommand{\mtcsetformat}[3]{%

```

`\mtc@mtf@abbrev` We now process the first argument and store the result in `\mtc@mtf@abbrev`:

`\mtc@fparam@abbrev`

```

2882 \def\mtc@mtf@abbrev{X}
2883 \def\mtc@fparam@abbrev{X}
2884 \@mtc@setformat@true
2885 \expandafter\ifx\csname mtc@typetable@#1\endcsname\relax
2886   \@mtc@setformat@false
2887   \def\mtc@mtf@abbrev{X}
2888   \mtcPackageError[<E0015>]{minitoc}%
2889     {\string\mtcsetformat \space has a wrong first argument
2890     \MessageBreak
2891     (#1).
2892     \MessageBreak
2893     It should be a mini-table type
2894     \MessageBreak
2895     (parttoc...sectlot)}}%
2896   {Correct the source code.\MessageBreak
2897   Type <return> and rerun LaTeX}
2898 \else
2899   \edef\mtc@mtf@abbrev{\@nameuse{mtc@typetable@#1}}
2900 \fi

```

`\mtc@fparam@abbrev` Then the second argument is processed and the result is stored in `\mtc@fparam@abbrev`:
`\@nameuse`

```

2901 \expandafter\ifx\csname mtc@formatparam@#2\endcsname\relax
2902 \@mtc@setformat@false
2903 \def\mtc@fparam@abbrev{X}
2904 \mtcPackageError[<E0016>]{minitoc}%
2905   {\string\mtcsetformat \space has a wrong second argument
2906     \MessageBreak
2907     (#2).
2908     \MessageBreak
2909     It should be a formatting param choosen from:\MessageBreak
2910     pagenumwidth, tocrightmargin, dotinterval}%
2911   {Correct the source code.\MessageBreak
2912     Type <return> and rerun LaTeX}
2913 \else
2914 \edef\mtc@fparam@abbrev{\@nameuse{mtc@formatparam@#2}}
2915 \fi

```

`\if@mtc@setformat@` The name of the storage macro is built and it receives the third parameter as value:

```

\mtc@tmpfm@name
\mtc@mtf@abbrev
\mtc@fparam@abbrev
2916 \if@mtc@setformat@
2917 \def\mtc@tmpfm@name{\mtc@mtf@abbrev\mtc@fparam@abbrev}
2918 \mtcPackageInfo[<I0016>]{minitoc}%
2919   {\string\mtcsetformat\space redefines the macro\MessageBreak
2920     "\mtc@tmpfm@name" as "\string#3"}
2921 \expandafter\def\csname\mtc@tmpfm@name\endcsname{#3}
2922 \else
2923 \mtcPackageError[<E0025>]{minitoc}%
2924   {The macro \string\mtcsetformat\space has incompatible\MessageBreak
2925     first (#1) and second (#2) arguments}%
2926   {Correct the source code.\MessageBreak
2927     Type <return> and rerun LaTeX}
2928 \fi
2929 }

```

9.66.6 The `\mtcsetpagenumbers` command

This command activates or inhibits page numbers in the mini-tables of a given kind. Its syntax is the following:

`\mtcsetpagenumbers{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.

`\if@mtc@setpagenumbers@` We define some flags:

`\if@mtc@spn@ok@`

2930 `\newif\if@mtc@setpagenumbers@ \@mtc@setpagenumbers@false`

2931 `\newif\if@mtc@spn@ok@`

`\mtcsetpagenumbers` We define the user-level macro. If the first argument is a star, we call the internal macro `\@ifundefined` `\mtcsetpagenumbers@` for each type of mini-table available; else, we call this internal `\mtcsetpagenumbers@` macro only once, for the specified type of mini-table.

```
2932 \newcommand{\mtcsetpagenumbers}[2]{%
2933   \expandafter\ifx\csname #1\endcsname\*\relax
2934   \@ifundefined{part}{}%
2935   {\mtcsetpagenumbers@{parttoc}{#2}}
2936   {\mtcsetpagenumbers@{partlof}{#2}}
2937   {\mtcsetpagenumbers@{partlot}{#2}}
2938   \@ifundefined{chapter}{}%
2939   {\mtcsetpagenumbers@{minitoc}{#2}}
2940   {\mtcsetpagenumbers@{minilof}{#2}}
2941   {\mtcsetpagenumbers@{minilot}{#2}}
2942   \@ifundefined{section}{}%
2943   {\mtcsetpagenumbers@{secttoc}{#2}}
2944   {\mtcsetpagenumbers@{sectlof}{#2}}
2945   {\mtcsetpagenumbers@{sectlot}{#2}}
2946   \else
2947   \mtcsetpagenumbers@{#1}{#2}%
2948   \fi
2949 }
```

`\mtcsetpagenumber@` Then the `\mtcsetpagenumbers@` internal macro, with two arguments:

2950 `\newcommand{\mtcsetpagenumbers@}[2]{%`

`\mtc@mttpn@abbrev` We process the first argument, a keyword of the `typetable` family, and store the result in `\mtc@pns@abbrev`

`\mtc@pns@abbrev`

```
2951 \def\mtc@mttpn@abbrev{X}
2952 \@mtc@setpagenumbers@true
2953 \def\mtc@pns@abbrev{}
2954 \expandafter\ifx\csname mtc@typetable@#1\endcsname\relax
2955   \@mtc@setpagenumbers@false
2956   \def\mtc@pns@abbrev{X}
2957   \def\mtc@mttpn@abbrev{X}
2958   \mtcPackageError[<E0017>]{minitoc}%
2959     {\string\mtcsetpagenumbers \space has a wrong first
2960     \MessageBreak
2961     argument (#1)}%
2962     {It should be a mini-table type
2963     \MessageBreak
2964     (parttoc...sectlot)\MessageBreak
```

```

2965      Correct the source code.\MessageBreak
2966      Type <return> and rerun LaTeX}
2967 \else
2968   \edef\mtc@mttpn@abbrev{\@nameuse{mtc@typetable@#1}}
2969 \fi

\if@mtc@spn@ok@ Then the second argument, a keyword of the YN family, and store the result in
\mtc@pns@abbrev \mtc@pns@abbrev. The name of the effective macro is built and the macro executed.
\mtc@mttpn@abbrev
\if@mtc@setpagenumbers@ 2970 \@mtc@spn@ok@true
\mtc@tmppn@name 2971 \expandafter\ifx\csname mtc@YN@#2\endcsname\relax
2972   \@mtc@spn@ok@false
2973   \def\mtc@pns@abbrev{X}
2974   \def\mtc@mttpn@abbrev{X}
2975   \@mtc@setpagenumbers@false
2976   \def\mtc@mttpn@abbrev{X}
2977   \mtcPackageError[<E0018>]{minitoc}%
2978     {\string\mtcsetpagenumbers \space has a wrong second
2979     \MessageBreak
2980     argument (#2)}%
2981     {It should be a boolean value (0/1, yes/no, on/off)\MessageBreak
2982     Correct the source code.\MessageBreak
2983     Type <return> and rerun LaTeX}
2984 \else
2985   \edef\mtc@pns@abbrev{\@nameuse{mtc@YN@#2}}
2986   \def\mtc@pns@abbrevX{X}
2987   \def\mtc@noX{mtc@noX}
2988   \def\mtc@tmppn@name{\mtc@pns@abbrev\mtc@mttpn@abbrev pagenumbers}
2989   \expandafter\ifx\csname mtc@\mtc@pns@abbrev X\endcsname\mtc@noX
2990     \mtcPackageInfo[<I0022>]{minitoc}%
2991     {Page numbers are inhibited\MessageBreak for the #1s}
2992   \else
2993     \mtcPackageInfo[<I0021>]{minitoc}%
2994     {Page numbers are activated\MessageBreak for the #1s}
2995   \fi
2996   \csname\mtc@tmppn@name\endcsname{}
2997 \fi
2998 }

```

9.66.7 The \mtcsetrules command

This macro is very similar to \mtcsetpagenumbers and its syntax is the same:

```
\mtcsetrules{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 horizontal rules. *on* and *off* have many synonyms.

Hence the code is similar.

`\if@mtc@setrules@` We define some flags:

`\if@mtc@sru@ok@`

```
2999 \newif\if@mtc@setrules@ \@mtc@setrules@false
3000 \newif\if@mtc@sru@ok@
```

`\mtcsetrules` We define the user-level macro. If the first argument is a star, we call the internal macro `\@ifundefined` `\mtcsetrules@` for each type of mini-table available; else, we call this internal macro only once, for the specified type of mini-table.

```
3001 \newcommand{\mtcsetrules}[2]{%
3002   \expandafter\ifx\csname #1\endcsname\*\relax
3003   \@ifundefined{part}{}%
3004   {\mtcsetrules@{parttoc}{#2}}
3005   \mtcsetrules@{partlof}{#2}
3006   \mtcsetrules@{partlot}{#2}}
3007   \@ifundefined{chapter}{}%
3008   {\mtcsetrules@{minitoc}{#2}}
3009   \mtcsetrules@{minilof}{#2}
3010   \mtcsetrules@{minilot}{#2}}
3011   \@ifundefined{section}{}%
3012   {\mtcsetrules@{secttoc}{#2}}
3013   \mtcsetrules@{sectlof}{#2}
3014   \mtcsetrules@{sectlot}{#2}}
3015   \else
3016   \mtcsetrules@{#1}{#2}%
3017 \fi
3018 }
```

`\mtcsetrules@` Then the `\mtcsetrules@` internal macro, which has two arguments:

```
3019 \newcommand{\mtcsetrules@}[2]{%
```

`\mtc@mttru@abbrev` We process the first argument, a keyword of the `typetable` family and store the result in

`\if@mtc@setrules@` `\mtc@mttru@abbrev`:

`\mtc@rusw@abbrev`

```
\@nameuse 3020 \def\mtc@mttru@abbrev{X}
3021 \@mtc@setrules@true
3022 \def\mtc@rusw@abbrev{}
3023 \expandafter\ifx\csname mtc@typetable@#1\endcsname\relax
3024   \@mtc@setrules@false
3025   \def\mtc@rusw@abbrev{X}
3026   \def\mtc@mttru@abbrev{X}
3027   \mtcPackageError[<E0019>]{minitoc}%
3028     {\string\mtcsetrules \space has a wrong first argument
3029     \MessageBreak
```



```

3030      (#1)}%
3031      {It should be a mini-table type
3032      \MessageBreak
3033      (parttoc...sectlot)
3034      \MessageBreak
3035      Correct the source code.\MessageBreak
3036      Type <return> and rerun LaTeX}
3037 \else
3038   \edef\mtc@mttru@abbrev{\@nameuse{mtc@typetable@#1}}
3039 \fi

```

```

\if@mtc@sru@ok@ Then the second argument, a keyword of the YN family, and store the result in
\mtc@rusw@abbrev \mtc@rusw@abbrev. The name of the effective macro is built and the macro executed.
\mtc@mttru@abbrev
if@setrules@false 3040 \@mtc@sru@ok@true
\mtc@noX 3041 \expandafter\ifx\csname mtc@YN@#2\endcsname\relax
\mtc@tmppn@name 3042   \@mtc@sru@ok@false
3043   \def\mtc@rusw@abbrev{X}
3044   \def\mtc@mttru@abbrev{X}
3045   \@mtc@setrules@false
3046   \mtcPackageError[<E0020>]{minitoc}%
3047   {string\mtcsetrules \space has a wrong second argument
3048   \MessageBreak
3049   (#2)}%
3050   {It should be a boolean value (0/1, yes/no, on/off)
3051   \MessageBreak
3052   Correct the source code.\MessageBreak
3053   Type <return> and rerun LaTeX}
3054 \else
3055   \edef\mtc@rusw@abbrev{\@nameuse{mtc@YN@#2}}
3056   \def\mtc@rusw@abbrevX{X}
3057   \def\mtc@noX{mtc@noX}
3058   \def\mtc@tmppn@name{\mtc@rusw@abbrev\mtc@mttru@abbrev rule}
3059   \expandafter\ifx\csname mtc@\mtc@rusw@abbrev X\endcsname\mtc@noX
3060     \mtcPackageInfo[<I0008>]{minitoc}%
3061     {Horizontal rules are inhibited\MessageBreak for the #1s}
3062   \else
3063     \mtcPackageInfo[<I0007>]{minitoc}%
3064     {Horizontal rules are activated\MessageBreak for the #1s}
3065   \fi
3066   \csname\mtc@tmppn@name\endcsname{}
3067 \fi
3068 }

```

9.66.8 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 “thispage” is used.

A family (`ltypetable`) for the long names of the types of mini-tables:

```

3069 \@namedef{mtc@ltypetable@parttoc}{parttoc}\def\mtc@ltypetable@parttoc{parttoc}
3070 \@namedef{mtc@ltypetable@partlof}{partlof}\def\mtc@ltypetable@partlof{partlof}
3071 \@namedef{mtc@ltypetable@partlot}{partlot}\def\mtc@ltypetable@partlot{partlot}
3072 \@namedef{mtc@ltypetable@minitoc}{minitoc}\def\mtc@ltypetable@minitoc{minitoc}
3073 \@namedef{mtc@ltypetable@minilof}{minilof}\def\mtc@ltypetable@minilof{minilof}
3074 \@namedef{mtc@ltypetable@minilot}{minilot}\def\mtc@ltypetable@minilot{minilot}
3075 \@namedef{mtc@ltypetable@secttoc}{secttoc}\def\mtc@ltypetable@secttoc{secttoc}
3076 \@namedef{mtc@ltypetable@sectlof}{sectlof}\def\mtc@ltypetable@sectlof{sectlof}
3077 \@namedef{mtc@ltypetable@sectlot}{sectlot}\def\mtc@ltypetable@sectlot{sectlot}

```

A family (`featureparam`) for the type of feature:

```

3078 \@namedef{mtc@featureparam@before}{before}%
3079 \def\mtc@featureparam@before{before}
3080 \@namedef{mtc@featureparam@after}{after}%
3081 \def\mtc@featureparam@after{after}
3082 \@namedef{mtc@featureparam@pagestyle}{thispage}%
3083 \def\mtc@featureparam@pagestyle{thispage}

```

And a family (`ft3`) to add “style” if it is a “pagestyle” feature:

```

3084 \@namedef{mtc@ft3@before}{}\expandafter\def\csname mtc@ft3@before\endcsname{}
3085 \@namedef{mtc@ft3@after}{}\expandafter\def\csname mtc@ft3@after\endcsname{}
3086 \@namedef{mtc@ft3@pagestyle}{style}%
3087 \expandafter\def\csname mtc@ft3@pagestyle\endcsname{style}

```

The `\mtcsetfeature` command has the following syntax:

```
\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.

`\if@mtc@setfeature@` We define a flag and the `\mtcsetfeature` command, with three arguments:
`\mtcsetfeature`

```

3088 \newif\if@mtc@setfeature@\@mtc@setfeature@true
3089 \newcommand{\mtcsetfeature}[3]{%

```

`\mtc@mtfeat@abbrev` We process the first argument, a keyword of the `ltypetable` family, and store the result
`\mtc@featparam@abbrev` in `\mtc@mtfeat@abbrev`:

```

3090 \def\mtc@mtfeat@abbrev{X}
3091 \def\mtc@featparam@abbrev{X}
3092 \@mtc@setfeature@true

```

```

3093 \expandafter\ifx\csname mtc@ltypetable@#1\endcsname\relax
3094   \@mtc@setfeature@false
3095   \def\mtc@mtfeat@abbrev{X}
3096   \mtcPackageError[<E0011>]{minitoc}%
3097     {\string\mtcsetfeature \space has a wrong first argument
3098     \MessageBreak
3099     (#1).
3100     \MessageBreak
3101     It should be a mini-table type
3102     \MessageBreak
3103     (parttoc...sectlot)}}%
3104   {Correct the source code.\MessageBreak
3105   Type <return> and rerun LaTeX}
3106 \else
3107   \edef\mtc@mtfeat@abbrev{\@nameuse{mtc@ltypetable@#1}}
3108 \fi

```

`\if@mtc@setfeature@` The second argument is a keyword of the `featureparam` family, the result is stored in
`\mtc@featparam@abbrev` `\mtc@featparam@`; and the complement is computed from the first argument, interpreted
`\mtc@featparam@third` as a keyword of the `ft3` family and whose result is stored in `\mtc@featparam@third`.
`\@nameuse`

```

3109 \expandafter\ifx\csname mtc@featureparam@#2\endcsname\relax
3110   \@mtc@setfeature@false
3111   \def\mtc@featparam@abbrev{X}
3112   \def\mtc@featparam@third{X}
3113   \mtcPackageError[<E0012>]{minitoc}%
3114     {\string\mtcsetfeature \space has a wrong second argument
3115     \MessageBreak
3116     (#2).
3117     \MessageBreak
3118     It should be a feature param
3119     \MessageBreak
3120     (before, after, pagestyle)}}%
3121   {Correct the source code.\MessageBreak
3122   Type <return> and rerun LaTeX}
3123 \else
3124   \edef\mtc@featparam@abbrev{\@nameuse{mtc@featureparam@#2}}
3125   \edef\mtc@featparam@third{\@nameuse{mtc@ft3@#2}}
3126 \fi

```

`\if@mtc@setfeature@` The name of the effective macro is built by concatenating these three pieces (named
`\mtc@tmpfeat@name` `\mtc@featparam@abbrev`, `\mtc@mtfeat@abbrev`, and `\mtc@featparam@third`
`\mtc@featparam@abbrev` respectively), then this macro is executed:
`\mtc@mtfeat@abbrev`
`\mtc@featparam@third`

```

3127 \if@mtc@setfeature@
3128   \def\mtc@tmpfeat@name%
3129     {\mtc@featparam@abbrev\mtc@mtfeat@abbrev\mtc@featparam@third}
3130   \mtcPackageInfo[<I0014>]{minitoc}%
3131   {\string\mtcsetfeature\space redefines the macro\MessageBreak
3132   "\csname mtc@tmpfeat@name\endcsname" as\MessageBreak "\string#3"}

```

```

3133 \expandafter\def\csname\mtc@tmpfeat@name\endcsname{#3}
3134 \else
3135 \mtcPackageError[<E0023>]{minitoc}%
3136 {The macro \string\mtcsetfeature\space has incompatible\MessageBreak
3137 first (#1) and second (#2) arguments}%
3138 {Correct the source code.\MessageBreak
3139 Type <return> and rerun LaTeX}
3140 \fi}

```

9.66.9 The `\mtcsetdepth` command

This command is very similar to the `\mtcsettitle` command. Its syntax is almost identical:

```
\mtcsetdepth{mini-table}{depth}
```



The *mini-table* type is a keyword like *minitoc*. The *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 [86]).

`\if@mtc@setdepth@` First, we declare a flag, set true:

```
3141 \newif\if@mtc@setdepth@\@mtc@setdepth@true
```

`\mtcsetdepth` Then we define the `\mtcsetdepth` command, with two arguments:

```
3142 \newcommand{\mtcsetdepth}[2]{%
```

`\mtc@mtade@abbrev` We process the first argument, a keyword of the *ltypetable* family. The result is stored in `\mtc@mtade@abbrev`:

`\if@mtc@setdepth@`

`\@nameuse`

```

3143 \def\mtc@mtade@abbrev{X}
3144 \@mtc@setdepth@true
3145 \expandafter\ifx\csname mtc@ltypetable@#1\endcsname\relax
3146 \@mtc@setdepth@false
3147 \def\mtc@mtade@abbrev{X}
3148 \mtcPackageError[<E0009>]{minitoc}%
3149 {\string\mtcsetdepth \space has a wrong first argument
3150 \MessageBreak (#1).
3151 \MessageBreak
3152 It should be a mini-table type
3153 \MessageBreak
3154 (parttoc...sectlot)}%
3155 {Correct the source code.\MessageBreak

```

```

3156      Type <return> and rerun LaTeX}
3157 \else
3158   \edef\mtc@mtade@abbrev{\@nameuse{mtc@ltypetable@#1}}
3159 \fi

```

```

\if@mtc@setdepth@ And we construct the name of the effective counter and gave it the value:
\mtc@tmpde@name
\mtc@mtade@abbrev 3160 \if@mtc@setdepth@
\mtc@toks 3161   \def\mtc@tmpde@name{\mtc@mtade@abbrev depth}
\setcounter 3162   \expandafter\noexpand\@ifundefined{c@\mtc@mtade@abbrev depth}%
3163     {\mtcPackageError[<E0008>]{minitoc}%
3164      {\string\mtcsetdepth \space attempts to use\MessageBreak
3165       an undefined counter (#1depth).}%
3166      {Correct the source code.\MessageBreak
3167       Type <return> and rerun LaTeX}}}%
3168   \mtc@toks{#2} % trick for explicit message using \the.
3169   \mtcPackageInfo[<I0013>]{minitoc}%
3170   {\string\mtcsetdepth\space redefines the counter\MessageBreak
3171    "\mtc@tmpde@name" as "\the\mtc@toks"}}%
3172   \expandafter\csname c@\mtc@tmpde@name\endcsname=#2}%
3173 \else
3174   \mtcPackageError[<E0010>]{minitoc}%
3175   {\string\mtcsetdepth:\space Illegal type of table (#1)}%
3176   {Correct the source code.\MessageBreak
3177    Type <return> and rerun LaTeX}{\relax}
3178 \fi}% end of \mtcsetdepth

```

9.67 The mtchideinmaintoc environment and siblings

\if@mtc@Himtoc@ The flag \if@mtc@Himtoc@ is used to detect an incorrect imbrication of this environment:

```

3179 \newif\if@mtc@Himtoc@ \@mtc@Himtoc@false

```

```

\mtc@savetocdepth We define a macro \mtc@savetocdepth to save the current value of the counter
\mtc@restoretocdepth tocdepth. Then we define this environment, which inserts into the TOC file this com-
mtchideinmaintoc mand and commands of the form \setcounter{tocdepth}{...}. Note that \xdef is
\if@mtc@Himtoc@ necessary! It also save and restore the value of the counter tocdepth, as the optional
\mtc@sv@tocdepth argument is the hiding depth of the entries in the main TOC.
\arabic
\addtocontents 3180 \newcommand{\mtc@savetocdepth}{\xdef\mtc@sv@tocdepth{\arabic{tocdepth}}}%
\setcounter 3181 \newcommand{\mtc@restoretocdepth}{\setcounter{tocdepth}{\mtc@sv@tocdepth}}%
3182 \newenvironment{mtchideinmaintoc}[1][-1]%
3183 {\if@mtc@Himtoc@\mtcPackageError[<E0005>]{minitoc}%
3184  {Imbrication of mtchideinmaintoc environments}%
3185  {The hiding in main ToC could be incorrect}}\fi

```

```

3186 \global\@mtc@Himtoc@true
3187 \addtocontents{toc}{\protect\mtc@savetocdepth}%
3188 \addtocontents{toc}{\protect\setcounter{tocdepth}{#1}}}%
3189 {\if@mtc@Himtoc@else\mtcPackageError[<E0031>]{minitoc}%
3190   {Unbalanced mtchideinmaintoc environment}%
3191   {The hiding in main ToC could be incorrect}\fi
3192 \global\@mtc@Himtoc@false
3193 \addtocontents{toc}{\protect\mtc@restoretocdepth}}%

```

\AtBeginDocument The mtchideinmainlof and 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.

```

\mtc@sv@lofdepth
\mtc@sv@tocdepth First, for the list of figures:
\mtc@svf@tocdepth
\arabic 3194 \newif\if@mtc@Himlof@ \@mtc@Himlof@false
\addtocontents 3195 \AtBeginDocument{%
\setcounter 3196 \@ifundefined{c@lofdepth}{%
3197   \newenvironment{mtchideinmainlof}[1][-1]%
3198   {\if@mtc@Himlof@\mtcPackageError[<E0003>]{minitoc}%
3199     {Imbrication of mtchideinmainlof environments}%
3200     {The hiding in main LoF could be incorrect}\fi
3201   \global\@mtc@Himlof@true
3202   \def\mtc@sv@tocdepth{\arabic{tocdepth}}%
3203   \def\mtc@sv@lofdepth{\arabic{tocdepth}}%
3204   \addtocontents{lof}{\protect\mtc@savetocdepth}%
3205   \addtocontents{lof}{\protect\setcounter{tocdepth}{#1}}}%
3206   {\if@mtc@Himlof@else\mtcPackageError[<E0029>]{minitoc}%
3207     {Unbalanced mtchideinmainlof environment}%
3208     {The hiding in main LoF could be incorrect}\fi
3209   \global\@mtc@Himlof@false
3210   \addtocontents{lof}{\protect\mtc@restoretocdepth}%
3211 }}%
3212 {%
3213   \newcommand{\mtc@savelofdepth}{\xdef\mtc@sv@lofdepth{\arabic{lofdepth}}}%
3214   \newcommand{\mtc@restorelofdepth}{\setcounter{lofdepth}{\mtc@sv@lofdepth}}%
3215   \newenvironment{mtchideinmainlof}[1][-1]%
3216   {\if@mtc@Himlof@\mtcPackageError[<E0003>]{minitoc}%
3217     {Imbrication of mtchideinmainlof environments}%
3218     {The hiding in main LoF could be incorrect}\fi
3219   \global\@mtc@Himlof@true
3220   \addtocontents{lof}{\protect\mtc@savelofdepth}%
3221   \addtocontents{lof}{\protect\setcounter{tocdepth}{#1}}}%
3222   {\if@mtc@Himlof@else\mtcPackageError[<E0029>]{minitoc}%
3223     {Unbalanced mtchideinmainlof environment}%
3224     {The hiding in main LoF could be incorrect}\fi
3225   \global\@mtc@Himlof@false
3226   \addtocontents{lof}{\protect\mtc@restoretocdepth}}}%

```

```

\AtBeginDocument Then for the list of tables:
\if@mtc@Himlot@
mtchideinmainlot 3227 \newif\if@mtc@Himlot@ \@mtc@Himlot@false
\mtc@sv@lotdepth 3228 \AtBeginDocument{%
\mtc@sv@tocdepth 3229 \@ifundefined{c@lotdepth}{%
\mtc@svt@tocdepth 3230 \newenvironment{mtchideinmainlot}[1][-1]%
3231 {\if@mtc@Himlot@\mtcPackageError[<E0004>]{minitoc}%
\arabic 3232 {Imbrication of mtchideinmainlot environments}%
3233 {The hiding in main LoT could be incorrect}\fi
\addtocontents 3234 \global\@mtc@Himlot@true
3235 \def\mtc@sv@tocdepth{\arabic{tocdepth}}%
3236 \def\mtc@sv@lotdepth{\arabic{lotdepth}}%
3237 \addtocontents{lot}{\protect\mtc@savetocdepth}%
3238 \addtocontents{lot}{\protect\setcounter{tocdepth}{#1}}}%
3239 {\if@mtc@Himlot@\else\mtcPackageError[<E0030>]{minitoc}%
3240 {Unbalanced mtchideinmainlot environment}%
3241 {The hiding in main LoT could be incorrect}\fi
3242 \global\@mtc@Himlot@false
3243 \addtocontents{lot}{\protect\mtc@restoretocdepth}%
3244 }}%
3245 {%
3246 \newcommand{\mtc@savelotdepth}{\xdef\mtc@sv@lotdepth{\arabic{lotdepth}}}%
3247 \newcommand{\mtc@restorelotdepth}{\setcounter{lotdepth}{\mtc@sv@lotdepth}}%
3248 \newenvironment{mtchideinmainlot}[1][-1]%
3249 {\if@mtc@Himlot@\mtcPackageError[<E0004>]{minitoc}%
3250 {Imbrication of mtchideinmainlot environments}%
3251 {The hiding in main LoT could be incorrect}\fi
3252 \global\@mtc@Himlot@true
3253 \addtocontents{lot}{\protect\mtc@savelotdepth}%
3254 \addtocontents{lot}{\protect\setcounter{lotdepth}{#1}}}%
3255 {\if@mtc@Himlot@\else\mtcPackageError[<E0030>]{minitoc}%
3256 {Unbalanced mtchideinmainlot environment}%
3257 {The hiding in main LoT could be incorrect}\fi
3258 \global\@mtc@Himlot@false
3259 \addtocontents{lot}{\protect\mtc@restorelotdepth}}}

```

9.68 Fixing the “Glossary” entry in the TOC

This macro is complex. Its syntax is:

```
\mtcfixglossary[part|chapter|section]
```

\@ifundefined Depending of 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:

```

3260 \ifundefined{chapter}{%
3261   \ifundefined{section}%
3262     {\mtcPackageWarningNoLine[<W0001>]{minitoc}%
3263      {\string\chapter\space and \string\section\space are undefined.%
3264       \MessageBreak Cannot use \string\mtcfixglossary \space
3265        without\MessageBreak optional argument [part]}}%
3266     \@ifundefined{part}%
3267       {\mtcPackageError[<E0001>]{minitoc}%
3268        {But \string\part\space is undefined}%
3269        {\string\mtcfixglossary\space not usable}}%
3270       {\mtcPackageWarningNoLine[<W0006>]{minitoc}%
3271        {\string\mtcfixglossary\space can only be used\MessageBreak
3272         with the [part] optional argument,\MessageBreak
3273         which becomes the default}}%
3274       \def\mtc@glofix@level{part}%
3275     }%
3276     {\def\mtc@glofix@level{section}}}%
3277 {\def\mtc@glofix@level{chapter}}

```

\if@mtcfixglossary@ Then we define a flag (\if@mtcfixglossary@) and the command \mtcfixglossary, which adds the necessary lines in the TOC, the LOF and the LOT.

\mtcfixglossary

\addcontentsline

```

3278 \newif\if@mtcfixglossary@ \mtcfixglossary@false
3279 \newcommand{\mtcfixglossary}[1][\mtc@glofix@level]{%
3280   \@mtcfixglossary@false
3281   \expandafter%
3282   \ifx\csname #1\endcsname\part\relax\mtcfixglossary@true\fi
3283   \expandafter%
3284   \ifx\csname #1\endcsname\chapter\relax\mtcfixglossary@true\fi
3285   \expandafter%
3286   \ifx\csname #1\endcsname\section\relax\mtcfixglossary@true\fi
3287   \if@mtcfixglossary@
3288     \addcontentsline{lof}{x\mtc@glofix@level}{}%
3289     \addcontentsline{lot}{x\mtc@glofix@level}{}%
3290     \csname mtcadd\mtc@glofix@level\endcsname\relax
3291   \else
3292     \mtcPackageError[<E0026>]{minitoc}%
3293     {The optional argument of \string\mtcfixglossary
3294      \MessageBreak
3295      is wrong}%
3296     {It must be omitted (\mtc@glofix@level),
3297      or be part, chapter or section}%
3298   \fi
3299 }%

```


9.69 Fixing the “Index” entry in the TOC

This macro is complex. Its syntax is:

`\mtcfixindex[part|chapter|section]`

`\@ifundefined` Depending of 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 `\mtc@ixfix@level` 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:

```

3300 \@ifundefined{chapter}{%
3301   \@ifundefined{section}%
3302     {\mtcPackageWarningNoLine[<W0002>]{minitoc}%
3303      {\string\chapter\space and \string\section\space are undefined.%
3304       \MessageBreak Cannot use \string\mtcfixindex \space
3305        without\MessageBreak optional argument [part]}}%
3306     \@ifundefined{part}%
3307       {\mtcPackageError[<E0002>]{minitoc}%
3308        {But \string\part\space is undefined}%
3309        {\string\mtcfixindex\space not usable}}%
3310       {\mtcPackageWarningNoLine[<W0007>]{minitoc}%
3311        {\string\mtcfixindex\space can only be used with\MessageBreak
3312         the [part] optional argument,\MessageBreak
3313         which becomes the default}}%
3314       \def\mtc@ixfix@level{part}%
3315     }%
3316     {\def\mtc@ixfix@level{section}}}%
3317 {\def\mtc@ixfix@level{chapter}}
```

`\if@mtcfixindex@` Then we define a flag and the command `\mtcfixindex`, which adds the necessary lines
`\mtcfixindex` in the TOC, the LOF and the LOT.
`\addcontentsline`

```

3318 \newif\if@mtcfixindex@ \@mtcfixindex@false
3319 \newcommand{\mtcfixindex}[1][\mtc@ixfix@level]{%
3320   \@mtcfixindex@false
3321   \expandafter%
3322   \ifx\csname #1\endcsname\part\relax\@mtcfixindex@true\fi
3323   \expandafter%
3324   \ifx\csname #1\endcsname\chapter\relax\@mtcfixindex@true\fi
3325   \expandafter%
3326   \ifx\csname #1\endcsname\section\relax\@mtcfixindex@true\fi
3327   \if@mtcfixindex@
3328     \addcontentsline{lof}{x\mtc@ixfix@level}{}%
3329     \addcontentsline{lot}{x\mtc@ixfix@level}{}%
3330     \csname mtcadd\mtc@ixfix@level\endcsname\relax
3331   \else
3332     \mtcPackageError[<E0027>]{minitoc}%

```

```

3333     {The optional argument of \string\mtcfixindex
3334     \MessageBreak
3335     is wrong}%
3336     {It must be omitted (\mtc@ixfix@level),
3337     or be part, chapter or section}%
3338 \fi
3339 }%

```

9.70 The \mtcselectlanguage command

`\mtcselectlanguage` 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 `\if@mtc@insellang@` `\IfFileExists` `\if@mtc@insellang@` is true while we are in this macro.

```

\@input
3340 \newif\if@mtc@insellang@ \@mtc@insellang@false
3341 \def\mtcselectlanguage#1{%
3342     \InputIfFileExists{#1.mld}%
3343     {\mtcPackageInfo[<I0010>]{minitoc}{The #1 language is selected.%
3344     \MessageBreak}}%
3345     {\mtcPackageError[<E0006>]{minitoc}%
3346     {#1 is not a known language, \MessageBreak
3347     #1.mld not found. \MessageBreak
3348     Command ignored}}%
3349     {See the minitoc documentation. \MessageBreak
3350     Correct the source using a valid language name. \MessageBreak
3351     Press RETURN}}%
3352 }

```

9.71 The \mtcloadmlo internal command

`\mtcloadmlo` This command loads a minitoc language object file *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 filecontents environments in the minitoc.ins file. But first, we verify that this .mlo file exists.



This command should not be directly invoked by the user. This is verified via the flag `\if@mtc@insellang@`.

```

3353 \def\mtcloadmlo#1{%
3354     \if@mtc@insellang@
3355     \InputIfFileExists{#1.mlo}%
3356     {\mtcPackageInfo[<I0011>]{minitoc}%
3357     {#1 minitoc language object selected.\MessageBreak}}%
3358     {\mtcPackageError[<E0007>]{minitoc}%

```

```

3359      {#1 is not a known minitoc\MessageBreak
3360      language object file (.mlo), \MessageBreak
3361      #1.mlo not found. \MessageBreak
3362      Command ignored}%
3363      {See the minitoc documentation. \MessageBreak
3364      Correct the source using a valid language name. \MessageBreak
3365      Press RETURN}}%
3366  \else
3367    \mtcPackageError[<E0032>]{minitoc}%
3368    {You are using the \string\mtcloadmlo\space command\MessageBreak
3369    outside of a .mld file}%
3370    {It will be ignored}
3371    \@mtc@insellang@false
3372  \fi
3373 }

```

9.72 The “coffee breaks”

\addcoffeeline For the minutes package [145] (by Knut LICKERT), we need some commands to insert special entries, undotted, in the TOC to mark “coffee breaks” ☕ in a conference. Hence \addtocontents we define \addcoffeeline, \coffeeline and \l@coffee, and internal commands \coffeeline \l@coffee analog to the standard internal commands to format the TOC.

```

\@Undottedtocline
3374 \def\addcoffeeline#1#2#3{%
3375   \addtocontents{#1}{\protect\coffeeline{#2}{#3}{\null}}}
3376 \def\coffeeline#1{\csname l@#1\endcsname}
3377 \newcommand*\l@coffee{\@Undottedtocline{1}{1.5em}{2.3em}}

```

9.73 Initialization of counters

\AtBeginDocument At the beginning of the document, we initialize the absolute counters for parts, chapters and sections, if they are defined.

```

\@ifundefined
\setcounter
3378 \AtBeginDocument{%
3379   \@ifundefined{c@ptc}{}{\setcounter{ptc}{0}}
3380   \@ifundefined{c@mtc}{}{\setcounter{mtc}{0}}
3381   \@ifundefined{c@stc}{}{\setcounter{stc}{0}}}

```

9.74 Declarations for simple options

These options are just setting a flag.

9.74.1 Options **tight** and **loose**, **k-tight** and **k-loose**

```
\DeclareOption These options influence the interline separation in the mini-tables.
  \iftightmtc
  \kiftightmtc 3382 \DeclareOption{tight}{\tightmtctrue}
                 3383 \DeclareOption{loose}{\tightmtcfalse} % default
                 3384 \DeclareOption{k-tight}{\ktightmtctrue}
                 3385 \DeclareOption{k-loose}{\ktightmtcfalse} % default
```

9.74.2 Options **checkfiles** and **nocheckfiles**

```
\DeclareOption These options activate or inhibit the checking for empty mini-table files.
\if@mtc@checkfiles
                 3386 \DeclareOption{checkfiles}{\@mtc@checkfilestrue} % default
                 3387 \DeclareOption{nocheckfiles}{\@mtc@checkfilesfalse}
```

9.74.3 Options **dotted** and **undotted**

```
\DeclareOption These options activate or inhibit the leaders (lines of dots) in the mini-tables.
\ifundottedmtc
                 3388 \DeclareOption{undotted}{\undottedmtctrue}
                 3389 \DeclareOption{dotted}{\undottedmtcfalse} % default
```

9.74.4 Option **notoccite**

```
\DeclareOption This option will later load the notoccite package [8].
\if@mtc@notoccite@
                 3390 \DeclareOption{notoccite}{\@mtc@notoccite@true}
```

9.74.5 Option **shorttext**

```
\DeclareOption This option forces the use of short extensions.
\if@longextensions@
                 3391 \DeclareOption{shorttext}{\@longextensions@false
                 3392 \mtcPackageWarningNoLine[<W0020>]{minitoc}%
                 3393 {You have forced the use of short extensions}}
```

9.75 The `insection` option

`\if@mtc@ss@insection@` This option is available only if `\chapter` is not defined and `\section` defined. It is to
`\@ifundefined` be revised when chapter/section level commands will ever be allowed together, sometime
`\DeclareOption` in the far away future, with a lot of luck (and work).

```

3394 \newif\if@mtc@ss@insection@ \@mtc@ss@insection@false
3395 \@ifundefined{chapter}{%
3396   \@ifundefined{section}{\DeclareOption{insection}{%
3397     \mtcPackageError[<E0035>]{minitoc}%
3398     {You have used the 'insection' option in
3399     \MessageBreak
3400     a document where chapters are defined.
3401     \MessageBreak
3402     This is not compatible: option ignored.}%
3403     {Remove this option.\MessageBreak
3404     Type <return> and rerun LaTeX}
3405     }}%
3406     {%
3407     \DeclareOption{insection}%
3408     {\@mtc@ss@insection@true}%
3409     }%
3410 }}

```

9.76 The `listfiles` and `nolistfiles` options

`\if@mtc@listfiles@` The `listfiles` option creates a file containing a list of the auxiliary files created by
`\DeclareOption` the `minitoc` package. This file is named `<document>.maf`. The `nolistfiles` option
inhibits this listing (default).

```

3411 \newif\if@mtc@listfiles@ \@mtc@listfiles@false
3412 \DeclareOption{listfiles}{\@mtc@listfiles@true}
3413 \DeclareOption{nolistfiles}{\@mtc@listfiles@false}

```

9.77 Language options

`\DeclareOption` Each language option reads the corresponding `language.mld` file via the specialized
`\mtcselectlanguage` macro `\mtcselectlanguage`. 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 can load another one
(as `american.mld` loads `english.mld`).

```

3414 \DeclareOption{acadian}{\mtcselectlanguage{acadian}}%
3415 \DeclareOption{acadien}{\mtcselectlanguage{acadien}}%
3416 \DeclareOption{afrikaan}{\mtcselectlanguage{afrikaan}}%
3417 \DeclareOption{afrikaans}{\mtcselectlanguage{afrikaans}}%
3418 \DeclareOption{albanian}{\mtcselectlanguage{albanian}}%
3419 \DeclareOption{american}{\mtcselectlanguage{american}}%
3420 \DeclareOption{arab}{\mtcselectlanguage{arab}}%
3421 \DeclareOption{arab2}{\mtcselectlanguage{arab2}}%
3422 \DeclareOption{arabi}{\mtcselectlanguage{arabi}}%
3423 \DeclareOption{arabic}{\mtcselectlanguage{arabic}}%
3424 \DeclareOption{armenian}{\mtcselectlanguage{armenian}}%
3425 \DeclareOption{australian}{\mtcselectlanguage{australian}}%
3426 \DeclareOption{austrian}{\mtcselectlanguage{austrian}}%
3427 \DeclareOption{bahasa}{\mtcselectlanguage{bahasa}}%
3428 \DeclareOption{bahasai}{\mtcselectlanguage{bahasai}}%
3429 \DeclareOption{bahasam}{\mtcselectlanguage{bahasam}}%
3430 \DeclareOption{bangla}{\mtcselectlanguage{bangla}}%
3431 \DeclareOption{basque}{\mtcselectlanguage{basque}}%
3432 \DeclareOption{bicig}{\mtcselectlanguage{bicig}}%
3433 \DeclareOption{bicig2}{\mtcselectlanguage{bicig2}}%
3434 \DeclareOption{bicig3}{\mtcselectlanguage{bicig3}}%
3435 \DeclareOption{bithe}{\mtcselectlanguage{bithe}}%
3436 \DeclareOption{brazil}{\mtcselectlanguage{brazil}}%
3437 \DeclareOption{brazilian}{\mtcselectlanguage{brazilian}}%
3438 \DeclareOption{breton}{\mtcselectlanguage{breton}}%
3439 \DeclareOption{british}{\mtcselectlanguage{british}}%
3440 \DeclareOption{bulgarian}{\mtcselectlanguage{bulgarian}}%
3441 \DeclareOption{bulgarianb}{\mtcselectlanguage{bulgarianb}}%
3442 \DeclareOption{buryat}{\mtcselectlanguage{buryat}}%
3443 \DeclareOption{buryat2}{\mtcselectlanguage{buryat2}}%
3444 \DeclareOption{canadian}{\mtcselectlanguage{canadian}}%
3445 \DeclareOption{canadien}{\mtcselectlanguage{canadien}}%
3446 \DeclareOption{castillan}{\mtcselectlanguage{castillan}}%
3447 \DeclareOption{castillian}{\mtcselectlanguage{castillian}}%
3448 \DeclareOption{catalan}{\mtcselectlanguage{catalan}}%
3449 \DeclareOption{chinese1}{\mtcselectlanguage{chinese1}}%
3450 \DeclareOption{chinese2}{\mtcselectlanguage{chinese2}}%
3451 \DeclareOption{croatian}{\mtcselectlanguage{croatian}}%
3452 \DeclareOption{czech}{\mtcselectlanguage{czech}}%
3453 \DeclareOption{danish}{\mtcselectlanguage{danish}}%
3454 \DeclareOption{dutch}{\mtcselectlanguage{dutch}}%
3455 \DeclareOption{english}{\mtcselectlanguage{english}}%
3456 \DeclareOption{english1}{\mtcselectlanguage{english1}}%
3457 \DeclareOption{english2}{\mtcselectlanguage{english2}}%
3458 \DeclareOption{esperant}{\mtcselectlanguage{esperant}}%
3459 \DeclareOption{esperanto}{\mtcselectlanguage{esperanto}}%
3460 \DeclareOption{estonian}{\mtcselectlanguage{estonian}}%
3461 \DeclareOption{ethiopia}{\mtcselectlanguage{ethiopia}}%
3462 \DeclareOption{ethiopian}{\mtcselectlanguage{ethiopian}}%
3463 \DeclareOption{ethiopian2}{\mtcselectlanguage{ethiopian2}}%
3464 \DeclareOption{farsi1}{\mtcselectlanguage{farsi1}}%
3465 \DeclareOption{farsi2}{\mtcselectlanguage{farsi2}}%
3466 \DeclareOption{farsi3}{\mtcselectlanguage{farsi3}}%
3467 \DeclareOption{finnish}{\mtcselectlanguage{finnish}}%

```

```

3468 \DeclareOption{finnish2}{\mtcselectlanguage{finnish2}}%
3469 \DeclareOption{français}{\mtcselectlanguage{français}}%
3470 \DeclareOption{french}{\mtcselectlanguage{french}}%
3471 \DeclareOption{french1}{\mtcselectlanguage{french1}}%
3472 \DeclareOption{french2}{\mtcselectlanguage{french2}}%
3473 \DeclareOption{frenchb}{\mtcselectlanguage{frenchb}}%
3474 \DeclareOption{frenchle}{\mtcselectlanguage{frenchle}}%
3475 \DeclareOption{frenchpro}{\mtcselectlanguage{frenchpro}}%
3476 \DeclareOption{galician}{\mtcselectlanguage{galician}}%
3477 \DeclareOption{german}{\mtcselectlanguage{german}}%
3478 \DeclareOption{germanb}{\mtcselectlanguage{germanb}}%
3479 \DeclareOption{germanb2}{\mtcselectlanguage{germanb2}}%
3480 \DeclareOption{greek}{\mtcselectlanguage{greek}}%
3481 \DeclareOption{greek-mono}{\mtcselectlanguage{greek-mono}}%
3482 \DeclareOption{greek-polydemo}{\mtcselectlanguage{greek-polydemo}}%
3483 \DeclareOption{greek-polykatha}{\mtcselectlanguage{greek-polykatha}}%
3484 \DeclareOption{guarani}{\mtcselectlanguage{guarani}}%
3485 \DeclareOption{hangul1}{\mtcselectlanguage{hangul1}}%
3486 \DeclareOption{hangul2}{\mtcselectlanguage{hangul2}}%
3487 \DeclareOption{hangul3}{\mtcselectlanguage{hangul3}}%
3488 \DeclareOption{hangul4}{\mtcselectlanguage{hangul4}}%
3489 \DeclareOption{hangul-u8}{\mtcselectlanguage{hangul-u8}}%
3490 \DeclareOption{hanja1}{\mtcselectlanguage{hanja1}}%
3491 \DeclareOption{hanja2}{\mtcselectlanguage{hanja2}}%
3492 \DeclareOption{hanja-u8}{\mtcselectlanguage{hanja-u8}}%
3493 \DeclareOption{hebrew}{\mtcselectlanguage{hebrew}}%
3494 \DeclareOption{hebrew2}{\mtcselectlanguage{hebrew2}}%
3495 \DeclareOption{hungarian}{\mtcselectlanguage{hungarian}}%
3496 \DeclareOption{icelandic}{\mtcselectlanguage{icelandic}}%
3497 \DeclareOption{indon}{\mtcselectlanguage{indon}}%
3498 \DeclareOption{indonesian}{\mtcselectlanguage{indonesian}}%
3499 \DeclareOption{interlingua}{\mtcselectlanguage{interlingua}}%
3500 \DeclareOption{irish}{\mtcselectlanguage{irish}}%
3501 \DeclareOption{italian}{\mtcselectlanguage{italian}}%
3502 \DeclareOption{italian2}{\mtcselectlanguage{italian2}}%
3503 \DeclareOption{japanese}{\mtcselectlanguage{japanese}}%
3504 \DeclareOption{japanese2}{\mtcselectlanguage{japanese2}}%
3505 \DeclareOption{japanese3}{\mtcselectlanguage{japanese3}}%
3506 \DeclareOption{japanese4}{\mtcselectlanguage{japanese4}}%
3507 \DeclareOption{japanese5}{\mtcselectlanguage{japanese5}}%
3508 \DeclareOption{kannada}{\mtcselectlanguage{kannada}}%
3509 \DeclareOption{khalkha}{\mtcselectlanguage{khalkha}}%
3510 \DeclareOption{latin}{\mtcselectlanguage{latin}}%
3511 \DeclareOption{latin2}{\mtcselectlanguage{latin2}}%
3512 \DeclareOption{latvian}{\mtcselectlanguage{latvian}}%
3513 \DeclareOption{letton}{\mtcselectlanguage{letton}}%
3514 \DeclareOption{lithuanian}{\mtcselectlanguage{lithuanian}}%
3515 \DeclareOption{lowersorbian}{\mtcselectlanguage{lowersorbian}}%
3516 \DeclareOption{lsorbian}{\mtcselectlanguage{lsorbian}}%
3517 \DeclareOption{magyar}{\mtcselectlanguage{magyar}}%
3518 \DeclareOption{magyar2}{\mtcselectlanguage{magyar2}}%
3519 \DeclareOption{magyar3}{\mtcselectlanguage{magyar3}}%
3520 \DeclareOption{malay}{\mtcselectlanguage{malay}}%
3521 \DeclareOption{malayalam-keli}{\mtcselectlanguage{malayalam-keli}}%

```

```

3522 \DeclareOption{malayalam-omega}{\mtcselectlanguage{malayalam-omega}}%
3523 \DeclareOption{malayalam-rachana}{\mtcselectlanguage{malayalam-rachana}}%
3524 \DeclareOption{malayalam-rachana2}{\mtcselectlanguage{malayalam-rachana2}}%
3525 \DeclareOption{manju}{\mtcselectlanguage{manju}}%
3526 \DeclareOption{meyalu}{\mtcselectlanguage{meyalu}}%
3527 \DeclareOption{mongol}{\mtcselectlanguage{mongol}}%
3528 \DeclareOption{naustrian}{\mtcselectlanguage{naustrian}}%
3529 \DeclareOption{newzealand}{\mtcselectlanguage{newzealand}}%
3530 \DeclareOption{ngerman}{\mtcselectlanguage{ngerman}}%
3531 \DeclareOption{ngermanb}{\mtcselectlanguage{ngermanb}}%
3532 \DeclareOption{ngermanb2}{\mtcselectlanguage{ngermanb2}}%
3533 \DeclareOption{norsk}{\mtcselectlanguage{norsk}}%
3534 \DeclareOption{norsk2}{\mtcselectlanguage{norsk2}}%
3535 \DeclareOption{nynorsk2}{\mtcselectlanguage{nynorsk2}}%
3536 \DeclareOption{polish}{\mtcselectlanguage{polish}}%
3537 \DeclareOption{polish2}{\mtcselectlanguage{polish2}}%
3538 \DeclareOption{polski}{\mtcselectlanguage{polski}}%
3539 \DeclareOption{portuges}{\mtcselectlanguage{portuges}}%
3540 \DeclareOption{portuguese}{\mtcselectlanguage{portuguese}}%
3541 \DeclareOption{romanian}{\mtcselectlanguage{romanian}}%
3542 \DeclareOption{romanian2}{\mtcselectlanguage{romanian2}}%
3543 \DeclareOption{romanian3}{\mtcselectlanguage{romanian3}}%
3544 \DeclareOption{russian}{\mtcselectlanguage{russian}}%
3545 \DeclareOption{russianb}{\mtcselectlanguage{russianb}}%
3546 \DeclareOption{russianc}{\mtcselectlanguage{russianc}}%
3547 \DeclareOption{russian2m}{\mtcselectlanguage{russian2m}}%
3548 \DeclareOption{russian2o}{\mtcselectlanguage{russian2o}}%
3549 \DeclareOption{russian-cca}{\mtcselectlanguage{russian-cca}}%
3550 \DeclareOption{russian-cca1}{\mtcselectlanguage{russian-cca1}}%
3551 \DeclareOption{russian-lh}{\mtcselectlanguage{russian-lh}}%
3552 \DeclareOption{russian-lhcyralt}{\mtcselectlanguage{russian-lhcyralt}}%
3553 \DeclareOption{russian-lhcyrkoi}{\mtcselectlanguage{russian-lhcyrkoi}}%
3554 \DeclareOption{russian-lhcyrwin}{\mtcselectlanguage{russian-lhcyrwin}}%
3555 \DeclareOption{samin}{\mtcselectlanguage{samin}}%
3556 \DeclareOption{scottish}{\mtcselectlanguage{scottish}}%
3557 \DeclareOption{serbian}{\mtcselectlanguage{serbian}}%
3558 \DeclareOption{serbianc}{\mtcselectlanguage{serbianc}}%
3559 \DeclareOption{slovak}{\mtcselectlanguage{slovak}}%
3560 \DeclareOption{slovene}{\mtcselectlanguage{slovene}}%
3561 \DeclareOption{spanish}{\mtcselectlanguage{spanish}}%
3562 \DeclareOption{spanish2}{\mtcselectlanguage{spanish2}}%
3563 \DeclareOption{spanish3}{\mtcselectlanguage{spanish3}}%
3564 \DeclareOption{spanish4}{\mtcselectlanguage{spanish4}}%
3565 \DeclareOption{swedish}{\mtcselectlanguage{swedish}}%
3566 \DeclareOption{swedish2}{\mtcselectlanguage{swedish2}}%
3567 \DeclareOption{thai}{\mtcselectlanguage{thai}}%
3568 \DeclareOption{turkish}{\mtcselectlanguage{turkish}}%
3569 \DeclareOption{uighur}{\mtcselectlanguage{uighur}}%
3570 \DeclareOption{uighur2}{\mtcselectlanguage{uighur2}}%
3571 \DeclareOption{uighur3}{\mtcselectlanguage{uighur3}}%
3572 \DeclareOption{UKenglish}{\mtcselectlanguage{UKenglish}}%
3573 \DeclareOption{ukraine}{\mtcselectlanguage{ukraine}}%
3574 \DeclareOption{ukrainian}{\mtcselectlanguage{ukrainian}}%
3575 \DeclareOption{uppersorbian}{\mtcselectlanguage{uppersorbian}}%

```



```

3576 \DeclareOption{USenglish}{\mtcselectlanguage{USenglish}}%
3577 \DeclareOption{usorbrian}{\mtcselectlanguage{usorbrian}}%
3578 \DeclareOption{vietnam}{\mtcselectlanguage{vietnam}}%
3579 \DeclareOption{vietnamese}{\mtcselectlanguage{vietnamese}}%
3580 \DeclareOption{welsh}{\mtcselectlanguage{welsh}}%
3581 \DeclareOption{xalx}{\mtcselectlanguage{xalx}}%
3582 \DeclareOption{xalx2}{\mtcselectlanguage{xalx2}}%
3583 \DeclareOption{xalx3}{\mtcselectlanguage{xalx3}}%

```

9.78 The hints option

`\DeclareOption` We declare the hints (default) and nohints options:
`\if@mtc@hints@`

```

3584 \DeclareOption{hints}{\@mtc@hints@true}
3585 \DeclareOption{nohints}{\@mtc@hints@false}

```

`\mtc@hints@begindoc` The hints option is made of three parts: the first, `\mtc@hints@begindoc`, is executed
`\AtBeginDocument` via `\AtBeginDocument` and looks if some packages or classes are loaded and 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.

`\mtc@hints@enddoc` The third and last part, `\mtc@hints@enddoc`, is executed via `\AtEndDocument` and
`\AtEndDocument` examines some flags set by the first and the second parts. Then, if necessary, it writes some infos in the .log file and/or warnings on the screen and in the .log file. Note that the hints option does not signal errors, only infos and warnings, so it does not stop the L^AT_EX run.

9.78.1 First part: `\mtc@hints@begindoc`

`\if@mtc@abstract@loaded@` We declare some flags and the first part of the hints option (for `\AtBeginDocument`):
`\mtc@hints@begindoc`

```

\if@mtc@toc@used@ 3586 \newif\if@mtc@abstract@loaded@ \@mtc@abstract@loaded@false
\if@mtc@lof@used@ 3587 \newif\if@mtc@toc@used@ \global\@mtc@toc@used@false
\if@mtc@lot@used@ 3588 \newif\if@mtc@lof@used@ \global\@mtc@lof@used@false
3589 \newif\if@mtc@lot@used@ \global\@mtc@lot@used@false
3590 \def\mtc@hints@begindoc{%
3591 \mtcPackageInfo[<I0049>]{minitoc(hints)}%
3592 {==> You requested the hints option\MessageBreak
3593 Some hints are eventually given below\@gobble}%

```

9.78.1.1 Hint about the alphanum package

`\@ifpackageloaded` We test the presence of the alphanum package (part of the jura class [77]), and emit a
`\if@mtc@hints@given@` warning, because this package is incompatible with minitoc:

```

3594 \@ifpackageloaded{alphanum}%
3595     {\@mtc@hints@given@true
3596       \mtcPackageWarningNoLine[<W0025>]{minitoc(hints)}%
3597       {--- The alphanum package is loaded. \MessageBreak
3598         It is incompatible\MessageBreak with the minitoc package}}}%

```

9.78.1.2 Hint about the appendix package

`\@ifpackageloaded` We test the presence of the appendix package [209]:
`\if@mtc@hints@given@`

```

3599 \@ifpackageloaded{appendix}{%
3600   \@mtc@hints@given@true
3601   \mtcPackageInfo[<I0042>]{minitoc(hints)}%
3602   {--- The appendix package is loaded. \MessageBreak
3603     See the minitoc package documentation\MessageBreak
3604     for specific precautions\@gobble}}}%

```

9.78.1.3 Hint about the tocbibind package

`\@ifpackageloaded` We test the presence of the tocbibind package [210]:
`\if@mtc@hints@given@`

```

3605 \@ifpackageloaded{tocbibind}%
3606     {\@mtc@hints@given@true
3607       \mtcPackageInfo[<I0046>]{minitoc(hints)}%
3608       {--- The tocbibind package is loaded. \MessageBreak
3609         See the minitoc package documentation\MessageBreak
3610         for specific precautions\@gobble}}}%

```

9.78.1.4 Hint about the KOMA-Script classes

`\@ifclassloaded` We test the presence of each minitoc-compatible KOMA-Script class [164]:
`\if@mtc@hints@given@`

```

3611 \@ifclassloaded{scrbook}%
3612     {\@mtc@hints@given@true
3613       \mtcPackageInfo[<I0043>]{minitoc(hints)}%
3614       {--- The KOMAScript scrbook class is loaded. \MessageBreak
3615         See the minitoc package documentation\MessageBreak
3616         for specific precautions\@gobble}}}%

```

```

3617 \@ifclassloaded{scrreprt}%
3618   {\@mtc@hints@given@true
3619     \mtcPackageInfo[<I0043>]{minitoc(hints)}%
3620     {--- The KOMAScript scrreprt class is loaded. \MessageBreak
3621       See the minitoc package documentation\MessageBreak
3622       for specific precautions\@gobble}}}%
3623 \@ifclassloaded{scrartcl}%
3624   {\@mtc@hints@given@true
3625     \mtcPackageInfo[<I0043>]{minitoc(hints)}%
3626     {--- The KOMAScript scrartcl class is loaded. \MessageBreak
3627       See the minitoc package documentation\MessageBreak
3628       for specific precautions\@gobble}}}%

```

9.78.1.5 Hint about the tocloft package

\@ifpackageloaded We test the presence of the tocloft package [207]:
\if@mtc@hints@given@

```

3629 \@ifpackageloaded{tocloft}%
3630   {\@mtc@hints@given@true
3631     \mtcPackageInfo[<I0047>]{minitoc(hints)}%
3632     {--- The tocloft package is loaded. \MessageBreak
3633       See the minitoc package documentation\MessageBreak
3634       for specific precautions\@gobble}}}%

```

9.78.1.6 Hint about the titletoc package

\@ifpackageloaded We test the presence of the titletoc package [27], and emit a warning, because this
\if@mtc@hints@given@ package is incompatible with minitoc:

```

3635 \@ifpackageloaded{titletoc}%
3636   {\@mtc@hints@given@true
3637     \mtcPackageWarningNoLine[<W0040>]{minitoc(hints)}%
3638     {--- The titletoc package is loaded. \MessageBreak
3639       It is incompatible\MessageBreak with the minitoc package}}}%

```

9.78.1.7 Hint about the placeins package

\@ifpackageloaded We test if the placeins package [9] is loaded and, if yes, we check that the right options
\if@mtc@ss@insection@ are selected:

```

\@ifpackagewith
\if@mtc@hints@given@ 3640 \@ifpackageloaded{placeins}%
3641   {\if@mtc@ss@insection@
3642     \@ifpackagewith{placeins}{section}}}%
3643   {\@mtc@hints@given@true

```

```

3644 \mtcPackageWarningNoLine[<W0031>]{minitoc(hints)}%
3645 {--- The placeins package is loaded
3646 \MessageBreak
3647 without the section option,
3648 \MessageBreak
3649 but minitoc used the insection option
3650 \MessageBreak
3651 which implies it. Try to inverse the
3652 \MessageBreak
3653 loading order and use coherent options.
3654 \MessageBreak
3655 You may have got a message\MessageBreak
3656 ! LaTeX Error: Option clash for package placeins}%
3657 }%
3658 \@ifpackagewith{placeins}{above}%
3659 {\@mtc@hints@given@true
3660 \mtcPackageWarningNoLine[<W0084>]{minitoc(hints)}%
3661 {--- The placeins package is loaded
3662 \MessageBreak
3663 with the above option,
3664 \MessageBreak
3665 but minitoc used the insection option
3666 \MessageBreak
3667 which is incompatible with it.
3668 \MessageBreak
3669 Try to remove the above option
3670 \MessageBreak
3671 and use coherent options}%
3672 }{}%
3673 \@ifpackagewith{placeins}{below}%
3674 {\@mtc@hints@given@true
3675 \mtcPackageWarningNoLine[<W0085>]{minitoc(hints)}%
3676 {--- The placeins package is loaded
3677 \MessageBreak
3678 with the below option,
3679 \MessageBreak
3680 but minitoc used the insection option
3681 \MessageBreak
3682 which is incompatible with it.
3683 \MessageBreak
3684 Try to remove the below option
3685 \MessageBreak
3686 and use coherent options}%
3687 }{}%
3688 \fi
3689 \@ifpackagelater{placeins}{2005/04/18}{}{}%
3690 \@mtc@hints@given@true
3691 \mtcPackageWarningNoLine[<W0032>]{minitoc(hints)}%
3692 {--- The placeins package loaded is
3693 \MessageBreak
3694 too old. It should have a version
3695 \MessageBreak
3696 date of 2005/04/18 at least}%
3697 }%

```

```
3698   }{}%
```

9.78.1.8 Hint about the memoir class

```
\@ifclassloaded We test if the memoir class [213, 214] is loaded:
\if@mtc@hints@given@
```

```
3699 \@ifclassloaded{memoir}%
3700   {\@mtc@hints@given@true
3701     \mtcPackageInfo[<I0044>]{minitoc(hints)}}%
3702     {--- The memoir class is loaded. \MessageBreak
3703       See the minitoc package documentation\MessageBreak
3704       for specific precautions\@gobble}{}%
3705   }{}%
```

9.78.1.9 Hint about the amsart and amsproc classes

```
\@ifclassloaded We test if the amsart or amsproc class is loaded and emit a warning, because these
\if@mtc@hints@given@ classes are incompatible with minitoc:
```

```
3706 \@ifclassloaded{amsart}%
3707   {\@mtc@hints@given@true
3708     \mtcPackageWarningNoLine[<W0026>]{minitoc(hints)}}%
3709     {--- The amsart class is loaded. \MessageBreak
3710       It is incompatible\MessageBreak with the minitoc package}}{}%
3711 \@ifclassloaded{amsproc}%
3712   {\@mtc@hints@given@true
3713     \mtcPackageWarningNoLine[<W0027>]{minitoc(hints)}}%
3714     {--- The amsproc class is loaded. \MessageBreak
3715       It is incompatible\MessageBreak with the minitoc package}}{}%
```

9.78.1.10 Hint about the amsbook class

```
\@ifclassloaded We test if the amsbook class is loaded:
\if@mtc@hints@given@
```

```
3716 \@ifclassloaded{amsbook}%
3717   {\@mtc@hints@given@true
3718     \mtcPackageInfo[<I0041>]{minitoc(hints)}}%
3719     {--- The amsbook class is loaded. \MessageBreak
3720       See the minitoc package documentation\MessageBreak
3721       for specific precautions\@gobble}{}%
3722   }{}%
```

9.78.1.11 Hint about the abstract package

```

\@ifpackageloaded We test the presence of the abstract package [208], then its options:
\@ifpackagewith
\if@mtc@hints@given@ 3723 \@ifpackageloaded{abstract}%
                      3724   {\@mtc@abstract@loaded@true%
                      3725   \@ifpackagewith{abstract}{addtotoc}%
                      3726   {\@mtc@hints@given@true
                      3727   \mtcPackageInfo[<I0040>]{minitoc(hints)}%
                      3728   {The ‘‘abstract’’ package has been\MessageBreak
                      3729   loaded with the ‘‘addtotoc’’ option.\MessageBreak
                      3730   You need to look at the\MessageBreak
                      3731   documentation to adjust.
                      3732   \MessageBreak}
                      3733   }}{}

```

9.78.1.12 Hint about the jura class

```

\@ifclassloaded We test if the jura class is loaded and emit a warning, because this class is incompatible
\if@mtc@hints@given@ with minitoc:

3734 \@ifclassloaded{jura}%
3735   {\@mtc@hints@given@true
3736   \mtcPackageWarningNoLine[<W0029>]{minitoc(hints)}%
3737   {--- The jura class is loaded. \MessageBreak
3738   It is incompatible\MessageBreak with the minitoc package}}{}%

```

9.78.1.13 Hint about the alteration of the sectionning commands

```

\part To check if the sectionning commands \part, \chapter or \section have
\chapter been altered by some package or in the preamble, we compare them (when
\section executing a \AtBeginDocument block) with their saved versions (saved by the
\AtBeginDocument minitoc package when it is loaded) \mtc@hints@part, \mtc@hints@chapter
\mtc@hints@part and \mtc@hints@section. For each sectionning command, we must perform the
\mtc@hints@chapter comparaisn for the command itself, its unstarred branch and its starred branch. But the
\mtc@hints@section hyperref package [179] may interfere, hence the formal precautions in the messages.

```

9.78.1.13.1 Alteration of \part :

```

\if@mtc@hints@w@ We check the alteration of \part, \@part and \@spart:
\@ifundefined
  \part 3739 \@mtc@hints@w@false
\mtc@hints@part 3740 \@ifundefined{part}{}{\ifx\part\mtc@hints@part\relax
  \@part
\mtc@hints@@part
  \@spart
\mtc@hints@@spart
\if@mtc@hints@given@
\if@mtc@hyper@used@

```

```

3741 \else\@mtc@hints@w@true\fi}
3742 \ifundefined{part}{\ifx\@part\mtc@hints@@part\relax
3743 \else\@mtc@hints@w@true\fi}
3744 \ifundefined{part}{\ifx\@spart\mtc@hints@@spart\relax
3745 \else\@mtc@hints@w@true\fi}
3746 \if@mtc@hints@w@\@mtc@hints@given@true%
3747 \mtcPackageWarningNoLine[<W0030>]{minitoc(hints)}%
3748 {--- The \string\part\space command is altered\MessageBreak
3749 after minitoc}
3750 \if@mtc@hyper@used@
3751 \mtcPackageWarningNoLine[<W0023>]{minitoc(hints)}%
3752 {--- It may be the consequence\MessageBreak
3753 of loading the ‘‘hyperref’’ package}
3754 \fi
3755 \fi

```

9.78.1.13.2 Alteration of \chapter :

```

\if@mtc@hints@w@ We check the alteration of \chapter, \@chapter and \@schapter:
\ifundefined
\chapter 3756 \@mtc@hints@w@false
\mtc@hints@chapter 3757 \ifundefined{chapter}{\ifx\chapter\mtc@hints@chapter\relax
\chapter 3758 \else\@mtc@hints@w@true\fi}%
\mtc@hints@@chapter 3759 \ifundefined{chapter}{\ifx\@chapter\mtc@hints@@chapter\relax
\@schapter 3760 \else\@mtc@hints@w@true\fi}%
\mtc@hints@@schapter 3761 \ifundefined{chapter}{\ifx\@schapter\mtc@hints@@schapter\relax
\if@mtc@hints@given@ 3762 \else\@mtc@hints@w@true\fi}%
\if@mtc@hyper@used@ 3763 \if@mtc@hints@w@\@mtc@hints@given@true%
3764 \mtcPackageWarningNoLine[<W0028>]{minitoc(hints)}%
3765 {--- The \string\chapter\space command is altered\MessageBreak
3766 after minitoc}
3767 \if@mtc@hyper@used@
3768 \mtcPackageWarningNoLine[<W0023>]{minitoc(hints)}%
3769 {--- It may be the consequence\MessageBreak
3770 of loading the ‘‘hyperref’’ package}
3771 \fi
3772 \fi

```

9.78.1.13.3 Alteration of \section :

```

\if@mtc@hints@w@ We check the alteration of \section, \@sect and \@ssect:
\ifundefined
\section 3773 \@mtc@hints@w@false
\mtc@hints@section 3774 \ifundefined{chapter}%
\@sect 3775 {\ifundefined{section}{\ifx\section\mtc@hints@section\relax\else
\mtc@hints@sect 3776 \@mtc@hints@w@true\fi}
\@ssect 3777 \ifundefined{section}{\ifx\@sect\mtc@hints@sect\relax\else
\mtc@hints@@ssect 3778 \@mtc@hints@w@true\fi}
\if@mtc@hints@given@
\if@mtc@hyper@used@

```

```

3779 \ifundefined{section}{{\ifx\ssect\mtc@hints@sssect\relax\else
3780 \mtc@hints@w@true\fi}
3781 \if@mtc@hints@w@\mtc@hints@given@true%
3782 \mtcPackageWarningNoLine[<W0039>]{minitoc(hints)}%
3783 {--- The \string\section\space command is altered\MessageBreak
3784 after minitoc}
3785 \if@mtc@hyper@used@
3786 \mtcPackageWarningNoLine[<W0023>]{minitoc(hints)}%
3787 {--- It may be the consequence\MessageBreak
3788 of loading the ‘‘hyperref’’ package}
3789 \fi
3790 \relax\else\fi}}}%

```

9.78.1.14 Hint about the coherence of the calling sequences of some commands

And finally, we prepare the coherence tests about the calling sequences of triplets of associated commands like `\doparttoc`, `\parttoc` and `\[fake]listofcontents`, and similar: to be able to use `\parttoc`, a table of contents file must have been created via `\[fake]listofcontents` and splitted into `parttoc` files via `\doparttoc`.

```

\if@mtc@hints@ Hence we initialize some flags:
\if@mtc@toc@used@
\if@mtc@lof@used@ 3791 \if@mtc@hints@
\if@mtc@lot@used@ 3792 \global\@mtc@toc@used@false
3793 \global\@mtc@lof@used@false
3794 \global\@mtc@lot@used@false

```

```

\mtc@sv@tableofcontents Then we patch the involved commands to set the corresponding flag when they are used.
\tableofcontents First, the commands for the main summaries:
\if@mtc@toc@used@
\mtc@sv@listoffigures 3795 \let\mtc@sv@tableofcontents\tableofcontents
\listoffigures 3796 \def\tableofcontents%
\if@mtc@lof@used@ 3797 {\global\@mtc@toc@used@true\mtc@sv@tableofcontents}
\mtc@sv@listoftables 3798 \let\mtc@sv@listoffigures\listoffigures
\listoftables 3799 \def\listoffigures%
\if@mtc@lot@used@ 3800 {\global\@mtc@lof@used@true\mtc@sv@listoffigures}
3801 \let\mtc@sv@listoftables\listoftables
3802 \def\listoftables%
3803 {\global\@mtc@lot@used@true\mtc@sv@listoftables}

```

```

\mtc@sv@faketableofcontents Then, their ‘‘fake’’ siblings:
\faketableofcontents
\if@mtc@toc@used@ 3804 \let\mtc@sv@faketableofcontents\faketableofcontents
\mtc@sv@fakelistoffigures 3805 \def\faketableofcontents%
\fake listoffigures 3806 {\global\@mtc@toc@used@true\mtc@sv@faketableofcontents}
\if@mtc@lof@used@ 3807 \let\mtc@sv@fakelistoffigures\fakelistoffigures
\mtc@sv@fakelistoftables
\fake listoftables
\if@mtc@lot@used@

```



```

3808 \def\fakehistoffigures%
3809   {\global\@mtc@lof@used@true\mtc@sv@fakehistoffigures}
3810 \let\mtc@sv@fakehistoftables\fakehistoftables
3811 \def\fakehistoftables%
3812   {\global\@mtc@lot@used@true\mtc@sv@fakehistoftables}
3813 \fi

```

\mtc@hints@begindoc And the \mtc@hints@begindoc definition is finished (it begins in section 9.78.1 on page 369):

```

3814 }

```

9.78.2 Final part: \mtc@hints@enddoc

\mtc@hints@enddoc The final part of the hints option is executed via \AtEndDocument. Its code is in the \AtEndDocument \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:

```

3815 \def\mtc@hints@enddoc{%

```

9.78.2.1 Hint about \sect-lof|lot and the insection option

```

\if@mtc@sect@floats@ We look if some section-level lists of figures or tables have been requested.
\if@dosectlof@used@
\if@dosectlot@used@ 3816 \if@dosectlof@used@\@mtc@sect@floats@true\fi
\if@sectlof@used@ 3817 \if@dosectlot@used@\@mtc@sect@floats@true\fi
\if@sectlot@used@ 3818 \if@sectlof@used@\@mtc@sect@floats@true\fi
\if@mtc@section@def@ 3819 \if@sectlot@used@\@mtc@sect@floats@true\fi
3820 \if@mtc@section@def@

```

```

\if@mtc@placeinsLoaded@ If yes, we verify that the placeins package [9] has been loaded with the correct options
\if@mtc@sect@floats@ or that the insection option of the minitoc package has been invoked. If not, a warning
\if@mtc@hints@w@ is given.
\if@mtc@hints@given@

```

```

3821 \if@mtc@placeinsLoaded@ \else
3822   \if@mtc@sect@floats@%
3823     \mtcPackageWarningNoLine[<W0056>]{minitoc(hints)}%
3824     {You are using \string\dosectlof\space and/or \MessageBreak
3825       \string\dosectlot, \string\sectlof\space and/or \string\sectlot,
3826       \MessageBreak
3827       hence the ‘insection’ package\MessageBreak
3828       option is recommended}%

```

```

3829 \@mtc@hints@w@true \@mtc@hints@given@true
3830 \fi
3831 \fi

```

9.78.2.2 Final part of the coherence 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@ 3832 \if@mtc@part@def@
\if@mtc@hints@given@ 3833 \if@parttoc@used@
\if@partlof@used@ 3834 \if@doparttoc@used@ \else
\if@dopartlof@used@ 3835 \mtcPackageWarningNoLine[<W0062>]{minitoc(hints)}%
\if@partlot@used@ 3836 {You have used \string\parttoc,\MessageBreak
\if@dopartlot@used@ 3837 but not \string\doparttoc}
3838 \@mtc@hints@given@true
3839 \fi
3840 \fi
3841 \if@partlof@used@
3842 \if@dopartlof@used@ \else
3843 \mtcPackageWarningNoLine[<W0060>]{minitoc(hints)}%
3844 {You have used \string\partlof,\MessageBreak
3845 but not \string\dopartlof}
3846 \@mtc@hints@given@true
3847 \fi
3848 \fi
3849 \if@partlot@used@
3850 \if@dopartlot@used@ \else
3851 \mtcPackageWarningNoLine[<W0061>]{minitoc(hints)}%
3852 {You have used \string\partlot,\MessageBreak
3853 but not \string\dopartlot}
3854 \@mtc@hints@given@true
3855 \fi
3856 \fi
3857 \fi

\if@mtc@chapter@def@ For the chapter level commands:
\if@minitoc@used@
\if@dominitoc@used@ 3858 \if@mtc@chapter@def@
\if@mtc@hints@given@ 3859 \if@minitoc@used@
\if@minilof@used@ 3860 \if@dominitoc@used@ \else
\if@dominilof@used@ 3861 \mtcPackageWarningNoLine[<W0059>]{minitoc(hints)}%
\if@minilot@used@ 3862 {You have used \string\minitoc,\MessageBreak
\if@dominilot@used@ 3863 but not \string\dominitoc}
3864 \@mtc@hints@given@true
3865 \fi

```

```

3866 \fi
3867 \if@minilof@used@
3868   \if@dominilof@used@\else
3869     \mtcPackageWarningNoLine[<W0057>]{minitoc(hints)}%
3870     {You have used \string\minilof,\MessageBreak
3871     but not \string\dominilof}
3872   \@mtc@hints@given@true
3873 \fi
3874 \fi
3875 \if@minilot@used@
3876   \if@dominilot@used@\else
3877     \mtcPackageWarningNoLine[<W0058>]{minitoc(hints)}%
3878     {You have used \string\minilot,\MessageBreak
3879     but not \string\dominilot}
3880   \@mtc@hints@given@true
3881 \fi
3882 \fi
3883 \fi

```

\if@mtc@section@def@ For the section level commands:

```

\if@secttoc@used@
\if@dosecttoc@used@ 3884 \if@mtc@section@def@
\if@mtc@hints@given@ 3885 \if@secttoc@used@
  \if@sectlof@used@ 3886   \if@dosecttoc@used@\else
\if@dosectlof@used@ 3887     \mtcPackageWarningNoLine[<W0065>]{minitoc(hints)}%
  \if@sectlot@used@ 3888     {You have used \string\secttoc,\MessageBreak
\if@dosectlot@used@ 3889     but not \string\dosecttoc}
3890   \@mtc@hints@given@true
3891 \fi
3892 \fi
3893 \if@sectlof@used@
3894   \if@dosectlof@used@\else
3895     \mtcPackageWarningNoLine[<W0063>]{minitoc(hints)}%
3896     {You have used \string\sectlof,\MessageBreak
3897     but not \string\dosectlof}
3898   \@mtc@hints@given@true
3899 \fi
3900 \fi
3901 \if@sectlot@used@
3902   \if@dosectlot@used@\else
3903     \mtcPackageWarningNoLine[<W0064>]{minitoc(hints)}%
3904     {You have used \string\sectlot,\MessageBreak
3905     but not \string\dosectlot}
3906   \fi
3907 \fi
3908 \fi

```

9.78.2.3 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 its a vain call. We do the same for each analog command.

```

\if@mtc@part@def@ Part level commands:
\if@doparttoc@used@
  \if@parttoc@used@ 3909 \if@mtc@part@def@
\if@mtc@hints@given@ 3910 \if@doparttoc@used@
\if@dopartlof@used@ 3911 \if@parttoc@used@\else
  \if@partlof@used@ 3912 \mtcPackageWarningNoLine[<W0075>]{minitoc(hints)}%
\if@dopartlot@used@ 3913 {You have used \string\doparttoc,\MessageBreak
  \if@partlot@used@ 3914 but not \string\parttoc}
  3915 \@mtc@hints@given@true
  3916 \fi
  3917 \fi
  3918 \if@dopartlof@used@
  3919 \if@partlof@used@\else
  3920 \mtcPackageWarningNoLine[<W0076>]{minitoc(hints)}%
  3921 {You have used \string\dopartlof,\MessageBreak
  3922 but not \string\partlof}
  3923 \@mtc@hints@given@true
  3924 \fi
  3925 \fi
  3926 \if@dopartlot@used@
  3927 \if@partlot@used@\else
  3928 \mtcPackageWarningNoLine[<W0077>]{minitoc(hints)}%
  3929 {You have used \string\dopartlot,\MessageBreak
  3930 but not \string\partlot}
  3931 \@mtc@hints@given@true
  3932 \fi
  3933 \fi
  3934 \fi

\if@mtc@chapter@def@ Chapter level commands:
\if@dominitoc@used@
  \if@minitoc@used@ 3935 \if@mtc@chapter@def@
\if@mtc@hints@given@ 3936 \if@dominitoc@used@
\if@dominilof@used@ 3937 \if@minitoc@used@\else
  \if@minilof@used@ 3938 \mtcPackageWarningNoLine[<W0078>]{minitoc(hints)}%
\if@dominilot@used@ 3939 {You have used \string\dominitoc,\MessageBreak
  \if@minilot@used@ 3940 but not \string\minitoc}
  3941 \@mtc@hints@given@true
  3942 \fi
  3943 \fi
  3944 \if@dominilof@used@
  3945 \if@minilof@used@\else
  3946 \mtcPackageWarningNoLine[<W0079>]{minitoc(hints)}%
  3947 {You have used \string\dominilof,\MessageBreak
  3948 but not \string\minilof}

```

```

3949 \@mtc@hints@given@true
3950 \fi
3951 \fi
3952 \if@dominilot@used@
3953 \if@minilot@used@else
3954 \mtcPackageWarningNoLine[<W0080>]{minitoc(hints)}%
3955 {You have used \string\dominilot,\MessageBreak
3956 but not \string\minilot}
3957 \@mtc@hints@given@true
3958 \fi
3959 \fi
3960 \fi

```

```

\if@mtc@section@def@ Section level commands:
\if@dosecttoc@used@
\if@secttoc@used@ 3961 \if@mtc@section@def@
\if@mtc@hints@given@ 3962 \if@dosecttoc@used@
\if@dosectlof@used@ 3963 \if@secttoc@used@else
\if@sectlof@used@ 3964 \mtcPackageWarningNoLine[<W0081>]{minitoc(hints)}%
\if@dosectlot@used@ 3965 {You have used \string\dosecttoc,\MessageBreak
\if@sectlot@used@ 3966 but not \string\secttoc}
3967 \@mtc@hints@given@true
3968 \fi
3969 \fi
3970 \if@dosectlof@used@
3971 \if@sectlof@used@else
3972 \mtcPackageWarningNoLine[<W0082>]{minitoc(hints)}%
3973 {You have used \string\dosectlof,\MessageBreak
3974 but not \string\sectlof}
3975 \@mtc@hints@given@true
3976 \fi
3977 \fi
3978 \if@dosectlot@used@
3979 \if@sectlot@used@else
3980 \mtcPackageWarningNoLine[<W0083>]{minitoc(hints)}%
3981 {You have used \string\dosectlot,\MessageBreak
3982 but not \string\sectlot}
3983 \fi
3984 \fi
3985 \fi
3986 \fi

```

9.78.2.4 Check if the main tables have been prepared (second part)

Another coherence test verifies that if \parttoc has been called, then the macro \tableofcontents or \faketableofcontents has also been called (to create the necessary contents file); and similar tests are made for the other mini-table commands.

```

\if@mtc@part@def@ Part level commands:
\if@mtc@hints@given@
\if@parttoc@used@ 3987 \if@mtc@part@def@
\ifmtc@toc@used@ 3988 \if@parttoc@used@
\if@partlof@used@ 3989 \if@mtc@toc@used@\else
\ifmtc@lof@used@ 3990 \mtcPackageWarningNoLine[<W0071>]{minitoc(hints)}%
\if@partlot@used@ 3991 {You have used \string\parttoc\space but not\MessageBreak
\ifmtc@lot@used@ 3992 \string\tableofcontents\MessageBreak nor
3993 \string\fake\tableofcontents}
3994 \@mtc@hints@given@true
3995 \fi
3996 \fi
3997 \if@partlof@used@
3998 \if@mtc@lof@used@\else
3999 \mtcPackageWarningNoLine[<W0069>]{minitoc(hints)}%
4000 {You have used \string\partlof\space but not\MessageBreak
4001 \string\listoffigures\MessageBreak nor
4002 \string\fake\listoffigures}
4003 \@mtc@hints@given@true
4004 \fi
4005 \fi
4006 \if@partlot@used@
4007 \if@mtc@lot@used@\else
4008 \mtcPackageWarningNoLine[<W0070>]{minitoc(hints)}%
4009 {You have used \string\partlot\space but not\MessageBreak
4010 \string\listoftables\MessageBreak nor
4011 \string\fake\listoftables}
4012 \@mtc@hints@given@true
4013 \fi
4014 \fi
4015 \fi

```

```

\if@mtc@chapter@def@ Chapter level commands:
\if@mtc@hints@given@
\if@minitoc@used@ 4016 \if@mtc@chapter@def@
\ifmtc@toc@used@ 4017 \if@minitoc@used@
\if@minilof@used@ 4018 \if@mtc@toc@used@\else
\ifmtc@lof@used@ 4019 \mtcPackageWarningNoLine[<W0068>]{minitoc(hints)}%
\if@minilot@used@ 4020 {You have used \string\minitoc\space but not\MessageBreak
\ifmtc@lot@used@ 4021 \string\tableofcontents\MessageBreak nor
4022 \string\fake\tableofcontents}
4023 \@mtc@hints@given@true
4024 \fi
4025 \fi
4026 \if@minilof@used@
4027 \if@mtc@lof@used@\else
4028 \mtcPackageWarningNoLine[<W0066>]{minitoc(hints)}%
4029 {You have used \string\minilof\space but not\MessageBreak
4030 \string\listoffigures\MessageBreak nor
4031 \string\fake\listoffigures}
4032 \@mtc@hints@given@true
4033 \fi

```

```

4034 \fi
4035 \if@minilot@used@
4036   \if@mtc@lot@used@\else
4037     \mtcPackageWarningNoLine[<W0067>]{minitoc(hints)}%
4038     {You have used \string\minilot\space but not\MessageBreak
4039       \string\listoftables\MessageBreak nor
4040       \string\fake\listoftables}
4041   \@mtc@hints@given@true
4042 \fi
4043 \fi
4044 \fi

```

\if@mtc@section@def@ Section level commands:

\if@mtc@hints@given@

```

\if@secttoc@used@ 4045 \if@mtc@section@def@
\ifmtc@toc@used@ 4046 \if@secttoc@used@
\if@sectlof@used@ 4047 \if@mtc@toc@used@\else
\ifmtc@lof@used@ 4048 \mtcPackageWarningNoLine[<W0074>]{minitoc(hints)}%
\if@sectlot@used@ 4049 {You have used \string\secttoc\space but not\MessageBreak
\ifmtc@lot@used@ 4050 \string\tableofcontents\MessageBreak nor
4051 \string\fake\tableofcontents}
4052 \@mtc@hints@given@true
4053 \fi
4054 \fi
4055 \if@sectlof@used@
4056   \if@mtc@lof@used@\else
4057     \mtcPackageWarningNoLine[<W0072>]{minitoc(hints)}%
4058     {You have used \string\sectlof\space but not\MessageBreak
4059       \string\listoffigures\MessageBreak nor
4060       \string\fake\listoffigures}
4061   \@mtc@hints@given@true
4062 \fi
4063 \fi
4064 \if@sectlot@used@
4065   \if@mtc@lot@used@\else
4066     \mtcPackageWarningNoLine[<W0073>]{minitoc(hints)}%
4067     {You have used \string\sectlot\space but not\MessageBreak
4068       \string\listoftables\MessageBreak nor
4069       \string\fake\listoftables}
4070   \@mtc@hints@given@true
4071 \fi
4072 \fi
4073 \fi

```

9.78.2.5 Check the number of mini-tables, in case of short extensions

```

\mtc@hints@checklongext If short extensions are used, you can use only 99 mini-tables of each kind. If more are
\if@longextensions@ created, the auxiliary files can be overwritten: the hundredth minitoc file \jobname.U100
\if@mtc@part@def@ has its name truncated to \jobname.U10, which is already the tenth minitoc file. Thus,
\value
\if@mtc@hints@given@true
\if@mtc@chapter@def@
\if@mtc@section@def@

```

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 ⁹, either redesign the document.

```

4074 \def\mtc@hints@checklongext{%
4075 \if@longextensions@
4076 \else
4077   \if@mtc@part@def@
4078     \ifnum 99 < \value{ptc}\relax
4079       \@mtc@hints@given@true
4080       \mtcPackageWarningNoLine[<W0054>]{minitoc(hints)}%
4081       {You have used short extensions\MessageBreak
4082        and more than 99 parts (\arabic{ptc})}
4083     \fi
4084   \fi
4085   \if@mtc@chapter@def@
4086     \ifnum 99 < \value{mtc}\relax
4087       \@mtc@hints@given@true
4088       \mtcPackageWarningNoLine[<W0053>]{minitoc(hints)}%
4089       {You have used short extensions\MessageBreak
4090        and more than 99 chapters (\arabic{mtc})}
4091     \fi
4092   \else
4093     \if@mtc@section@def@
4094       \ifnum 99 < \value{stc}\relax
4095         \@mtc@hints@given@true
4096         \mtcPackageWarningNoLine[<W0055>]{minitoc(hints)}%
4097         {You have used short extensions\MessageBreak
4098          and more than 99 sections (\arabic{stc})}
4099       \fi
4100     \fi
4101   \fi
4102 \fi}
4103 \mtc@hints@checklongext

```

9.78.2.6 Final part of the hint about the sectsty package

\if@mtc@sectstyLoaded@ We test if sectsty has been loaded before (correct) or after (incorrect) minitoc. See section 9.9.1 on page 242.

\if@mtc@hints@given@

```

4104 \if@mtc@sectstyLoaded@\else
4105   \if@mtc@sectstyLoaded@a@
4106     \mtcPackageWarningNoLine[<W0037>]{minitoc(hints)}%
4107     {The sectsty package should be\MessageBreak
4108      loaded BEFORE the minitoc package}
4109   \@mtc@hints@given@true
4110 \fi
4111 \fi

```

⁹ On the long term, a good investment.

9.78.2.7 Final part of the hint about the varsects package

`\if@mtc@varsectsLoaded@` We test if varsects has been loaded before (correct) or after (incorrect) minitoc. See
`\if@mtc@varsectsLoaded@a@` section 9.9.2 on page 242.
`\if@mtc@hints@given@`

```

4112 \if@mtc@varsectsLoaded@\else
4113   \if@mtc@varsectsLoaded@a@
4114     \mtcPackageWarningNoLine[<W0038>]{minitoc(hints)}%
4115     {The varsects package should be\MessageBreak
4116       loaded BEFORE the minitoc package}
4117     \@mtc@hints@given@true
4118   \fi
4119 \fi

```

9.78.2.8 Final part of the hint about the fncychap package

`\if@mtc@fncychapLoaded@` We test if fncychap has been loaded before (correct) or after (incorrect) minitoc. See
`\if@mtc@fncychapLoaded@a@` section 9.9.3 on page 242.
`\if@mtc@hints@given@`

```

4120 \if@mtc@fncychapLoaded@\else
4121   \if@mtc@fncychapLoaded@a@
4122     \mtcPackageWarningNoLine[<W0086>]{minitoc(hints)}%
4123     {The fncychap package should be\MessageBreak
4124       loaded BEFORE the minitoc package}
4125     \@mtc@hints@given@true
4126   \fi
4127 \fi

```

9.78.2.9 Final part of the hint about the quotchap package

`\if@mtc@quotchapLoaded@` We test if quotchap has been loaded before (correct) or after (incorrect) minitoc. See
`\if@mtc@quotchapLoaded@a@` section 9.9.4 on page 242.
`\if@mtc@hints@given@`

```

4128 \if@mtc@quotchapLoaded@\else
4129   \if@mtc@quotchapLoaded@a@
4130     \mtcPackageWarningNoLine[<W0087>]{minitoc(hints)}%
4131     {The quotchap package should be\MessageBreak
4132       loaded BEFORE the minitoc package}
4133     \@mtc@hints@given@true
4134   \fi
4135 \fi

```

9.78.2.10 Final part of the hint about the romannum package

`\if@mtc@romannumLoaded@` We test if romannum has been loaded before (correct) or after (incorrect) minitoc. See
`\if@mtc@romannumLoaded@a@` section 9.9.5 on page 243.
`\if@mtc@hints@given@`

```

4136 \if@mtc@romannumLoaded@\else
4137   \if@mtc@romannumLoaded@a@
4138     \mtcPackageWarningNoLine[<W0088>]{minitoc(hints)}%
4139     {The romannum package should be\MessageBreak
4140      loaded BEFORE the minitoc package}
4141     \@mtc@hints@given@true
4142   \fi
4143 \fi

```

9.78.2.11 Final part of the hint about the sfheaders package

`\if@mtc@sfheadersLoaded@` We test if sfheaders has been loaded before (correct) or after (incorrect) minitoc. See
`\if@mtc@sfheadersLoaded@a@` section 9.9.6 on page 243.
`\if@mtc@hints@given@`

```

4144 \if@mtc@sfheadersLoaded@\else
4145   \if@mtc@sfheadersLoaded@a@
4146     \mtcPackageWarningNoLine[<W0089>]{minitoc(hints)}%
4147     {The sfheaders package should be\MessageBreak
4148      loaded BEFORE the minitoc package}
4149     \@mtc@hints@given@true
4150   \fi
4151 \fi

```

9.78.2.12 Final part of the hint about the alnumsec package

`\if@mtc@alnumsecLoaded@` We test if alnumsec has been loaded before (correct) or after (incorrect) minitoc. See
`\if@mtc@alnumsecLoaded@a@` section 9.9.7 on page 243.
`\if@mtc@hints@given@`

```

4152 \if@mtc@alnumsecLoaded@\else
4153   \if@mtc@alnumsecLoaded@a@
4154     \mtcPackageWarningNoLine[<W0090>]{minitoc(hints)}%
4155     {The alnumsec package should be\MessageBreak
4156      loaded BEFORE the minitoc package}
4157     \@mtc@hints@given@true
4158   \fi
4159 \fi

```

9.78.2.13 Final part of the hint about the captcont package

`\if@mtc@captcontLoaded@` We test if captcont has been loaded before (correct) or after (incorrect) minitoc. See
`\if@mtc@captcontLoaded@a@` section 9.9.8 on page 243.
`\if@mtc@hints@given@`

```

4160 \if@mtc@captcontLoaded@\else
4161   \if@mtc@captcontLoaded@a@
4162     \mtcPackageWarningNoLine[<W0091>]{minitoc(hints)}%
4163     {The captcont package should be\MessageBreak
4164       loaded BEFORE the minitoc package}
4165     \@mtc@hints@given@true
4166   \fi
4167 \fi

```

9.78.2.14 Final part of the hint about the caption package

`\if@mtc@captionLoaded@` We test if caption has been loaded before (correct) or after (incorrect) minitoc. See
`\if@mtc@captionLoaded@a@` section 9.9.9 on page 244.
`\if@mtc@hints@given@`

```

4168 \if@mtc@captionLoaded@\else
4169   \if@mtc@captionLoaded@a@
4170     \mtcPackageWarningNoLine[<W0033>]{minitoc(hints)}%
4171     {The caption package should be\MessageBreak
4172       loaded BEFORE the minitoc package}
4173     \@mtc@hints@given@true
4174   \fi
4175 \fi

```

9.78.2.15 Final part of the hint about the caption2 package

`\if@mtc@captionIILoaded@` We test if caption2 has been loaded before (correct) or after (incorrect) minitoc. See
`\if@mtc@captionIILoaded@a@` section 9.9.10 on page 244.
`\if@mtc@hints@given@`

```

4176 \if@mtc@captionIILoaded@\else
4177   \if@mtc@captionIILoaded@a@
4178     \mtcPackageWarningNoLine[<W0034>]{minitoc(hints)}%
4179     {The caption2 package should be\MessageBreak
4180       loaded BEFORE the minitoc package}
4181     \@mtc@hints@given@true
4182   \fi
4183 \fi

```

9.78.2.16 Final part of the hint about the ccaption package

`\if@mtc@ccaptionLoaded@` We test if ccaption has been loaded before (correct) or after (incorrect) minitoc. See
`\if@mtc@ccaptionLoaded@a@` section 9.9.11 on page 244.
`\if@mtc@hints@given@`

```

4184 \if@mtc@ccaptionLoaded@\else
4185   \if@mtc@ccaptionLoaded@a@
4186     \mtcPackageWarningNoLine[<W0035>]{minitoc(hints)}%
4187     {The ccaption package should be\MessageBreak
4188       loaded BEFORE the minitoc package}
4189     \@mtc@hints@given@true
4190   \fi
4191 \fi

```

9.78.2.17 Final part of the hint about the mcaption package

`\if@mtc@mcaptionLoaded@` We test if mcaption has been loaded before (correct) or after (incorrect) minitoc. See
`\if@mtc@mcaptionLoaded@a@` section 9.9.12 on page 244.
`\if@mtc@hints@given@`

```

4192 \if@mtc@mcaptionLoaded@\else
4193   \if@mtc@mcaptionLoaded@a@
4194     \mtcPackageWarningNoLine[<W0036>]{minitoc(hints)}%
4195     {The mcaption package should be\MessageBreak
4196       loaded BEFORE the minitoc package}
4197     \@mtc@hints@given@true
4198   \fi
4199 \fi

```

9.78.2.18 Check if empty mini-tables have been detected

We test for each kind of mini-tables.

`\if@mtc@empty@parttoc@` For parttocs:

```

4200 \if@mtc@empty@parttoc@
4201   \mtcPackageWarningNoLine[<W0046>]{minitoc(hints)}%
4202   {You have attempted to insert\MessageBreak empty parttocs}
4203   \@mtc@hints@given@true
4204 \fi

```

`\if@mtc@empty@partlof@` For partlofs:

```

4205 \if@mtc@empty@partlof@

```

```

4206 \mtcPackageWarningNoLine[<W0044>]{minitoc(hints)}%
4207 {You have attempted to insert\MessageBreak empty partlofs}
4208 \@mtc@hints@given@true
4209 \fi

```

\if@mtc@empty@partlot@ For partlots:

```

4210 \if@mtc@empty@partlot@
4211 \mtcPackageWarningNoLine[<W0045>]{minitoc(hints)}%
4212 {You have attempted to insert\MessageBreak empty partlots}
4213 \@mtc@hints@given@true
4214 \fi

```

\if@mtc@empty@minitoc@ For minitocs:

```

4215 \if@mtc@empty@minitoc@
4216 \mtcPackageWarningNoLine[<W0043>]{minitoc(hints)}%
4217 {You have attempted to insert\MessageBreak empty minitocs}
4218 \@mtc@hints@given@true
4219 \fi

```

\if@mtc@empty@minilofs@ For minilofs:

```

4220 \if@mtc@empty@minilofs@
4221 \mtcPackageWarningNoLine[<W0041>]{minitoc(hints)}%
4222 {You have attempted to insert\MessageBreak empty minilofs}
4223 \@mtc@hints@given@true
4224 \fi

```

\if@mtc@empty@minilots@ For minilots:

```

4225 \if@mtc@empty@minilots@
4226 \mtcPackageWarningNoLine[<W0042>]{minitoc(hints)}%
4227 {You have attempted to insert\MessageBreak empty minilots}
4228 \@mtc@hints@given@true
4229 \fi

```

\if@mtc@empty@secttoc@ For secttocs:

```

4230 \if@mtc@empty@secttoc@
4231 \mtcPackageWarningNoLine[<W0049>]{minitoc(hints)}%
4232 {You have attempted to insert\MessageBreak empty secttocs}
4233 \@mtc@hints@given@true
4234 \fi

```

`\if@mtc@empty@sectlof@` For sectlofs:

```
4235 \if@mtc@empty@sectlof@
4236   \mtcPackageWarningNoLine[<W0047>]{minitoc(hints)}%
4237   {You have attempted to insert\MessageBreak empty sectlofs}
4238   \@mtc@hints@given@true
4239 \fi
```

`\if@mtc@empty@sectlot@` For sectlots:

```
4240 \if@mtc@empty@sectlot@
4241   \mtcPackageWarningNoLine[<W0042>]{minitoc(hints)}%
4242   {You have attempted to insert\MessageBreak empty sectlots}
4243   \@mtc@hints@given@true
4244 \fi
```

9.78.2.19 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 .log file.

`\if@firstpartis@used@` Obsolete macro `\firstpartis`:

```
4245 \if@firstpartis@used@
4246   \mtcPackageWarningNoLine[<W0051>]{minitoc(hints)}%
4247   {You have invoked an obsolete\MessageBreak command: \string\firstpartis}
4248   \@mtc@hints@given@true
4249 \fi
```

`\if@firstchapteris@used@` Obsolete macro `\firstchapteris`:

```
4250 \if@firstchapteris@used@
4251   \mtcPackageWarningNoLine[<W0050>]{minitoc(hints)}%
4252   {You have invoked an obsolete\MessageBreak command: \string\firstchapteris}
4253   \@mtc@hints@given@true
4254 \fi
```

`\if@firstsectionis@used@` Obsolete macro `\firstsectionis`:

```
4255 \if@firstsectionis@used@
4256   \mtcPackageWarningNoLine[<W0052>]{minitoc(hints)}%
4257   {You have invoked an obsolete\MessageBreak command: \string\firstsectionis}
4258   \@mtc@hints@given@true
4259 \fi
```

9.78.2.20 Check if some hints have been written

`\if@mtc@hints@given@` 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 `.log` file if no problems have been detected. And we terminate the `\mtc@hints@enddoc` macro by a closing brace.

```

4260 \if@mtc@hints@given@
4261   \mtcPackageWarningNoLine[<W0024>]{minitoc(hints)}%
4262     {Some hints have been written\MessageBreak
4263       in the \jobname.log file}
4264 \else
4265   \mtcPackageInfo[<I0019>]{minitoc(hints)}%
4266     {No hints have been written\MessageBreak
4267       in the \jobname.log file.\@gobble}
4268 \fi
4269 }
```

9.79 Processing of options

`\ExecuteOptionsenglish` First, we apply the default language option, `english`:

```
4270 \ExecuteOptions{english}
```

`\ProcessOptions*` Then, we execute all requested options: for most options, it is just setting a flag, or loading a file for the language options.

```
4271 \ProcessOptions*
```

We now examine the flags for some options and execute the necessary actions.

9.79.1 Processing the `insection` option

`\if@mtc@ss@insection@` For the `insection` option, we load the `placeins` package [9] with its options `verbose` and `section`, after the `flafter` package (described in [138] and [159, page 286]); the correct loading is verified:
`\RequirePackage`
`\@ifpackageloaded`
`\if@mtc@placeinsLoaded@`

```

4272 \if@mtc@ss@insection@
4273   \RequirePackage{flafter}[2000/07/23]%
4274   \RequirePackage[section,verbose]{placeins}[2005/04/18]%
4275   \@ifpackageloaded{placeins}%
4276     {\@mtc@placeinsLoaded@true}{\@mtc@placeinsLoaded@false}%
4277 \fi
```

9.79.2 Processing the notoccite option

`\if@mtc@notoccite@` For the notoccite option, we just load the notoccite package [8]:
`\RequirePackage`

```
4278 \if@mtc@notoccite@
4279   \RequirePackage{notoccite}%
4280 \fi
```

9.79.3 Processing the listfiles option

```
\mtc@maf We define the \mtc@maf macro which closes \tf@mtc and reopens it to write into
\tf@mtc <document>.maf. It calls \mtc@maf@long or \mtc@maf@short (long or short exten-
\if@longextensions@ sions), then closes \tf@mtc. \mtc@maf@long or \mtc@maf@short writes the names
\mtc@maf@long of the existing auxiliary files using decrementing loops on the associated counters, and
\mtc@maf@short includes <document>.mtc in the list (but not <document>.maf10). We must also check
\mtc@addtomaf the existence of \jobname.mtc1 if long extensions are used.
\IfFileExists
\jobname 4281 \def\mtc@maf{%
4282   \mtcPackageInfo[<I0009>]{minitoc}%
4283   {Listing minitoc auxiliary files\MessageBreak
4284    Creating the \jobname.maf file\@gobble}
4285   \immediate\closeout\tf@mtc
4286   \immediate\openout\tf@mtc \jobname.maf
4287   \if@longextensions@\mtc@maf@long\else\mtc@maf@short\fi
4288   \immediate\closeout\tf@mtc}
4289 \def\mtc@addtomaf#1{%
4290   \IfFileExists{#1}{\immediate\write\tf@mtc{#1}}{}
4291 \def\mtc@maf@long{%
4292   \mtc@addtomaf{\jobname.mtc}
4293   \@ifundefined{c@ptc}{}{\loop\ifnum\c@ptc>\z@\relax
4294     \mtc@addtomaf{\jobname.ptc\arabic{ptc}}
4295     \mtc@addtomaf{\jobname.plf\arabic{ptc}}
4296     \mtc@addtomaf{\jobname.plt\arabic{ptc}}
4297     \advance\c@ptc\m@ne\repeat}
4298   \@ifundefined{c@mtc}{\mtc@addtomaf{\jobname.mtc1}}{%
4299     \loop\ifnum\c@mtc>\z@\relax
4300       \mtc@addtomaf{\jobname.mtc\arabic{mtc}}
4301       \mtc@addtomaf{\jobname.mlf\arabic{mtc}}
4302       \mtc@addtomaf{\jobname.mlt\arabic{mtc}}
4303       \advance\c@mtc\m@ne\repeat}
4304   \@ifundefined{c@stc}{}{\loop\ifnum\c@stc>\z@\relax
4305     \mtc@addtomaf{\jobname.stc\arabic{stc}}
4306     \mtc@addtomaf{\jobname.slf\arabic{stc}}
4307     \mtc@addtomaf{\jobname.slt\arabic{stc}}
4308     \advance\c@stc\m@ne\repeat}
4309 \def\mtc@maf@short{%
4310   \mtc@addtomaf{\jobname.mtc}
```

¹⁰Some users could made a cleanup using this file as a list of files to delete, so it should not be in the list.


```

4311 \@ifundefined{c@ptc}{}{\loop\ifnum\c@ptc>\z@\relax
4312     \mtc@addtomaf{\jobname.P\arabic{ptc}}
4313     \mtc@addtomaf{\jobname.G\arabic{ptc}}
4314     \mtc@addtomaf{\jobname.U\arabic{ptc}}
4315     \advance\c@ptc@m@ne\repeat}
4316 \@ifundefined{c@mtc}{}{\loop\ifnum\c@mtc>\z@\relax
4317     \mtc@addtomaf{\jobname.M\arabic{mtc}}
4318     \mtc@addtomaf{\jobname.F\arabic{mtc}}
4319     \mtc@addtomaf{\jobname.T\arabic{mtc}}
4320     \advance\c@mtc@m@ne\repeat}
4321 \@ifundefined{c@stc}{}{\loop\ifnum\c@stc>\z@\relax
4322     \mtc@addtomaf{\jobname.S\arabic{stc}}
4323     \mtc@addtomaf{\jobname.H\arabic{stc}}
4324     \mtc@addtomaf{\jobname.V\arabic{stc}}
4325     \advance\c@stc@m@ne\repeat}}

```

`\if@mtc@listfiles@` If this option is active, we call `\mtc@maf` in a `\AtEndDocument` block.

```

\AtEndDocument
\mtc@maf 4326 \if@mtc@listfiles@{\AtEndDocument{\mtc@maf}}\else\fi

```

9.79.4 Processing the hints option

`\if@mtc@hints@` For the hints option, we set its first part in a `\AtBeginDocument` block and its third (last) part in a `\AtEndDocument` block:

```

\AtBeginDocument
\mtc@hints@begindoc
\AtEndDocument 4327 \if@mtc@hints@
\mtc@hints@enddoc 4328 \AtBeginDocument{\mtc@hints@begindoc}%
4329 \AtEndDocument{\mtc@hints@enddoc}%
4330 \fi

```

9.79.5 Saving the sectionning commands

And, at least, we save the definitions of sectionning commands (and of their unstarred and starred branches), for comparaisons (this is a part of the hints option executed in the preamble):

`\@ifundefined` For the `\part` command:

```

\mtc@hints@part
\part 4331 \@ifundefined{part}{}{\let\mtc@hints@part\part
\mtc@hints@@part 4332 \let\mtc@hints@@part\@part
\@part 4333 \let\mtc@hints@@spart\@spart}
\mtc@hints@@spart
\@spart

```

```

\@ifundefined For the \chapter command:
\mtc@hints@chapter
  \chapter 4334 \@ifundefined{chapter}{}{\let\mtc@hints@chapter\chapter
\mtc@hints@@chapter 4335 \let\mtc@hints@@chapter\@chapter
  \@chapter 4336 \let\mtc@hints@@schapter\@schapter}
\mtc@hints@schapter
  \@schapter

```

```

\@ifundefined For the \section command:
\mtc@hints@section
  \section 4337 \@ifundefined{section}{}{\let\mtc@hints@section\section
\mtc@hints@@sect 4338 \let\mtc@hints@@sect\@sect
  \@sect 4339 \let\mtc@hints@@ssect\@ssect}
\mtc@hints@ssect
  \@ssect

```

And the package is terminated.

```
4340 </minitoc>
```

9.80 The `fminitoc.dtx` file

`\jobname` This short file is necessary to create the french documentation. Its rôle is to set `\jobname`
`\input` to `fminitoc` in place of `minitoc`. As `minitoc.ins` generates the files `minitoc.lan` and `fminitoc.lan` which sets a language number `\LANG`, and `minitoc.dtx` reads `\jobname.lan`, the documentation can be in several languages (english and french) in `minitoc.dtx`, the language being selected by `\ifcase\LANG\relax ... \or\relax ... \fi` constructs. The `\relax` primitives are necessary to avoid bad surprises.

```

4341 <fminitoc>
4342 \ProvidesFile{fminitoc.dtx}[2006/08/03 minitoc french documentation start file]
4343 \input{minitoc.dtx}
4344 </fminitoc>

```

Chapter 10

Commented code of `mtcoff.sty`

Contents

10.1	Why <code>mtcoff.sty</code> ?	395
10.2	Identification of the package	396
10.3	Faking counters and dimensions	396
10.4	Faking simple commands	398
10.5	Faking commands with one optional argument	399
10.6	Disabling the internal commands	400
10.7	Disabling the font commands	400
10.8	Disabling the <code>\mtcset...</code> commands	402
10.9	Disabling the new <code>\l@...</code> commands	402
10.10	Ignore the obsolete commands	402
10.11	Disabling the <code>\mtcselectlanguage</code> command	402
10.12	Disabling the <code>\mtcloadmlo</code> command	403
10.13	Disabling the commands for the horizontal rules	403
10.14	Disabling the commands for the page numbers	403
10.15	Disabling the mini-table features commands	404
10.16	Disabling miscellaneous flags and commands	405
10.17	Caution for some commands	405
10.18	Disabling commands for “coffee”	406
10.19	Disabling the <code>mtchideinmain...</code> environments	407
10.20	Inhibition of the <code>\mtc@saveXXXdepth</code> and <code>\mtc@restoreXXXdepth</code> commands	407
10.21	Disabling the <code>\mtcfixglossary</code> command	407
10.22	Disabling the <code>\mtcfixindex</code> command	407
10.23	Disabling the <code>\addstarred...</code> commands	408

10.1 Why `mtcoff.sty`?

The `minitoc` package [93, 94] 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:

```
\usepackage[...options...]{minitoc}
%\usepackage{mtcoff}
```

then the *minitoc* package is activated with the specified options. If you modify this two lines this way:

```
%\usepackage[...options...]{minitoc}
\usepackage{mtcoff}
```

then the *minitoc* package is deactivated 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` First, we identify the package and check the version of \LaTeX ¹; we need the *mtcmess* package to write messages with unique identifiers.

```
4345 \*mtcoff)
4346 \NeedsTeXFormat{LaTeX2e}%
4347 \ProvidesPackage{mtcoff}[2006/06/27 v3.0 The mtcoff package]
4348 \RequirePackage{mtcmess}[2006/03/14]
```

10.3 Faking counters and dimensions

`\count@` As *minitoc* declares some counters and dimensions registers, we fake them using
`\dimen@` `\count@` or `\dimen@`. For `\mtcskipamount`, we must use its default definition,
`\c@minitocdepth` `\bigskipamount`.

```
\mtcindent
\mtcskipamount 4349 \let\c@minitocdepth\count@
\bigskipamount 4350 \let\mtcindent\dimen@
\c@parttocdepth 4351 \let\mtcskipamount\bigskipamount
\mtcindent 4352 \let\c@parttocdepth\count@
\c@secttocdepth
\stcindent
```

¹ 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*.

```

4353 \let\ptcindent\dimen@
4354 \let\c@secttocdepth\count@
4355 \let\stcindent\dimen@

```

```

\c@mtc The basic mini-table counters are provided by \count@:
\c@ptc
\c@stc 4356 \let\c@mtc\count@
4357 \let\c@ptc\count@
4358 \let\c@stc\count@

```

```

\mtcgapbeforeheads The gaps before and after parttoc heads receive their default values:
\mtcgapafterheads

```

```

4359 \def\mtcgapbeforeheads{50\p@}
4360 \def\mtcgapafterheads{40\p@}

```

```

\@ifundefined We must define the macros for the vertical kernings between the minitables and their
\kernafterparttoc before the bottom rule. The default values are used. We must issue a warning if one of
\kernafterpartlof these macros is used.
\kernafterpartlot

```

```

\kernaftersecttoc 4361 \@ifundefined{part}{}%
\kernaftersectlof 4362 \def\kernafterparttoc{%
\kernaftersectlot 4363 \mtcoffwarn@true
\kernafterminitoc 4364 \mtcPackageWarning[<F0008>]{mtcoff}%
\kernafterminilof 4365 {The macro \string\kernafterparttoc \MessageBreak
4366 should not be used out of context\MessageBreak}%
\kernafterminilot 4367 \kern-1.\baselineskip\kern.5ex}%
4368 \def\kernafterpartlof{%
4369 \mtcoffwarn@true
4370 \mtcPackageWarning[<F0008>]{mtcoff}%
4371 {The macro \string\kernafterpartlof \MessageBreak
4372 should not be used out of context\MessageBreak}%
4373 \kern-1.\baselineskip\kern.5ex}%
4374 \def\kernafterpartlot{%
4375 \mtcoffwarn@true
4376 \mtcPackageWarning[<F0008>]{mtcoff}%
4377 {The macro \string\kernafterpartlot \MessageBreak
4378 should not be used out of context\MessageBreak}%
4379 \kern-1.\baselineskip\kern.5ex}%
4380 }%
4381 \@ifundefined{chapter}{%
4382 \@ifundefined{section}{}%
4383 }%
4384 \def\kernaftersecttoc{%
4385 \mtcoffwarn@true
4386 \mtcPackageWarning[<F0008>]{mtcoff}%
4387 {The macro \string\kernaftersecttoc \MessageBreak
4388 should not be used out of context\MessageBreak}%
4389 \kern-1.\baselineskip\kern.5ex}%

```

```

4390 \def\kernaftersectlof{%
4391 \mtcoffwarn@true
4392 \mtcPackageWarning[<F0008>]{mtcoff}%
4393 {The macro \string\kernaftersectlof \MessageBreak
4394 should not be used out of context\MessageBreak}%
4395 \kern-1.\baselineskip\kern.5ex}%
4396 \def\kernaftersectlot{%
4397 \mtcoffwarn@true
4398 \mtcPackageWarning[<F0008>]{mtcoff}%
4399 {The macro \string\kernaftersectlot \MessageBreak
4400 should not be used out of context\MessageBreak}%
4401 \kern-1.\baselineskip\kern.5ex}%
4402 }%
4403 }%
4404 {%
4405 \def\kernafterminitoc{%
4406 \mtcoffwarn@true
4407 \mtcPackageWarning[<F0008>]{mtcoff}%
4408 {The macro \string\kernafterminitoc \MessageBreak
4409 should not be used out of context\MessageBreak}%
4410 \kern-.5\baselineskip\kern.5ex}%
4411 \def\kernafterminilof{%
4412 \mtcoffwarn@true
4413 \mtcPackageWarning[<F0008>]{mtcoff}%
4414 {The macro \string\kernafterminilof \MessageBreak
4415 should not be used out of context\MessageBreak}%
4416 \kern-1.\baselineskip\kern0.ex}%
4417 \def\kernafterminilot{%
4418 \mtcoffwarn@true
4419 \mtcPackageWarning[<F0008>]{mtcoff}%
4420 {The macro \string\kernafterminilot \MessageBreak
4421 should not be used out of context\MessageBreak}%
4422 \kern-1.\baselineskip\kern0.ex}%
4423 }%

```

10.4 Faking simple commands

```

\mtcskip Some user commands are easy to fake:
\fakeableofcontents
\fakeinfofigures 4424 \let\mtcskip\relax
\fakeinfoftables 4425 \let\fakeableofcontents\relax
\adjustptc 4426 \let\fakeinfofigures\relax
\adjustmtc 4427 \let\fakeinfoftables\relax
\adjuststc 4428 \newcommand{\adjustptc}[1][1]{\relax}
\decrementptc 4429 \newcommand{\adjustmtc}[1][1]{\relax}
\decrementmtc 4430 \newcommand{\adjuststc}[1][1]{\relax}
\decrementstc 4431 \let\decrementptc\relax
\decrementstc 4432 \let\decrementmtc\relax
\incrementptc 4433 \let\decrementstc\relax
\incrementmtc 4434 \let\incrementptc\relax
\incrementstc

```

```
4435 \let\incrementmtc\relax
4436 \let\incrementstc\relax
```

```
\partend   The following commands are not directly called by the user, in normal circumstances, but
\partbegin must be faked:
\chapterend
\chapterbegin 4437 \let\partend\relax
\sectend      4438 \let\partbegin\relax
\sectbegin    4439 \let\chapterend\relax
              4440 \let\chapterbegin\relax
              4441 \let\sectend\relax
              4442 \let\sectbegin\relax
```

10.5 Faking commands with one optional argument

`\gobbleopt@` The user commands with an optional argument are faked using the internal `\ifnextchar` (to get the optional argument) and the new utility command `\gobbleopt@`.

```
4443 \def\gobbleopt@[#1]{\relax}
```

`\@ifnextchar` Commands for part level mini-tables:

```
\doparttoc
\dopartlof 4444 \def\doparttoc{\@ifnextchar[\gobbleopt@]{\gobbleopt@[1]}}
\dopartlot 4445 \def\dopartlof{\@ifnextchar[\gobbleopt@]{\gobbleopt@[1]}}
\parttoc   4446 \def\dopartlot{\@ifnextchar[\gobbleopt@]{\gobbleopt@[1]}}
\partlof   4447 \def\parttoc{\@ifnextchar[\gobbleopt@]{\gobbleopt@[1]}}
\partlot   4448 \def\partlof{\@ifnextchar[\gobbleopt@]{\gobbleopt@[1]}}
           4449 \def\partlot{\@ifnextchar[\gobbleopt@]{\gobbleopt@[1]}}
```

`\@ifnextchar` Commands for chapter level mini-tables:

```
\dominitoc
\dominilof 4450 \def\dominitoc{\@ifnextchar[\gobbleopt@]{\gobbleopt@[1]}}
\dominilot 4451 \def\dominilof{\@ifnextchar[\gobbleopt@]{\gobbleopt@[1]}}
\minitoc   4452 \def\dominilot{\@ifnextchar[\gobbleopt@]{\gobbleopt@[1]}}
\minilof   4453 \def\minitoc{\@ifnextchar[\gobbleopt@]{\gobbleopt@[1]}}
\minilot   4454 \def\minilof{\@ifnextchar[\gobbleopt@]{\gobbleopt@[1]}}
           4455 \def\minilot{\@ifnextchar[\gobbleopt@]{\gobbleopt@[1]}}
```

```

\@ifnextchar  Commands for section level mini-tables:
\dosecttoc
\dosectlof 4456 \def\dosecttoc{\@ifnextchar[{\gobbleopt@}{\gobbleopt@[1]}}
\dosectlot 4457 \def\dosectlof{\@ifnextchar[{\gobbleopt@}{\gobbleopt@[1]}}
\secttoc 4458 \def\dosectlot{\@ifnextchar[{\gobbleopt@}{\gobbleopt@[1]}}
\sectlof 4459 \def\secttoc{\@ifnextchar[{\gobbleopt@}{\gobbleopt@[1]}}
\sectlot 4460 \def\sectlof{\@ifnextchar[{\gobbleopt@}{\gobbleopt@[1]}}
4461 \def\sectlot{\@ifnextchar[{\gobbleopt@}{\gobbleopt@[1]}}

\@ifnextchar  Command \mtcprepare:
\mtcprepare
4462 \def\mtcprepare{\@ifnextchar[{\gobbleopt@}{\gobbleopt@[1]}}

```

10.6 Disabling the internal commands

```

\@gobbletwo  We need also to disable some minitoc commands, with \relax (macros with no
\tf@mtc      argument) or \@gobbletwo (macros with two arguments):
\mtc@string
\appendixmtc 4463 \let\tf@mtc\count@
\l@xchapter 4464 \let\mtc@string\relax
\chapter 4465 \let\appendixmtc\relax
\pchapter 4466 \let\l@xchapter\@gobbletwo
\psect 4467 \let\chapter\relax
\l@xpart 4468 \let\pchapter\relax
\part 4469 \let\psect\relax
\l@xsect 4470 \let\l@xpart\@gobbletwo
\sect 4471 \let\part\relax
4472 \let\l@xsect\@gobbletwo
4473 \let\xsect\relax

```

10.7 Disabling the font commands

\empty We disable the minitoc font commands (like \mtcSSfont) with \empty, because some users might have used:

```
\renewcommand{\mtcSSfont}{...}
```

which will not work if we use \relax here.


```

\ptcfont  Fonts for part level mini-tables:
\ptcCfont
\ptcSfont 4474 \let\ptcfont\empty
\ptcSSfont 4475 \let\ptcCfont\empty
\ptcSSSfont 4476 \let\ptcSfont\empty
\ptcPfont 4477 \let\ptcSSfont\empty
\ptcSPfont 4478 \let\ptcSSSfont\empty
\plffont 4479 \let\ptcPfont\empty
\plfSfont 4480 \let\ptcSPfont\empty
\pltfont 4481 \let\plffont\empty
\pltSfont 4482 \let\plfSfont\empty
\ptifont 4483 \let\pltfont\empty
          4484 \let\pltSfont\empty
          4485 \let\ptifont\empty

\mtcfont  Fonts for chapter level mini-tables:
\mtcSfont
\mtcSSfont 4486 \let\mtcfont\empty
\mtcSSSfont 4487 \let\mtcSfont\empty
\mtcPfont 4488 \let\mtcSSfont\empty
\mtcSPfont 4489 \let\mtcSSSfont\empty
\mlffont 4490 \let\mtcPfont\empty
\mlfSfont 4491 \let\mtcSPfont\empty
\mltfont 4492 \let\mlffont\empty
\mltSfont 4493 \let\mlfSfont\empty
\mtifont 4494 \let\mltfont\empty
          4495 \let\mltSfont\empty
          4496 \let\mtifont\empty

\stcfont  Fonts for section level mini-tables:
\stcSSfont
\stcSSSfont 4497 \let\stcfont\empty
\stcPfont 4498 \let\stcSSfont\empty
\stcSPfont 4499 \let\stcSSSfont\empty
\slffont 4500 \let\stcPfont\empty
\slfSfont 4501 \let\stcSPfont\empty
\sltfont 4502 \let\slffont\empty
\sltSfont 4503 \let\slfSfont\empty
\stifont 4504 \let\sltfont\empty
          4505 \let\sltSfont\empty
          4506 \let\stifont\empty

```

```

\coffeeFont  Font for “coffee” lines:

          4507 \let\coffeeFont\empty

```

10.8 Disabling the `\mtcset...` commands

```

\@gobbletwo These commands use two or three mandatory arguments:
\mtcsetdepth
\mtcsetfont 4508 \let\mtcsetdepth\@gobbletwo
\mtcsetttitlefont 4509 \def\mtcsetfont#1#2#3{\empty}
\mtcsetttitle 4510 \let\mtcsetttitlefont\@gobbletwo
\mtcsetformat 4511 \let\mtcsetttitle\@gobbletwo
\mtcsetfeature 4512 \def\mtcsetformat#1#2#3{\empty}
\mtcsetpagenumbers 4513 \def\mtcsetfeature#1#2#3{\empty}
\mtcsetrules 4514 \let\mtcsetpagenumbers\@gobbletwo
4515 \let\mtcsetrules\@gobbletwo

```

10.9 Disabling the new `\l@...` commands

```

\l@starpart The minitoc package defines the \l@starXXX commands to format TOC entries for
\l@starchapter starred sectioning commands. We reset to the unstarred version, when necessary:
\l@starsection
\l@starsubsection 4516 \@ifundefined{part}{}{\let\l@starpart\l@part}
\l@starsubsubsection 4517 \@ifundefined{chapter}{}{\let\l@starchapter\l@chapter}
\l@starparagraph 4518 \@ifundefined{section}{}{\let\l@starsection\l@section}
\l@starsubparagraph 4519 \@ifundefined{subsection}{}{\let\l@starsubsection\l@subsection}
4520 \@ifundefined{subsubsection}{}{\let\l@starsubsubsection\l@subsubsection}
4521 \@ifundefined{paragraph}{}{\let\l@starparagraph\l@paragraph}
4522 \@ifundefined{subparagraph}{}{\let\l@starsubparagraph\l@subparagraph}

```

10.10 Ignore the obsolete commands

```

\@gobble We just ignore the obsolete commands (with one mandatory argument):
\firstpartis
\firstchapteris 4523 \let\firstpartis\@gobble
\firstsectionis 4524 \let\firstchapteris\@gobble
4525 \let\firstsectionis\@gobble

```

10.11 Disabling the `\mtcselectlanguage` command

```

\@gobble This command has one mandatory argument:
\mtcselectlanguage
4526 \let\mtcselectlanguage\@gobble

```

10.12 Disabling the `\mtcloadmlo` command

```
\@gobble This command has one mandatory argument:
\mtcloadmlo
4527 \let\mtcloadmlo\@gobble
```

10.13 Disabling the commands for the horizontal rules

```
\ptcrule These commands have no argument:
\noptcrule
\mtcrule 4528 \let\ptcrule\relax
\nomtcrule 4529 \let\noptcrule\relax
\stcrule 4530 \let\mtcrule\relax
\nostcrule 4531 \let\nomtcrule\relax
\plfrule 4532 \let\stcrule\relax
\noplfrule 4533 \let\nostcrule\relax
\mlfrule 4534 \let\plfrule\relax
\nomlfrule 4535 \let\noplfrule\relax
\slfrule 4536 \let\mlfrule\relax
\noslfrule 4537 \let\nomlfrule\relax
\pltrule 4538 \let\noslfrule\relax
\nopltrule 4539 \let\pltrule\relax
\mltrule 4540 \let\nopltrule\relax
\nomltrule 4541 \let\mltrule\relax
\sltrule 4542 \let\nomltrule\relax
\nosltrule 4543 \let\sltrule\relax
4544 \let\nosltrule\relax
4545 \let\nosltrule\relax
```

10.14 Disabling the commands for the page numbers

```
\ptcpagenumbers These commands have no argument:
\noptcpagenumbers
\mtcpagenumbers 4546 \let\mtcpagenumbers\relax
\nomtcpagenumbers 4547 \let\nomtcpagenumbers\relax
\stcpagenumbers 4548 \let\stcpagenumbers\relax
\nostcpagenumbers 4549 \let\nostcpagenumbers\relax
\plfpagenumbers 4550 \let\ptcpagenumbers\relax
\noplfpagenumbers 4551 \let\noptcpagenumbers\relax
\mlfpagenumbers 4552 \let\mlfpagenumbers\relax
\nomlfpagenumbers 4553 \let\nomlfpagenumbers\relax
\slfpagenumbers 4554 \let\slfpagenumbers\relax
\noslfpagenumbers 4555 \let\noslfpagenumbers\relax
4556 \let\plfpagenumbers\relax
\pltpagenumbers
\nopltpagenumbers
\mltpagenumbers
\nomltpagenumbers
\sltpagenumbers
\nosltpagenumbers
```



```

\beforesecttoc  Commands for section level mini-tables:
\beforesectlof
\beforesectlot 4582 \let\beforesecttoc\empty
\aftersecttoc  4583 \let\beforesectlof\empty
\aftersectlof  4584 \let\beforesectlot\empty
\aftersectlot  4585 \let\aftersecttoc\empty
\thispagesecttocstyle 4586 \let\aftersectlof\empty
\thispagesectlofsty 4587 \let\aftersectlot\empty
\thispagesectlotstyle 4588 \let\thispagesecttocstyle\empty
\thispagesectlotstyle 4589 \let\thispagesectlofsty\empty
\thispagesectlotstyle 4590 \let\thispagesectlotstyle\empty

```

10.16 Disabling miscellaneous flags and commands

```

\if@longextensions@ There are some flags and commands that it is wise to declare:
\iftightmtc
\ifktightmtc 4591 \newif\if@longextensions@ \@longextensions@true
\ifundottedmtc 4592 \newif\iftightmtc \tightmtcfalse
\l@listof 4593 \newif\ifktightmtc \ktightmtcfalse
\ifmtcsecondpart 4594 \newif\ifundottedmtc \undottedmtcfalse
\chapter 4595 \newif\ifmtcsecondpart \mtcsecondpartfalse
4596 \let\l@listof\chapter

```



10.17 Caution for some commands

`\AtBeginDocument` 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 a `\AtEndDocument` block to signal that you have used these commands:

```

4597 \newif\ifmtcoffwarn@ \mtcoffwarn@false
4598 \AtEndDocument{\ifmtcoffwarn@
4599   \mtcPackageWarningNoLine[<F0007>]{mtcoff}%
4600   {You should scan (backwards) your .log
4601   \MessageBreak
4602   file to find some commands needing
4603   \MessageBreak
4604   to be replaced if you decide to
4605   \MessageBreak
4606   DEFINITELY stop using minitoc for this
4607   \MessageBreak
4608   document. It is more wise to keep the
4609   \MessageBreak
4610   \string\usepackage\space lines for minitoc and mtcoff
4611   \MessageBreak
4612   and to comment out only one of them}
4613 \fi}

```

```

\mtcaddchapter Then these commands are disabled and they set the flag and give a warning (useful to get
\mtcaddsection the line number):
\mtcaddpart
\ifmtcoffwarn@ 4614 \newcommand{\mtcaddchapter}[1][\mtcoffwarn@true
\mtc@ck 4615 \mtcPackageWarning[<F0004>]{mtcoff}%
\addcontentsline 4616 {\protect\mtcaddchapter{...} should be replaced\MessageBreak
4617 by \protect\addcontentsline{toc}{chapter}{...}\MessageBreak}
4618 \def\mtc@ck{#1}
4619 \ifx\mtc@ck\empty
4620 \else
4621 \addcontentsline{toc}{chapter}{#1}%
4622 \fi}
4623 \newcommand{\mtcaddsection}[1][\mtcoffwarn@true
4624 \mtcPackageWarning[<F0006>]{mtcoff}%
4625 {\protect\mtcaddsection{...} should be replaced\MessageBreak
4626 by \protect\addcontentsline{toc}{section}{...}\MessageBreak}
4627 \def\mtc@ck{#1}
4628 \ifx\mtc@ck\empty
4629 \else
4630 \addcontentsline{toc}{part}{#1}%
4631 \fi}
4632 \newcommand{\mtcaddpart}[1][\mtcoffwarn@true
4633 \mtcPackageWarning[<F0005>]{mtcoff}%
4634 {\protect\mtcaddpart{...} should be replaced\MessageBreak
4635 by \protect\addcontentsline{toc}{part}{...}\MessageBreak}
4636 \def\mtc@ck{#1}
4637 \ifx\mtc@ck\empty
4638 \else
4639 \addcontentsline{toc}{part}{#1}%
4640 \fi}

```

10.18 Disabling commands for “coffee”

```

\addcoffeeline We disable the commands relative to “coffee” lines, and the specific version of contents
\coffeeline lines without leaders of dots:
\@gobble
\@Undottedtocline 4641 \def\addcoffeeline#1#2#3{\relax}
\@Undottedtoclinep 4642 \let\coffeeline\@gobble
4643 \let\l@coffee\relax
4644 \def\@Undottedtocline#1#2#3#4#5{\relax}
4645 \def\@Undottedtoclinep#1#2#3#4#5{\relax}

```

10.19 Disabling the `mtchideinmain...` environments

`mtchideinmaintoc` These environments accept one optional argument:
`mtchideinmainlof`
`mtchideinmainlot`

```

4646 \newenvironment{mtchideinmaintoc}[1][-1]%
4647     {\empty}{\empty}
4648 \newenvironment{mtchideinmainlof}[1][-1]%
4649     {\empty}{\empty}
4650 \newenvironment{mtchideinmainlot}[1][-1]%
4651     {\empty}{\empty}

```

10.20 Inhibition of the `\mtc@saveXXXdepth` and `\mtc@restoreXXXdepth` commands

`\mtc@savetocdepth` We must inhibit these commands, inserted in the `.toc`, `.lof` and `.lot` files by the hiding
`\mtc@savelofdepth` commands. So we will not have to delete these files when switching from the `minitoc`
`\mtc@savelotdepth` package to the `mtc` package.
`\mtc@restoretocdepth`
`\mtc@restorelofdepth`

```

4652 \let\mtc@savetocdepth\empty
4653 \let\mtc@savelofdepth\empty
4654 \let\mtc@savelotdepth\empty
4655 \let\mtc@restoretocdepth\empty
4656 \let\mtc@restorelofdepth\empty
4657 \let\mtc@restorelotdepth\empty

```

10.21 Disabling the `\mtcfixglossary` command

`\mtcfixglossary` This command accepts one optional argument:

```

4658 \newcommand{\mtcfixglossary}[1][\relax]

```

10.22 Disabling the `\mtcfixindex` command

`\mtcfixindex` This command accepts one optional argument:

```

4659 \newcommand{\mtcfixindex}[1][\relax]

```

10.23 Disabling the `\addstarred...` commands

`\ifmtcoffwarn@` These commands should be replaced by standard commands, but `mtcoff` simulates and gives a warning, which will be reminded at the end of document:

`\addstarredpart`

`\addstarredchapter`

`\addstarredsection`

```

4660 \def\addstarredpart#1{\mtcoffwarn@true
\addcontentsline 4661   \mtcPackageWarning[<F0002>]{mtcoff}%
4662   {\protect\addstarredpart{...} should be replaced by\MessageBreak
4663   \protect\addcontentsline{toc}{part}{...}\MessageBreak}
4664   \addcontentsline{toc}{part}{#1}}
4665 \def\addstarredchapter#1{\mtcoffwarn@true
4666   \mtcPackageWarning[<F0001>]{mtcoff}%
4667   {\protect\addstarredchapter{...} should be replaced by\MessageBreak
4668   \protect\addcontentsline{toc}{chapter}{...}\MessageBreak}
4669   \addcontentsline{toc}{chapter}{#1}}
4670 \def\addstarredsection#1{\mtcoffwarn@true
4671   \mtcPackageWarning[<F0003>]{mtcoff}%
4672   {\protect\addstarredsection{...} should be replaced by\MessageBreak
4673   \protect\addcontentsline{toc}{section}{...}\MessageBreak}
4674   \addcontentsline{toc}{section}{#1}}

```

And the `mtcoff` package is terminated.

```
4675 </mtcoff>
```


Chapter 11

Commented code of `mtcmess.sty`

```
\mtcPackageInfo  To make easier the search of a message in the documentation1, we will assign an unique
\PackageInfo    identifier to each message of the minitoc and mtcoff packages. As the standard com-
\MessageBreak   mands for such messages do not include this feature, we make extended versions, with
\mtcPackageWarning \PackageWarning the same syntax, plus a first optional argument:
\PackageWarning
\mtcPackageWarningNoLine 4676 <{*mtcmess>
\PackageWarningNoLine 4677 \ProvidesPackage{mtcmess}[2006/03/14]%
\mtcPackageError 4678 \NeedsTeXFormat{LaTeX2e}[1996/06/01]%
\PackageError 4679 \newcommand{\mtcPackageInfo}[3][[]%
4680     {\PackageInfo{#2}{#1\MessageBreak #3}}%
4681 \newcommand{\mtcPackageWarning}[3][[]%
4682     {\PackageWarning{#2}{#1\MessageBreak #3}}%
4683 \newcommand{\mtcPackageWarningNoLine}[3][[]%
4684     {\PackageWarningNoLine{#2}{#1\MessageBreak #3}}%
4685 \newcommand{\mtcPackageError}[4][[]%
4686     {\PackageError{#2}{#1\MessageBreak #3}{#4}}%
4687 </mtcmess>
```

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 will be used by at least two packages (`minitoc` and `mtcoff`) and because they could be useful for other packages.

¹ By example, using the search facility of some PDF reader utility.

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.cls class. This correction is no more necessary after the 2005/09/25 version of memoir.cls.

```
4688 (*mtcpatchmem)
4689 \NeedsTeXFormat{LaTeX2e}[1996/06/01]%
4690 \ProvidesPackage{mtcpatchmem}%
4691   [2005/09/30 v47 Package mtcpatchmem]
4692 \RequirePackage{mtcmess}[2006/03/14]
4693 \mtcPackageInfo[<M0001>]{mtcpatchmem}%
4694   {mtcpatchmem package to patch the memoir class\@gobble}
4695 \renewcommand{\@m@chapter}[1][{}]{%
4696   \def\ch@pt@c{#1}% capture first optional arg
4697   \@ifnextchar[{\@chapter}{\@chapter[]}%
4698 }
4699 \def\@chapter[#1]#2{%
4700 % if |\ch@pt@c| is empty, no [ was found at all. Use #2| as
4701 % entry for all fields.
4702   \ifx\ch@pt@c\@empty
4703     \def\f@rtoc{#2}%
4704     \def\f@rhdr{#2}%
4705   \else
4706 % otherwise at least one [ was found. If |#1| is empty then only
4707 % one was found.
4708     \let\f@rtoc\ch@pt@c
4709     \ifx\@empty#1\@empty
4710       \let\f@rhdr\ch@pt@c
4711     \else
4712       \def\f@rhdr{#1}%
4713     \fi
4714   \fi
4715   \ifnum \c@secnumdepth >\m@ne\relax
4716     \if@mainmatter
4717       \refstepcounter{chapter}%
```

```

4718     \fi
4719 \fi
4720 \chaptermark{\f@rhdr}%
4721 \ifartopt
4722     \@makechapterhead{#2}%
4723     \@afterheading
4724 \else
4725     \insertchapterspace
4726     \if@twocolumn
4727         \@topnewpage[\@makechapterhead{#2}]%
4728     \else
4729         \@makechapterhead{#2}%
4730     \fi
4731     \@afterheading
4732 \fi
4733 \ifnum \c@secnumdepth >\m@ne\relax
4734     \if@mainmatter
4735         \ifanappendix
4736             \addcontentsline{toc}{appendix}{%
4737                 \protect\chapternumberline{\thechapter}\f@rtoc}%
4738         \else
4739             \addcontentsline{toc}{chapter}{%
4740                 \protect\chapternumberline{\thechapter}\f@rtoc}%
4741         \fi
4742     \else
4743         \addcontentsline{toc}{chapter}{\f@rtoc}%
4744     \fi
4745 \else
4746     \addcontentsline{toc}{chapter}{\f@rtoc}%
4747 \fi
4748 \ifheadnameref\M@gettitle{\f@rhdr}\else\M@gettitle{\f@rtoc}\fi
4749 }
4750 </mtcpatchmem>

```

Chapter 13

The language definition (.mld) and object (.mlo) files

Contents

13.1	“Acadian” language: acadian.mld	416
13.2	“Acadien” language: acadien.mld	417
13.3	“Afrikaan” language: afrikaan.mld	417
13.4	“Afrikaans” language: afrikaans.mld	417
13.5	“Albanian” language: albanian.mld	418
13.6	“American” language: american.mld	418
13.7	“Arab” language: arab.mld	418
13.8	“Arab2” language: arab2.mld	419
13.9	“Arabi” language: arabi.mld	419
13.10	“Arabic” language: arabic.mld	420
13.11	“Armenian” language: armenian.mld	420
13.12	“Australian” language: australian.mld	421
13.13	“Austrian” language: austrian.mld	421
13.14	“Bahasa” language: bahasa.mld	421
13.15	“Bahasai” language: bahasai.mld	422
13.16	“Bahasam” language: bahasam.mld	422
13.17	“Bangla” language: bangla.mld	423
13.18	“Basque” language: basque.mld	423
13.19	“Bicig” language: bicig.mld	424
13.20	“Bicig2” language: bicig2.mld	424
13.21	“Bicig3” language: bicig3.mld	425
13.22	“Bithe” language: bithe.mld	426
13.23	“Brazil” language: brazil.mld	426
13.24	“Brazilian” language: brazilian.mld	427
13.25	“Breton” language: breton.mld	427
13.26	“British” language: british.mld	428
13.27	“Bulgarian” language: bulgarian.mld	428
13.28	“Bulgarianb” language: bulgarianb.mld	429
13.29	“Buryat” language: buryat.mld	429
13.30	“Buryat2” language: buryat2.mld	430

13.31	“Canadian” language: canadian.mld	430
13.32	“Canadien” language: canadien.mld	431
13.33	“Castillan” language: castillan.mld	431
13.34	“Castillian” language: castillian.mld	431
13.35	“Catalan” language: catalan.mld	431
13.36	“Chinese1” language: chinese1.mld and chinese1.mlo	432
13.37	“Chinese2” language: chinese2.mld and chinese2.mlo	432
13.38	“Croatian” language: croatian.mld	433
13.39	“Czech” language: czech.mld	433
13.40	“Danish” language: danish.mld	434
13.41	“Dutch” language: dutch.mld	434
13.42	“English” language: english.mld	435
13.43	“English1” language: english1.mld	436
13.44	“English2” language: english2.mld	436
13.45	“Esperant” language: esperant.mld	437
13.46	“Esperanto” language: esperanto.mld	437
13.47	“Estonian” language: estonian.mld	438
13.48	“Ethiopia” language: ethiopia.mld	438
13.49	“Ethiopian” language: ethiopian.mld	439
13.50	“Ethiopian2” language: ethiopian2.mld	439
13.51	“Farsi1” language: farsi1.mld and farsi1.mlo	440
13.52	“Farsi2” language: farsi2.mld and farsi2.mlo	440
13.53	“Farsi3” language: farsi3.mld	441
13.54	“Finnish” language: finnish.mld	441
13.55	“Finnish2” language: finnish2.mld	442
13.56	“Francais” language: francais.mld	442
13.57	“French” language: french.mld	443
13.58	“French1” language: french1.mld	443
13.59	“French2” language: french2.mld	444
13.60	“Frenchb” language: frenchb.mld	445
13.61	“Frenchle” language: frenchle.mld	445
13.62	“Frenchpro” language: frenchpro.mld	445
13.63	“Galician” language: galician.mld	445
13.64	“German” language: german.mld	446
13.65	“Germanb” language: germanb.mld	446
13.66	“Germanb2” language: germanb2.mld	447
13.67	“Greek” language: greek.mld	447
13.68	“Greek-mono” language: greek-mono.mld	448
13.69	“Greek-polydemo” language: greek-polydemo.mld	449
13.70	“Greek-polykatha” language: greek-polykatha.mld	450
13.71	“Guarani” language: guarani.mld	451
13.72	“Hangul1” language: hangul1.mld and hangul1.mlo	451
13.73	“Hangul2” language: hangul2.mld and hangul2.mlo	452
13.74	“Hangul3” language: hangul3.mld and hangul3.mlo	453
13.75	“Hangul4” language: hangul4.mld and hangul4.mlo	453
13.76	“Hangul-u8” language: hangul-u8.mld and hangul-u8.mlo	454
13.77	“Hanja1” language: hanja1.mld and hanja1.mlo	454
13.78	“Hanja2” language: hanja2.mld and hanja2.mlo	455
13.79	“Hanja-u8” language: hanja-u8.mld and hanja-u8.mlo	455
13.80	“Hebrew” language: hebrew.mld	456
13.81	“Hebrew2” language: hebrew2.mld	457
13.82	“Hungarian” language: hungarian.mld	457
13.83	“Icelandic” language: icelandic.mld	458

13.84	“Indon” language: indon.mld	458
13.85	“Indonesian” language: indonesian.mld	458
13.86	“Interlingua” language: interlingua.mld	459
13.87	“Irish” language: irish.mld	459
13.88	“Italian” language: italian.mld	460
13.89	“Italian2” language: italian2.mld	460
13.90	“Japanese” language: japanese.mld and japanese.mlo	461
13.91	“Japanese2” language: japanese2.mld and japanese2.mlo	461
13.92	“Japanese3” language: japanese3.mld and japanese3.mlo	462
13.93	“Japanese4” language: japanese4.mld and japanese4.mlo	462
13.94	“Japanese5” language: japanese5.mld and japanese.mlo	463
13.95	“Kannada” language: kannada.mld	463
13.96	“Khalka” language: khalkha.mld	464
13.97	“Latin” language: latin.mld	464
13.98	“Latin2” language: latin2.mld	465
13.99	“Latvian” language: latvian.mld	465
13.100	“Letton” language: letton.mld	466
13.101	“Lithuanian” language: lithuanian.mld	466
13.102	“Lowersorbian” language: lowersorbian.mld	467
13.103	“Lsorbian” language: lsorbian.mld	467
13.104	“Magyar” language: magyar.mld	468
13.105	“Magyar2” language: magyar2.mld	468
13.106	“Magyar3” language: magyar.mld	469
13.107	“Malay” language: malay.mld	469
13.108	“Malayalam-keli” language: malayalam-keli.mld	469
13.109	“Malayalam-omega” language: malayalam-omega.mld and malayalam-omega.mlo	470
13.110	“Malayalam-rachana” language: malayalam-rachana.mld	470
13.111	“Malayalam-rachana2” language: malayalam-rachana2.mld	471
13.112	“Manju” language: manju.mld	472
13.113	“Meyalu” language: meyalu.mld	472
13.114	“Mongol” language: mongol.mld	472
13.115	“Naustrian” language: naustrian.mld	473
13.116	“Newzealand” language: newzealand.mld	473
13.117	“Ngerman” language: ngerman.mld	473
13.118	“Ngermanb” language: ngermanb.mld	473
13.119	“Ngermanb2” language: ngermanb2.mld	474
13.120	“Norsk” language: norsk.mld	475
13.121	“Norsk2” language: norsk2.mld	475
13.122	“Nynorsk” language: nynorsk.mld	476
13.123	“Nynorsk2” language: nynorsk2.mld	476
13.124	“Polish” language: polish.mld	477
13.125	“Polish2” language: polish2.mld	477
13.126	“Polski” language: polski.mld	478
13.127	“Portuges” language: portuges.mld	478
13.128	“Portuguese” language: portuguese.mld	479
13.129	“Romanian” language: romanian.mld	479
13.130	“Romanian2” language: romanian2.mld	480
13.131	“Romanian3” language: romanian3.mld	480
13.132	“Russian” language: russian.mld	481
13.133	“Russian2m” language: russian2m.mld	481
13.134	“Russian2o” language: russian2o.mld	482
13.135	“Russianb” language: russianb.mld	483

13.136	“Russianc” language: russianc.mld	484
13.137	“Russian-cca” language: russian-cca.mld and russian-cca.mlo	485
13.138	“Russian-cca1” language: russian-cca1.mld and russian-cca1.mlo	485
13.139	“Russian-lh” language: russian-lh.mld and russian-lh.mlo	486
13.140	“Russian-lhcyralt” language: russian-lhcyralt.mld and russian-lhcyralt.mlo	486
13.141	“Russian-lhcyrkoi” language: russian-lhcyrkoi.mld and russian-lhcyrkoi.mlo	487
13.142	“Russian-lhcyrwin” language: russian-lhcyrwin.mld and russian-lhcyrwin.mlo	487
13.143	“Samin” language: samini.mld	488
13.144	“Scottish” language: scottish.mld	488
13.145	“Serbian” language: serbian.mld	489
13.146	“Serbianc” language: serbianc.mld	489
13.147	“Slovak” language: slovak.mld	490
13.148	“Slovene” language: slovene.mld	490
13.149	“Spanish” language: spanish.mld	491
13.150	“Spanish2” language: spanish2.mld	492
13.151	“Spanish3” language: spanish3.mld	492
13.152	“Spanish4” language: spanish4.mld	493
13.153	“Swedish” language: swedish.mld	493
13.154	“Swedish2” language: swedish2.mld	494
13.155	“Thai” language: thai.mld and thai.mlo	494
13.156	“Turkish” language: turkish.mld	495
13.157	“Uighur” language: uighur.mld	495
13.158	“Uighur2” language: uighur2.mld	495
13.159	“Uighur3” language: uighur3.mld	496
13.160	“UKenglish” language: UKenglish.mld	496
13.161	“Ukraineb” language: ukraineb.mld	496
13.162	“Ukrainian” language: ukrainian.mld	496
13.163	“Uppersorbian” language: uppersorbian.mld	497
13.164	“USenglish” language: USenglish.mld	498
13.165	“Usorbian” language: usorbian.mld	498
13.166	“Vietnam” language: vietnam.mld	498
13.167	“Vietnamese” language: vietnamese.mld	499
13.168	“Welsh” language: welsh.mld	499
13.169	“Xalx” language: xalx.mld	500
13.170	“Xalx2” language: xalx2.mld	500
13.171	“Xalx3” language: xalx3.mld	501

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. It is loaded either via a package option¹ in the \usepackage command for the minitoc package, either via the command:

```
\mtcselectlanguage \mtcselectlanguage{\language}
```

¹ It can also be a global option for the document.

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):

<code>\ptctitle</code>	• <code>\ptctitle</code>	• <code>\mtctitle</code>	• <code>\stctitle</code>
<code>\plftitle</code>	• <code>\plftitle</code>	• <code>\mlftitle</code>	• <code>\slftitle</code>
<code>\plttitle</code>	• <code>\plttitle</code>	• <code>\mlttitle</code>	• <code>\slttitle</code>



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 [29, 30, 31, 48], the CJK system [143, 144], the $\mathcal{L}\text{A}\text{T}\text{E}\text{X}$ system [124], the ANTOMEGA system [127], the ArabTeX [130, 131], BangTeX [169], ethiop [23], FarsiTeX², guarani [26], Malayalam [3, 4], MonTeX [87, 89], or ArmTeX [90] 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 languages³, 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 file⁴, not generated by the .dtx files in the current version of minitoc package. The



adequate input encoding must be set up by the user *before* loading the .mld file via `\mtcselectlanguage`.

`filecontents` To go around this limitation, the `minitoc.ins` file uses `filecontents` environments to generate the .mlo files.

13.1 “Acadian” language: `acadian.mld`

The `acadian` language⁵ is just french, so we load the `french.mld` file (see section 13.57 on page 443):

```
4751 <acadian>
4752 \ProvidesFile{acadian.mld}[2004/12/14]
4753 \mtcselectlanguage{french}%
4754 </acadian>
```

² By Mohammad GHODSI (ghodsi@rose.ipm.ac.ir) and FarsiTeX Project Group. See the FarsiTeX site at <http://www.farsitex.org>

³ Mainly for chinese, farsi (iranian), hangûl (korean), hanja (korean), japanese, malayalam-omega, thai, and russian variants.

⁴ The extension .mlo means *minitoc language object*.

⁵ Spoken in some parts of the south of the USA, like Louisiane.

13.2 “Acadien” language: `acadien.mld`

The “acadien” language⁶ is just french (“acadien” is the french term for “acadian”), so we load the `french.mld` file (see section 13.57 on page 443):

```
4755 <*acadien>
4756 \ProvidesFile{acadien.mld}[2004/12/14]
4757 \mtcselectlanguage{french}%
4758 </acadien>
```

13.3 “Afrikaan” language: `afrikaan.mld`

The titles for the “afrikaan” language⁷ are taken from the `dutch.dtx` file in the `babel` package [30, 31, 32]:

```
4759 <*afrikaan>
4760 \ProvidesFile{afrikaan.mld}[2006/01/13]
4761 %% Afrikaan(s) titles for minitoc.sty
4762 %% from dutch.dtx (babel)
4763 %% Braams, Johannes
4764 \def\ptctitle{Inhoudsopgawe}%
4765 \def\plftitle{Lys van figure}%
4766 \def\plttitle{Lys van tabelle}%
4767 %%
4768 \def\mtctitle{Inhoudsopgawe}%
4769 \def\mlftitle{Lys van figure}%
4770 \def\mlttitle{Lys van tabelle}%
4771 %%
4772 \def\stctitle{Inhoudsopgawe}%
4773 \def\slftitle{Lys van figure}%
4774 \def\slttitle{Lys van tabelle}%
4775 </afrikaan>
```

13.4 “Afrikaans” language: `afrikaans.mld`

“Afrikaans” is a synonym of “afrikaan”, so we just load `afrikaan.mld` (see section 13.3):

```
4776 <*afrikaans>
4777 \ProvidesFile{afrikaans.mld}[2004/12/14]
4778 \mtcselectlanguage{afrikaan}%
4779 </afrikaans>
```

⁶ Spoken in some parts of the south of the USA, like Louisiane.

⁷ Spoken in South Africa, it has dutch origins; compare with section 13.41 on page 434.

13.5 “Albanian” language: `albanian.mld`

The titles for the “albanian” language are taken from the `albanian.dtx` file in the `babel` package [30, 31, 75]:

```

4780 <*albanian>
4781 \ProvidesFile{albanian.mld}[2006/01/13]
4782 %% Albanian titles for minitoc.sty.
4783 %% from albabian.dtx (babel)
4784 %% Contribution of Adi Zaimi (zamilst at yahoo.com).
4785 \def\ptctitle{P\ "ermbajta}%
4786 \def\plftitle{Figurat}%
4787 \def\plttitle{Tabelat}%
4788 %%
4789 \def\mtctitle{P\ "ermbajta}%
4790 \def\mlftitle{Figurat}%
4791 \def\mlttitle{Tabelat}%
4792 %%
4793 \def\stctitle{P\ "ermbajta}%
4794 \def\slftitle{Figurat}%
4795 \def\slttitle{Tabelat}%
4796 </albanian>

```

13.6 “American” language: `american.mld`

The “american” language is just like “english”⁸, so we just load `english.mld` (see section 13.42 on page 435):

```

4797 <*american>
4798 \ProvidesFile{american.mld}[2004/12/14]
4799 \mtcselectlanguage{english}%
4800 </american>

```

13.7 “Arab” language: `arab.mld`

The titles for the “arab” language are taken from the Arab_T_E_X package [130, 131], which should be used, with the associated fonts.

```

4801 <*arab>
4802 \ProvidesFile{arab.mld}[1999/03/16]
4803 %% Arabic titles for minitoc.sty

```

⁸ It should be true for the mini-table titles; the languages themselves have some differences, like the hyphenation rules.

```

4804 %% Needs arabic fonts (cf. documentation of arabtex)
4805 %% (strings taken from arabtex; to be used with arabtex)
4806 \def\ptctitle{al-mu.htawayAtu}%
4807 \def\plftitle{qA'imaTu a.s-.suwari}%
4808 \def\plttitle{qA'imaTu al-^gadAwili}
4809 %%
4810 \def\mtctitle{al-mu.htawayAtu}%
4811 \def\mlftitle{qA'imaTu a.s-.suwari}%
4812 \def\mlttitle{qA'imaTu al-^gadAwili}%
4813 %%
4814 \def\stctitle{al-mu.htawayAtu}%
4815 \def\slftitle{qA'imaTu a.s-.suwari}%
4816 \def\slttitle{qA'imaTu al-^gadAwili}%
4817 </arab>

```

13.8 “Arab2” language: arab2.mld

The titles for the “arab2” language are taken from the Arab \TeX package [130, 131], which should be used, with the associated fonts. It is a variant of the “arab” language.

```

4818 <*arab2>
4819 \ProvidesFile{arab2.mld}[2006/03/31]
4820 %% Arabic titles for minitoc.sty. Variant.
4821 %% Needs arabic fonts (cf. documentation of arabtex)
4822 %% (strings taken from arabtex; to be used with arabtex)
4823 {\makeatletter\global\let\mtcArabTok@a@tok}%
4824 \def\ptctitle{\mtcArabTok(al-muHtawayAtu)}%
4825 \def\plftitle{\mtcArabTok(qAQAIMaTu aS-Suwari)}%
4826 \def\plttitle{\mtcArabTok(qAQAIMaTu al-GadAwili)}%
4827 %%
4828 \def\mtctitle{\mtcArabTok(al-muHtawayAtu)}%
4829 \def\mlftitle{\mtcArabTok(qAQAIMaTu aS-Suwari)}%
4830 \def\mlttitle{\mtcArabTok(qAQAIMaTu al-GadAwili)}%
4831 %%
4832 \def\stctitle{\mtcArabTok(al-muHtawayAtu)}%
4833 \def\slftitle{\mtcArabTok(qAQAIMaTu aS-Suwari)}%
4834 \def\slttitle{\mtcArabTok(qAQAIMaTu al-GadAwili)}%
4835 </arab2>

```

13.9 “Arabi” language: arabi.mld

The titles for the “arabi” language are taken from the Arabi package [116] (by Youssef JABRI), which should be used, with the associated fonts.

```

4836 (*arabi)
4837 \ProvidesFile{arabi.mld}[2006/07/27]
4838 %% Arabic titles for minitoc.sty.
4839 %% Needs arabic fonts (cf. documentation of the Arabi package,
4840 %% Youssef Jabri)
4841 %% (strings taken from arabi.ldf; to be used with arabi)
4842 \def\ptctitle{\R{\alef\lam\fa\ha\ra\seen}}%
4843 \def\plftitle{\R{\qaf\alef\yahamza\meem\T\space \alef\lam\alefhamza\sheen\kaf\alef\lam}}%
4844 \def\plttitle{\R{\qaf\alef\yahamza\meem\T\space \alef\lam\jeem\dal\alef\waw\lam}}%
4845 %%
4846 \def\mtctitle{\R{\alef\lam\fa\ha\ra\seen}}%
4847 \def\mlftitle{\R{\qaf\alef\yahamza\meem\T\space \alef\lam\alefhamza\sheen\kaf\alef\lam}}%
4848 \def\mlttitle{\R{\qaf\alef\yahamza\meem\T\space \alef\lam\jeem\dal\alef\waw\lam}}%
4849 %%
4850 \def\stctitle{\R{\alef\lam\fa\ha\ra\seen}}%
4851 \def\slftitle{\R{\qaf\alef\yahamza\meem\T\space \alef\lam\alefhamza\sheen\kaf\alef\lam}}%
4852 \def\slttitle{\R{\qaf\alef\yahamza\meem\T\space \alef\lam\jeem\dal\alef\waw\lam}}%
4853 %%
4854 </arabi>

```

13.10 “Arabic” language: arabic.mld

“Arabic” is a synonym for “arab”, so we just load arab.mld (see section 13.7 on page 418):

```

4855 (*arabic)
4856 \ProvidesFile{arabic.mld}[2005/02/10]
4857 \mtcselectlanguage{arab}%
4858 </arabic>

```

13.11 “Armenian” language: armenian.mld

The titles for the “armenian” language are taken from the ArmTeX package [90], which should be used, with the associated fonts.

```

4859 (*armenian)
4860 \ProvidesFile{armenian.mld}[1999/06/28]
4861 %% Armenian titles for minitoc.sty
4862 %% from ArmTeX
4863 %% Sergei Dachian (Serguei.Dachian@univ-lemans.fr)
4864 \def\ptctitle{Bovandakuthyun}%
4865 \def\plftitle{Patkerner i cank}%
4866 \def\plttitle{Aghyusakner i cank}%
4867 %%
4868 \def\mtctitle{Bovandakuthyun}%
4869 \def\mlftitle{Patkerner i cank}%

```

```
4870 \def\mltttitle{Aghyusakneri cank}%
4871 %%
4872 \def\stctitle{Bovandakuthyun}%
4873 \def\slfttitle{Patkerner i cank}%
4874 \def\sltttitle{Aghyusakneri cank}%
4875 </armenian>
```

13.12 “Australian” language: `australian.mld`

The “australian” language is just like “english”, so we just load `english.mld` (see section 13.42 on page 435):

```
4876 <*australian>
4877 \ProvidesFile{australian.mld}[2006/01/11]
4878 \mtcselectlanguage{english}%
4879 </australian>
```

13.13 “Austrian” language: `austrian.mld`

For the mini-table titles, the “austrian” language is like the “german” language, so we load `german.mld` (see section 13.64 on page 446):

```
4880 <*austrian>
4881 \ProvidesFile{austrian.mld}[2004/12/14]
4882 \mtcselectlanguage{german}%
4883 </austrian>
```

13.14 “Bahasa” language: `bahasa.mld`

The “bahasa” language is just like “bahasai”, so we just load `bahasai.mld` (see section 13.15 on the next page):

```
4884 <*bahasa>
4885 \ProvidesFile{bahasa.mld}[2006/01/11]
4886 \mtcselectlanguage{bahasai}%
4887 </bahasa>
```

13.15 “Bahasai” language: bahasai.mld

The titles of the mini-tables for the “bahasai” language⁹ (bahasa indonesia / bahasa meyalu) are taken from the file `bahasa.dtx` in the `babel` package [30, 31, 56]. Specific fonts are needed. See also section 13.16. The word “bahasa” means “language” in bahasa. For other names for this language, see sections 13.14 on the page before, 13.84 on page 458, and 13.85 on page 458.

```

4888 (*bahasai)
4889 \ProvidesFile{bahasai.mld}[2006/01/13]
4890 %% Bahasa Indonesia titles for minitoc.sty
4891 %% From bahasa.dtx in the babel package
4892 %% Knappen, Jörg and Mart, Terry
4893 \def\ptctitle{Daftar Isi}%
4894 \def\plftitle{Daftar Gambar}%
4895 \def\plttitle{Daftar Tabel}%
4896 %%
4897 \def\mtctitle{Daftar Isi}%
4898 \def\mlftitle{Daftar Gambar}%
4899 \def\mlttitle{Daftar Tabel}%
4900 %%
4901 \def\stctitle{Daftar Isi}%
4902 \def\slftitle{Daftar Gambar}%
4903 \def\slttitle{Daftar Tabel}%
4904 \end{bahasai}

```

13.16 “Bahasam” language: bahasam.mld

The titles of the mini-tables for the “bahasam” language (Bahasa Malaysia)¹⁰ are taken from the file `bahasam.dtx` in the `babel` package [30, 31, 57]. Specific fonts are needed. See also section 13.14 on the preceding page. For other names for this language, see sections 13.107 on page 469 and 13.113 on page 472.

```

4905 (*bahasam)
4906 \ProvidesFile{bahasam.mld}[2006/01/13]
4907 %% Bahasa Malaysia titles for minitoc.sty
4908 %% From bahasam.dtx in the babel package
4909 %% Knappen, Jörg and Mart, Terry
4910 \def\ptctitle{Kandungan}%
4911 \def\plftitle{Senarai Gambar}%
4912 \def\plttitle{Senarai Jadual}%
4913 %%
4914 \def\mtctitle{Kandungan}%

```

⁹ Bahasa is spoken in Indonesia and Malaysia, with different pronunciations and titles but the same writing. Bahasai is the Indonesian variant.

¹⁰ Spoken in Indonesia and Malaysia, with different pronunciations and titles but the same writing. Bahasam is the Malaysian variant.

```

4915 \def\mlftitle{Senarai Gambar}%
4916 \def\mltttitle{Senarai Jadual}%
4917 %%
4918 \def\stctitle{Kandungan}%
4919 \def\slftitle{Senarai Gambar}%
4920 \def\slttitle{Senarai Jadual}%
4921 </bahasam>

```

13.17 “Bangla” language: `bangla.mld`

The titles for the “bangla” language ¹¹ are taken from the `BangTeX` package [169]; they need specific fonts.

```

4922 <{*bangla}
4923 \ProvidesFile{bangla.mld}[2006/03/31]
4924 %% Bangla titles for minitoc.sty
4925 %% from BangTeX
4926 %% Needs specific fonts
4927 \def\ptctitle{suu\*c*ipotRo}%
4928 \def\plftitle{cho\*b*ir ta\*l*ika}%
4929 \def\plttitle{cho\*k*er ta\*l*ika}%
4930 %%
4931 \def\mtctitle{suu\*c*i}%
4932 \def\mlftitle{cho\*b*ir ta\*l*ika}%
4933 \def\mltttitle{cho\*k*er ta\*l*ika}%
4934 %%
4935 \def\stctitle{suu\*c*i}%
4936 \def\slftitle{cho\*b*ir ta\*l*ika}%
4937 \def\slttitle{cho\*k*er ta\*l*ika}%
4938 </bangla>

```

13.18 “Basque” language: `basque.mld`

The titles for the “basque” language ¹² are taken from the `basque.dtx` file in the `babel` package [30, 31, 37]. It seems that 8 bits fonts are preferable.

```

4939 <{*basque}
4940 \ProvidesFile{basque.mld}[2006/01/13]
4941 %% Basque titles for minitoc.sty
4942 %% from basque.dtx (babel)
4943 %% Aguirregabiria, Juan M. and Sanchez, Julio
4944 %% Needs special fonts
4945 \def\ptctitle{Gaien Aurkibidea}%

```

¹¹ Spoken in Bangladesh and some parts of India.

¹² Spoken in the Basque country, in the north of Spain and south-west of France.

```

4946 \def\plftitle{Irudien Zerrenda}%
4947 \def\pltttitle{Taulen Zerrenda}%
4948 %%
4949 \def\mtctitle{Gaien Aurkibidea}%
4950 \def\mlftitle{Irudien Zerrenda}%
4951 \def\mltttitle{Taulen Zerrenda}%
4952 %%
4953 \def\stctitle{Gaien Aurkibidea}%
4954 \def\slftitle{Irudien Zerrenda}%
4955 \def\slttitle{Taulen Zerrenda}%
4956 </basque>

```

13.19 “Bicig” language: bicig.mld

The titles for the “bicig” language¹³ are taken from the MonT_EX package [87, 89]. This language requires specific fonts. See also sections 13.114 on page 472, 13.20, and 13.21 on the following page.

```

4957 <{*bicig}
4958 \ProvidesFile{bicig.mld}[1999/03/16]
4959 %% Mongol (Bicig) titles for minitoc.sty
4960 %% needs mongol fonts
4961 \def\ptctitle{\bcg{Garcig}}%
4962 \def\plftitle{\bcg{zuraG-un zigsaaalt}}%
4963 \def\pltttitle{\bcg{k"usn"agti"in jagsaaalt}}%
4964 %%
4965 \def\mtctitle{\bcg{Garcig}}%
4966 \def\mlftitle{\bcg{zuraG-un zigsaaalt}}%
4967 \def\mltttitle{\bcg{k"usn"agti"in jagsaaalt}}%
4968 %%
4969 \def\stctitle{\bcg{Garcig}}%
4970 \def\slftitle{\bcg{zuraG-un zigsaaalt}}%
4971 \def\slttitle{\bcg{k"usn"agti"in jagsaaalt}}%
4972 </bicig>

```

13.20 “Bicig2” language: bicig2.mld

The titles for the “bicig2” language¹⁴ are taken from the MonT_EX package [87, 89]. This language requires specific fonts. See also sections 13.114 on page 472, 13.19, and 13.21 on the following page.

¹³Bicig is a written form of the mongolian language. It is also known as Uighur. See also section 13.157 on page 495.

¹⁴Bicig, or uighur, is a written form of the mongolian language, bicig2 is a variant. See also section 13.158 on page 495.


```

4973 <*bicig2>
4974 \ProvidesFile{bicig2.mld}[2005/11/16]
4975 %% Mongol (Bicig2) titles for minitoc.sty
4976 %% needs mongol fonts
4977 \def\ptctitle{garcag}%
4978 \def\plftitle{zirug-un zigsagalda}%
4979 \def\pltttitle{kuisunukdu-yin zigsagalda}%
4980 %%
4981 \def\mtctitle{garcag}%
4982 \def\mlftitle{zirug-un zigsagalda}%
4983 \def\mltttitle{kuisunukdu-yin zigsagalda}%
4984 %%
4985 \def\stctitle{garcag}%
4986 \def\slftitle{zirug-un zigsagalda}%
4987 \def\sltttitle{kuisunukdu-yin zigsagalda}%
4988 </bicig2>

```

13.21 “Bicig3” language: bicig3.mld

The titles for the “bicig3” language¹⁵ are taken from the MonTeX package [87, 89]. This language requires specific fonts. See also sections 13.114 on page 472, 13.19 on the page before and 13.20 on the preceding page.

```

4989 <*bicig3>
4990 \ProvidesFile{bicig3.mld}[2006/03/31]
4991 %% Mongol (Bicig3) titles for minitoc.sty
4992 %% needs mongol fonts
4993 \def\ptctitle{aguulag=a}%
4994 \def\plftitle{zirug-un zigsagalda}%
4995 \def\pltttitle{kuisunukdu-yin zigsagalda}%
4996 %%
4997 \def\mtctitle{aguulag=a}%
4998 \def\mlftitle{zirug-un zigsagalda}%
4999 \def\mltttitle{kuisunukdu-yin zigsagalda}%
5000 %%
5001 \def\stctitle{aguulag=a}%
5002 \def\slftitle{zirug-un zigsagalda}%
5003 \def\sltttitle{kuisunukdu-yin zigsagalda}%
5004 </bicig3>

```

¹⁵ Bicig, or uighur, is a written form of the mongolian language, bicig3 is a variant. See also section 13.159 on page 496.

13.22 “Bithe” language: `bithe.mld`

The titles for the “bithe” language¹⁶ are taken from the MonTeX package [87, 89]. This language requires specific fonts. See also sections 13.114 on page 472 and 13.112 on page 472.

The Manju writing, or *bithe* system is a close relative of the Mongolian system; the basic letter shapes are the same. Yet for Manju, a set of diacritics (*dots and circles*) was designed to the effect that all the ambiguities of Mongolian are eliminated.

```

5005 <*bithe>
5006 \ProvidesFile{bithe.mld}[2005/11/16]
5007 %% Manju (bithe) titles for minitoc.sty
5008 %% needs mongol fonts
5009 \def\ptctitle{garcag}%
5010 \def\plftitle{zirug-un? afaha}%
5011 \def\pltttitle{kuisunukdu-yin? afaha}%
5012 %%
5013 \def\mtctitle{garcag}%
5014 \def\mlftitle{zirug-un? afaha}%
5015 \def\mltttitle{kuisunukdu-yin? afaha}%
5016 %%
5017 \def\stctitle{garcag}%
5018 \def\slftitle{zirug-un? afaha}%
5019 \def\sltttitle{kuisunukdu-yin? afaha}%
5020 </bithe>

```

13.23 “Brazil” language: `brazil.mld`

The titles for the “brazil” language¹⁷ are taken from the `portuges.dtx` file in the `babel` package [30, 31, 67]:

```

5021 <*brazil>
5022 \ProvidesFile{brazil.mld}[2006/01/13]
5023 %% Portugues (brazil) titles for minitoc.sty
5024 %% from portuges.dtx (babel)
5025 %% Ramalhet, Jose Pedro and "de Lima", Arnaldo Viegas
5026 \def\ptctitle{Sum\'ario}%
5027 \def\plftitle{Lista de Figuras}%
5028 \def\pltttitle{Lista de Tabelas}%
5029 %%
5030 \def\mtctitle{Sum\'ario}%
5031 \def\mlftitle{Lista de Figuras}%
5032 \def\mltttitle{Lista de Tabelas}%

```

¹⁶Bithe is a written form of the manju variant of the mongolian language.

¹⁷It is the portuguese dialect spoken in Brazil. These titles are *different* in Brazil and in Portugal. Arnaldo VIEGAS DE LIMA contributed brasilian translations. See section 13.128 on page 479.

```

5033 %%
5034 \def\stctitle{Sum\ 'ario}%
5035 \def\slftitle{Lista de Figuras}%
5036 \def\sltttitle{Lista de Tabelas}%
5037 </brazil>

```

13.24 “Brazilian” language: brazilian.mld

The “brazilian” language is just like “brazil”, so we just load `brazil.mld` (see section 13.23 on the preceding page):

```

5038 < *brazilian>
5039 \ProvidesFile{brazilian.mld}[2005/07/11]
5040 \mtcselectlanguage{brazil}%
5041 </brazilian>

```

13.25 “Breton” language: breton.mld

The titles for the “breton” language¹⁸ are taken from `breton.dtx` file in the `babel` package [30, 31, 68]:

```

5042 < *breton>
5043 \ProvidesFile{breton.mld}[2006/01/13]
5044 %% Breton titles for minitoc.sty
5045 %% from breton.dtx (babel)
5046 %% Rolland, Christian
5047 \def\ptctitle{Taolenn}%
5048 \def\plftitle{Listenn ar Figurenno\ 'u}%
5049 \def\pltttitle{Listenn an taolenno\ 'u}%
5050 %%
5051 \def\mtctitle{Taolenn}%
5052 \def\mlftitle{Listenn ar Figurenno\ 'u}%
5053 \def\mltttitle{Listenn an taolenno\ 'u}%
5054 %%
5055 \def\stctitle{Taolenn}%
5056 \def\slftitle{Listenn ar Figurenno\ 'u}%
5057 \def\sltttitle{Listenn an taolenno\ 'u}%
5058 </breton>

```

¹⁸Spoken as a local celtic dialect in french Brittany.

13.26 “British” language: `british.mld`

The “british” language is just like “english”, so we just load `english.mld` (see section 13.42 on page 435):

```
5059 <{*british}
5060 \ProvidesFile{british.mld}[2005/07/11]
5061 \mtcselectlanguage{english}%
5062 </{*british}
```

13.27 “Bulgarian” language: `bulgarian.mld`

The titles for the “bulgarian” language are taken from the `bulgarian.dtx` file in the `babel` package [30, 31, 42]; they require specific cyrillic fonts. See also section 13.28 on the following page.

```
5063 <{*bulgarian}
5064 \ProvidesFile{bulgarian.mld}[2006/01/13]
5065 %% Bulgarian titles for minitoc.sty
5066 %% from bulgarian.dtx (babel)
5067 %% Boshnakov, Georgi
5068 %% Needs special fonts
5069 \def\ptctitle{%
5070   {\cyr\CYRS\cyrhrdsn\cyrd\cyrhrdsn\cyrr\cyrzh\cyra\cyrn\cyri\cyre}}%
5071 \def\plftitle{%
5072   {\cyr\CYRS\cyrp\cyri\cyr\cyrhrdsn\cyrk\ %
5073     \cyrn\cyra\ \cyrf\cyri\cyrg\cyru\cyrr\cyri\cyrt\cyre}}%
5074 \def\plttitle{%
5075   {\cyr\CYRS\cyrp\cyri\cyr\cyrhrdsn\cyrk\ %
5076     \cyrn\cyra\ \cyrt\cyra\cyrb\cyrl\cyri\cyrc\cyri\cyrt\cyre}}%
5077 %%
5078 \def\mtctitle{%
5079   {\cyr\CYRS\cyrhrdsn\cyrd\cyrhrdsn\cyrr\cyrzh\cyra\cyrn\cyri\cyre}}%
5080 \def\mlftitle{%
5081   {\cyr\CYRS\cyrp\cyri\cyr\cyrhrdsn\cyrk\ %
5082     \cyrn\cyra\ \cyrf\cyri\cyrg\cyru\cyrr\cyri\cyrt\cyre}}%
5083 \def\mlttitle{%
5084   {\cyr\CYRS\cyrp\cyri\cyr\cyrhrdsn\cyrk\ %
5085     \cyrn\cyra\ \cyrt\cyra\cyrb\cyrl\cyri\cyrc\cyri\cyrt\cyre}}%
5086 %%
5087 \def\stctitle{%
5088   {\cyr\CYRS\cyrhrdsn\cyrd\cyrhrdsn\cyrr\cyrzh\cyra\cyrn\cyri\cyre}}%
5089 \def\slftitle{%
5090   {\cyr\CYRS\cyrp\cyri\cyr\cyrhrdsn\cyrk\ %
5091     \cyrn\cyra\ \cyrf\cyri\cyrg\cyru\cyrr\cyri\cyrt\cyre}}%
5092 \def\slttitle{%
5093   {\cyr\CYRS\cyrp\cyri\cyr\cyrhrdsn\cyrk\ %
5094     \cyrn\cyra\ \cyrt\cyra\cyrb\cyrl\cyri\cyrc\cyri\cyrt\cyre}}%
```

5095 </bulgarian>

13.28 “Bulgarianb” language: bulgarianb.mld

The titles for the “bulgarianb” (upper bulgarian) language are taken from the `russianb.dtx` file of the `babel` package [30, 31, 58, 136]; they require specific cyrillic fonts. See also section 13.27 on the page before.

```

5096 <*bulgarianb>
5097 \ProvidesFile{bulgarianb.mld}[2006/03/06]
5098 %% Upper bulgarian titles for minitoc.sty
5099 %% From russianb.dtx
5100 %% Needs cyrillic fonts for upper bulgarian
5101 \def\ptctitle{%
5102   {\cyr\CYRS\cyrhrdsn\cyrd\cyrhrdsn\cyrr\cyrzh\cyra\cyrn\cyri\cyre}}%
5103 \def\plftitle{% Figuri
5104   {\cyr \CYRF\cyri\cyrg\cyru\cyrr\cyri}}%
5105 \def\pltttitle{% Tablici
5106   {\cyr \CYRT\cyra\cyrb\cyrl\cyri\cyrc\cyri}}%
5107 %%
5108 \def\mtctitle{% Sydyrzhanie
5109   {\cyr\CYRS\cyrhrdsn\cyrd\cyrhrdsn\cyrr\cyrzh\cyra\cyrn\cyri\cyre}}%
5110 \def\mlftitle{% Figurite
5111   {\cyr \CYRF\cyri\cyrg\cyru\cyrr\cyri}}%
5112 \def\mltttitle{% Tablici
5113   {\cyr \CYRT\cyra\cyrb\cyrl\cyri\cyrc\cyri}}%
5114 %%
5115 \def\stctitle{% Sydyrzhanie
5116   {\cyr\CYRS\cyrhrdsn\cyrd\cyrhrdsn\cyrr\cyrzh\cyra\cyrn\cyri\cyre}}%
5117 \def\slftitle{% Figuri
5118   {\cyr \CYRF\cyri\cyrg\cyru\cyrr\cyri}}%
5119 \def\slttitle{% Tablici
5120   {\cyr \CYRT\cyra\cyrb\cyrl\cyri\cyrc\cyri}}%
5121 </bulgarianb>

```

13.29 “Buryat” language: buryat.mld

The titles for the “buryat” language¹⁹ are taken from the `MonTeX` package [87, 89]. This language requires specific fonts. See also section 13.114 on page 472.

```

5122 <*buryat>
5123 \ProvidesFile{buryat.mld}[1999/03/16]
5124 %% Buryat titles for minitoc.sty
5125 %% Needs special fonts

```

¹⁹Spoken in some regions of Mongolia, in the Buryat republic, near Lake Baikal.

```

5126 \def\ptctitle{{\mnr Gar{\sh}ag}}%
5127 \def\plftitle{{\mnr Zuraga"i jagsaalt}}%
5128 \def\plttitle{{\mnr X"usn"ag"at"a"i jagsaalt}}%
5129 %%
5130 \def\mtctitle{{\mnr Gar{\sh}ag}}%
5131 \def\mlftitle{{\mnr Zuraga"i jagsaalt}}%
5132 \def\mlttitle{{\mnr X"usn"ag"at"a"i jagsaalt}}%
5133 %%
5134 \def\stctitle{{\mnr Gar{\sh}ag}}%
5135 \def\slftitle{{\mnr Zuraga"i jagsaalt}}%
5136 \def\slttitle{{\mnr X"usn"ag"at"a"i jagsaalt}}%
5137 </buryat>

```

13.30 “Buryat2” language: buryat2.mld

The titles for the “buryat2” language (a variant for the buryat language, see section 13.29 on the page before) are taken from the MonTeX package [87, 89]. This language requires specific fonts. See also section 13.114 on page 472.

```

5138 <{*buryat2}
5139 \ProvidesFile{buryat2.mld}[1999/03/16]
5140 %% Buryat2 titles for minitoc.sty
5141 %% Needs special fonts
5142 \def\ptctitle{{\mnr Aguulga}}%
5143 \def\plftitle{{\mnr Zuraga"i jagsaalt}}%
5144 \def\plttitle{{\mnr X"usn"ag"at"a"i jagsaalt}}%
5145 %%
5146 \def\mtctitle{{\mnr Aguulga}}%
5147 \def\mlftitle{{\mnr Zuraga"i jagsaalt}}%
5148 \def\mlttitle{{\mnr X"usn"ag"at"a"i jagsaalt}}%
5149 %%
5150 \def\stctitle{{\mnr Aguulga}}%
5151 \def\slftitle{{\mnr Zuraga"i jagsaalt}}%
5152 \def\slttitle{{\mnr X"usn"ag"at"a"i jagsaalt}}%
5153 </buryat2>

```

13.31 “Canadian” language: canadian.mld



The “canadian” language (note the final “ian”) is just the *english* language spoken in Canada. We just load the file `english.mld` (see section 13.42 on page 435):

```

5154 <{*canadian}
5155 \ProvidesFile{canadian.mld}[2004/12/14]
5156 \mtcselectlanguage{english}%
5157 </canadian>

```

13.32 “Canadien” language: `canadien.mld`



The “canadien” language (note the final “ien”) is just the *french* language spoken in Canada. We just load the file `french.mld` (see section 13.57 on page 443):

```
5158 <*canadien>
5159 \ProvidesFile{canadien.mld}[2004/12/14]
5160 \mtcselectlanguage{french}%
5161 </canadien>
```

13.33 “Castillan” language: `castillan.mld`

The “castillan” language is more known as “spanish”, but is spoken mainly in Castile, a part of central Spain. We just load the `spanish.mld` file (see section 13.149 on page 491):

```
5162 <*castillan>
5163 \ProvidesFile{castillan.mld}[2004/12/14]
5164 %% Castillan (spanish) titles for minitoc.sty
5165 \mtcselectlanguage{spanish}%
5166 </castillan>
```

13.34 “Castillian” language: `castillian.mld`

The “castillian” language is more known as “spanish”, but is spoken mainly in Castile, a part of central Spain. “Castillian” is the english name for “castillan”. We just load the `spanish.mld` file (see section 13.149 on page 491):

```
5167 <*castillian>
5168 \ProvidesFile{castillian.mld}[2005/07/01]
5169 %% Castillian (spanish) titles for minitoc.sty
5170 \mtcselectlanguage{spanish}%
5171 </castillian>
```

13.35 “Catalan” language: `catalan.mld`

The titles for the “catalan” language²⁰ are taken from the `catalan.dtx` file in the babel package [30, 31, 39]:

²⁰Spoken in Catalunya, the eastern part of Spain, around Barcelona.

```

5172 <*catalan>
5173 \ProvidesFile{catalan.mld}[2006/01/13]
5174 %% Catalan titles for minitoc.sty
5175 %% from catalan.dtx (babel)
5176 %% Badenes, Goncal
5177 \def\ptctitle{\`Index}%
5178 \def\plftitle{\`Index de figures}%
5179 \def\plttitle{\`Index de taules}%
5180 %%
5181 \def\mtctitle{\`Index}%
5182 \def\mlftitle{Figures}%
5183 \def\mlttitle{Taules}%
5184 %%
5185 \def\stctitle{\`Index}%
5186 \def\slftitle{Figures}%
5187 \def\slttitle{Taules}%
5188 </catalan>

```

13.36 “Chinese1” language: chinese1.mld and chinese1.mlo

There are several variants for the chinese language. The “chinese1” language uses titles taken from the Bg5.cap file in the CJK system [143, 144]. Special fonts are needed, of course. See also section 13.37.

The titles for the “chinese1” language contain characters that cannot be easily generated, hence we load chinese1.mlo.

```

5189 <*chinese1>
5190 \ProvidesFile{chinese1.mld}[2005/01/28]
5191 %% From the file file Bg5.cap of the CJK package
5192 %%   for using Asian logographs (Chinese/Japanese/Korean) with LaTeX2e
5193 %%   created by Werner Lemberg <wl@gnu.org>
5194 %% Version 4.5.2 (28-Mar-2003)
5195 %% Chinese captions: character set: Big 5, encoding: Big 5
5196 %%
5197 \mtcloadmlo{chinese1}%
5198 </chinese1>

```

13.37 “Chinese2” language: chinese2.mld and chinese2.mlo

The “chinese2” language uses titles taken from the Bg5.cpx file in the CJK system [143, 144]. Special fonts are needed, of course. See also section 13.36.

The titles for the “chinese2” language contain characters that cannot be easily generated, hence we load `chinese2.mlo`.

```

5199 < *chinese2 >
5200 \ProvidesFile{chinese2.mld}[2005/01/28]
5201 %% From the file Bg5.cpx of the CJK package
5202 %% for using Asian logographs (Chinese/Japanese/Korean) with LaTeX2e
5203 %% created by Werner Lemberg <wl@gnu.org>
5204 %%
5205 %% Version 4.5.2 (28-Mar-2003)
5206 %% Chinese captions: character set: Big 5, encoding: Big 5, preprocessed
5207 %%
5208 \mtcloadmlo{chinese2}%
5209 < /chinese2 >

```

13.38 “Croatian” language: `croatian.mld`

The titles for the “croatian” language are taken from the file `croatian.dtx` file in the `babel` package [30, 31, 64]:

```

5210 < *croatian >
5211 \ProvidesFile{croatian.mld}[2006/01/13]
5212 %% Croatian titles for minitoc.sty
5213 %% from croatian.dtx (babel)
5214 %% Pai\{c}, Alan
5215 \def\ptctitle{Sadr\v{z}aj}%
5216 \def\plftitle{Slike}%
5217 \def\plttitle{Tablice}%
5218 %%
5219 \def\mtctitle{Sadr\v{z}aj}%
5220 \def\mlftitle{Slike}%
5221 \def\mlttitle{Tablice}%
5222 %%
5223 \def\stctitle{Sadr\v{z}aj}%
5224 \def\slftitle{Slike}%
5225 \def\slttitle{Tablice}%
5226 < /croatian >

```

13.39 “Czech” language: `czech.mld`

The titles for the “czech” language are taken from the file `czech.dtx` in the `babel` package [30, 31, 61]:

```

5227 < *czech >
5228 \ProvidesFile{czech.mld}[2006/01/13]

```

```

5229 %% Czech titles for minitoc.sty
5230 %% from czech.dtx (babel)
5231 %% Lokajicek, Milos
5232 %% Warning: defines \w as a ring accent
5233 \def\w#1{\accent'27 #1}
5234 \def\ptctitle{Obsah}%
5235 \def\plftitle{Seznam obr\'azk\w{u}}%
5236 \def\plttitle{Seznam tabulek}%
5237 %%
5238 \def\mtctitle{Obsah}%
5239 \def\mlftitle{Seznam obr\'azk\w{u}}%
5240 \def\mlttitle{Seznam tabulek}%
5241 %%
5242 \def\stctitle{Obsah}%
5243 \def\slftitle{Seznam obr\'azk\w{u}}%
5244 \def\slttitle{Seznam tabulek}%
5245 </czech>

```

13.40 “Danish” language: danish.mld

The titles for the “danish” language are taken from the danish.dtx file in the babel package [30, 31, 59]:

```

5246 < *danish>
5247 \ProvidesFile{danish.mld}[2006/01/13]
5248 %% Danish titles for minitoc.sty
5249 %% from danish.dtx (babel)
5250 %% Larsen, Henning
5251 \def\ptctitle{Indholdsfortegnelse}%
5252 \def\plftitle{Figurer}%
5253 \def\plttitle{Tabeller}%
5254 %%
5255 \def\mtctitle{Indholdsfortegnelse}%
5256 \def\mlftitle{Figurer}%
5257 \def\mlttitle{Tabeller}%
5258 %%
5259 \def\stctitle{Indholdsfortegnelse}%
5260 \def\slftitle{Figurer}%
5261 \def\slttitle{Tabeller}%
5262 </danish>

```

13.41 “Dutch” language: dutch.mld

The titles for the “dutch” language are taken from the dutch.dtx file in the babel package [30, 31, 32]:

```

5263 <*dutch>
5264 \ProvidesFile{dutch.mld}[2006/01/13]
5265 % Dutch titles for minitoc.sty
5266 %% from dutch.dtx (babel)
5267 %% Braams, Johannes
5268 \def\ptctitle{Inhoudsopgave}%
5269 \def\plftitle{Lijst van figuren}%
5270 \def\plttitle{Lijst van tabellen}%
5271 %%
5272 \def\mtctitle{Inhoudsopgave}%
5273 \def\mlftitle{Lijst van figuren}%
5274 \def\mlttitle{Lijst van tabellen}%
5275 %%
5276 \def\stctitle{Inhoudsopgave}%
5277 \def\slftitle{Lijst van figuren}%
5278 \def\slttitle{Lijst van tabellen}%
5279 </dutch>

```

13.42 “English” language: english.mld



The titles for the “english” language are taken from the english.dtx file in the babel package [30, 31, 33]. *It is the default language.*

See also sections 13.6 on page 418, 13.12 on page 421, 13.26 on page 428, 13.31 on page 430, 13.116 on page 473, 13.160 on page 496, and 13.164 on page 498.

```

5280 <*english>
5281 \ProvidesFile{english.mld}[2006/01/13]
5282 %% English titles for minitoc.sty
5283 %% from english.dtx (babel)
5284 %% Braams, Johannes
5285 \def\ptctitle{Table of Contents}%
5286 \def\plftitle{List of Figures}%
5287 \def\plttitle{List of Tables}%
5288 %%
5289 \def\mtctitle{Contents}%
5290 \def\mlftitle{Figures}%
5291 \def\mlttitle{Tables}%
5292 %%
5293 \def\stctitle{Contents}%
5294 \def\slftitle{Figures}%
5295 \def\slttitle{Tables}%
5296 </english>

```

13.43 “English1” language: english1.mld

The titles for the “english1” language are taken from the english.dtx file in the babel package [30, 31, 33], with some adaptations for the part-level titles.

```

5297 (*english1)
5298 \ProvidesFile{english1.mld}[2006/03/30]
5299 %% English titles for minitoc.sty
5300 %% from english.dtx (babel) Braams, Johannes
5301 %% ptctitle, plftitle and pltttitle modified (JPFD)
5302 \def\ptctitle{\ifnum\value{part}=1\relax
5303   Table of Contents of the First Part\relax
5304   \else Table of Contents of Part~\Roman{part}\fi}%
5305 \def\plftitle{\ifnum\value{part}=1\relax
5306   List of Figures in the First Part\relax
5307   \else List of Figures in Part~\Roman{part}\fi}%
5308 \def\pltttitle{\ifnum\value{part}=1\relax
5309   List of Tables in the First Part\relax
5310   \else List of Tables in Part~\Roman{part}\fi}%
5311 %%
5312 \def\mtctitle{Contents}%
5313 \def\mlftitle{Figures}%
5314 \def\mltttitle{Tables}%
5315 %%
5316 \def\stctitle{Contents}%
5317 \def\slftitle{Figures}%
5318 \def\sltttitle{Tables}%
5319 </english1>

```

13.44 “English2” language: english2.mld

The titles for the “english2” language are taken from the english.dtx file in the babel package [30, 31, 33], with some adaptations for the part-level titles.

```

5320 (*english2)
5321 \ProvidesFile{english2.mld}[2006/03/30]
5322 %% English titles for minitoc.sty
5323 %% from english.dtx (babel) Braams, Johannes
5324 %% ptctitle, plftitle and pltttitle modified (JPFD)
5325 \def\mtcEnglishIIpart{\ifcase\value{part}%
5326 \or the First Part\or the Second Part\or
5327 the Third Part\or the Fourth Part\or the Fifth Part\or
5328 the Sixth Part\or the Seventh Part\or the Eighth Part\or
5329 the Ninth Part\or the Tenth Part\or the Eleventh Part\or
5330 the Twelfth Part\or the Thirteenth Part\or the Fourteenth Part\or
5331 the Fifteenth Part\or the Sixteenth Part\or the Seventeenth Part\or
5332 the Eighteenth Part\or the Nineteenth Part\or
5333 the Twentieth Part\else Part~\Roman{part}\fi}

```

```

5334 \def\ptctitle{Contents of \mtcEnglishIIpart}
5335 \def\plftitle{List of Figures in \mtcEnglishIIpart}
5336 \def\plttitle{List of Tables in \mtcEnglishIIpart}
5337 %%
5338 \def\mtctitle{Contents}%
5339 \def\mlftitle{Figures}%
5340 \def\mlttitle{Tables}%
5341 %%
5342 \def\stctitle{Contents}%
5343 \def\slftitle{Figures}%
5344 \def\slttitle{Tables}%
5345 </english2>

```

13.45 “Esperant” language: `esperant.mld`

The titles for the “esperant” (espéranto) language are taken from the `esperanto.dtx` file in the `babel` package [30, 31, 62]. The esperanto (artificial) language was created in 1887 by Lejzer Ludwig ZAMENOF²¹ (1859–1917).

```

5346 < *esperant>
5347 \ProvidesFile{esperant.mld}[2006/01/13]
5348 %% Esperanto titles for minitoc.sty
5349 %% from esperanto.dtx (babel)
5350 %% Marti, Ruiz-Altaba and Knappen. Jörg
5351 \def\ptctitle{Enhavo}%
5352 \def\plftitle{Listo de figuroj}%
5353 \def\plttitle{Listo de tabeloj}%
5354 %%
5355 \def\mtctitle{Enhavo}%
5356 \def\mlftitle{Listo de figuroj}%
5357 \def\mlttitle{Listo de tabeloj}%
5358 %%
5359 \def\stctitle{Enhavo}%
5360 \def\slftitle{Listo de figuroj}%
5361 \def\slttitle{Listo de tabeloj}%
5362 </esperant>

```

13.46 “Esperanto” language: `esperanto.mld`

The “esperanto” and “esperant” languages are synonyms, so we just load the `esperant.mld` file (see section 13.45):

```

5363 < *esperanto>
5364 \ProvidesFile{esperanto.mld}[2004/12/14]

```

²¹ See <http://uea.org/> and <http://www.esperanto-france.org/> for more information.

```
5365 \mtcselectlanguage{esperant}%
5366 </esperanto>
```

13.47 “Estonian” language: estonian.mld

The titles for the “estonian” language are taken from the `estonian.dtx` file in the `babel` package [30, 31, 69]:

```
5367 <(*estonian)
5368 \ProvidesFile{estonian.mld}[2006/01/13]
5369 %% Estonian titles for minitoc.sty
5370 %% from estonian.dtx (babel)
5371 %% Saar, Enn
5372 \def\ptctitle{Sisukord}%
5373 \def\plftitle{Joonised}%
5374 \def\plttitle{Tabelid}%
5375 %%
5376 \def\mtctitle{Sisukord}%
5377 \def\mlftitle{Joonised}%
5378 \def\mlttitle{Tabelid}%
5379 %%
5380 \def\stctitle{Sisukord}%
5381 \def\slftitle{Joonised}%
5382 \def\slttitle{Tabelid}%
5383 </estonian>
```

13.48 “Ethiopia” language: ethiopia.mld

The titles for the “ethiopia” language are taken from the `ethiop` package [23]. Specific fonts are needed. See also section 13.50 on the next page.

```
5384 <(*ethiopia)
5385 \ProvidesFile{ethiopia.mld}[1999/03/16]
5386 %% Ethiopian titles for minitoc.sty
5387 %% Needs special fonts
5388 \def\ptctitle{yezate}%
5389 \def\plftitle{%
5390   ya\eth@doaltchar{85}‘elo\eth@doaltchar{109}
5391   mAw\eth@doaltchar{187}}%
5392 \def\plttitle{%
5393   yasane\eth@doaltchar{176}ra\eth@doaltchar{149}
5394   mAw\eth@doaltchar{187}}%
5395 %%
5396 \def\mtctitle{yezate}%
5397 \def\mlftitle{%
5398   ya\eth@doaltchar{85}‘elo\eth@doaltchar{109}
```

```

5399      mAwe\eth@doaltchar{187}}%
5400 \def\mltttitle{%
5401      yasane\eth@doaltchar{176}ra\eth@doaltchar{149}
5402      mAwe\eth@doaltchar{187}}%
5403 %%
5404 \def\stctitle{yezate}%
5405 \def\slfttitle{%
5406      ya\eth@doaltchar{85}'elo\eth@doaltchar{109}
5407      mAwe\eth@doaltchar{187}}%
5408 \def\sltttitle{%
5409      yasane\eth@doaltchar{176}ra\eth@doaltchar{149}
5410      mAwe\eth@doaltchar{187}}%
5411 </ethiopia>

```

13.49 “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 13.48 on the preceding page). See also section 13.48 on the page before.

```

5412 <*ethiopian>
5413 \ProvidesFile{ethiopian.mld}[2004/12/14]
5414 %% Needs ethopian special fonts
5415 \mtcselectlanguage{ethiopia}%
5416 </ethiopian>

```

13.50 “Ethiopian2” language: `ethiopian2.mld`

The titles for the “ethiopian2” language (for Omega) are taken from the `ethiop` package [23]. Specific fonts are needed. See also section 13.48 on the preceding page.

```

5417 <*ethiopian2>
5418 \ProvidesFile{ethiopian2.mld}[2006/01/30]
5419 %% Ethiopian titles for minitoc.sty with Omega.
5420 %% Needs special fonts
5421 \def\ptctitle{^12ed^12d8^1275}%
5422 \def\plftitle{^12e8^1225^12d5^120e^127d
5423      ^121b^12cd^12b}%
5424 \def\pltttitle{^12e8^1230^1295^1230^1228^12e5
5425      ^121b^12cd^12b}%
5426 %%
5427 \def\mtctitle{^12ed^12d8^1275}%
5428 \def\mlftitle{^12e8^1225^12d5^120e^127d
5429      ^121b^12cd^12b}%
5430 \def\mltttitle{^12e8^1230^1295^1230^1228^12e5

```

```

5431  ^^^^121b^^^^12cd^^^^132b}%
5432 %%
5433 \def\stctitle{^^^^12ed^^^^12d8^^^^1275}%
5434 \def\slftitle{^^^^12e8^^^^1225^^^^12d5^^^^120e^^^^127d
5435  ^^^^121b^^^^12cd^^^^132b}%
5436 \def\sltttitle{^^^^12e8^^^^1230^^^^1295^^^^1320^^^^1228^^^^12e5
5437  ^^^^121b^^^^12cd^^^^132b}%
5438 </ethiopian2>

```

13.51 “Farsi1” language: farsi1.mld and farsi1.mlo

There are several variants for the farsi language, spoken in Iran. The “farsi1” language uses titles taken from the `farsi.sty` file in the FarsiTeX system²², by Mohammad GHODSI, Roozbeh POURNADER, Hassan ABOLHASSANI, and others. Special fonts are needed, of course. See also section 13.52.

The titles for the “farsi1” language contain characters that cannot be easily generated, hence we load `farsi1.mlo`.

```

5439 <(*farsi1)
5440 \ProvidesFile{farsi1.mld}[2005/09/13]
5441 %% From farsi.sty of the FarsiTeX project
5442 %% by Dr Mohammad Ghodsi,
5443 %% Roozbeh Pournader (roozbeh@sharif.edu),
5444 %% Hassan Abolhassani, and others.
5445 %% http://www.farsitex.org
5446 \mtcloadmlo{farsi1}%
5447 </farsi1>

```

13.52 “Farsi2” language: farsi2.mld and farsi2.mlo

There are several variants for the farsi language, spoken in Iran. The “farsi2” language uses titles taken from the `farsi.sty` file in the FarsiTeX system²², by Mohammad GHODSI, Roozbeh POURNADER, Hassan ABOLHASSANI, and others. Special fonts are needed, of course. See also section 13.51.

The titles for the “farsi2” language contain characters that cannot be easily generated, hence we load `farsi2.mlo`.

```

5448 <(*farsi2)
5449 \ProvidesFile{farsi2.mld}[2005/09/13]
5450 %% From farsi.sty of the FarsiTeX project

```

²²By Mohammad GHODSI (ghodsi@rose.ipm.ac.ir) and FarsiTeX Project Group. See the FarsiTeX site at <http://www.farsitex.org>


```

5451 %% by Dr Mohammad Ghodsi,
5452 %% Roozbeh Pournader (roozbeh@sharif.edu),
5453 %% Hassan Abolhassani, and others.
5454 %% http://www.farsitex.org
5455 \mtcloadmlo{farsi2}%
5456 </farsi2>

```

13.53 “Farsi3” language: farsi3.mld

There are several variants for the farsi language, spoken in Iran. The “farsi3” language uses titles taken from the farsi.ldf file in the Arabi system[116] by Youssef JABRI. Special fonts are needed, of course.

```

5457 <{*farsi3}>
5458 \ProvidesFile{farsi3.mld}[2006/07/27]
5459 %% From farsi.ldf of the Arabi system
5460 %% by Youssef Jabri.
5461 \def\ptctitle{\FR{\fa\ha\ra\seen\taa\space\meem\nun\dal\ra\jeem\alef\taa}}%
5462 \def\plftitle{\FR{\lam\ya\seen\taa\ \alef\sheen\kaf\alef\lam}}%
5463 \def\pltttitle{\FR{\lam\ya\seen\taa\ \jeem\dal\alef\waw\lam}}%
5464 %%
5465 \def\mtctitle{\FR{\fa\ha\ra\seen\taa\space\meem\nun\dal\ra\jeem\alef\taa}}%
5466 \def\mlftitle{\FR{\lam\ya\seen\taa\ \alef\sheen\kaf\alef\lam}}%
5467 \def\mltttitle{\FR{\lam\ya\seen\taa\ \jeem\dal\alef\waw\lam}}%
5468 %%
5469 \def\stctitle{\FR{\fa\ha\ra\seen\taa\space\meem\nun\dal\ra\jeem\alef\taa}}%
5470 \def\slftitle{\FR{\lam\ya\seen\taa\ \alef\sheen\kaf\alef\lam}}%
5471 \def\sltttitle{\FR{\lam\ya\seen\taa\ \jeem\dal\alef\waw\lam}}%
5472 </farsi3>

```

13.54 “Finnish” language: finnish.mld

The titles for the “finnish” language are taken from the finnish.dtx file in the babel package [30, 31, 54]. See also section 13.55 on the next page.

```

5473 <{*finnish}>
5474 \ProvidesFile{finnish.mld}[2006/03/20]
5475 %% Finnish titles for minitoc.sty
5476 %% from finnish.dtx (babel)
5477 %% Kanerva, Mikko and Reino, Keranen
5478 \def\ptctitle{Sis\alt{o}}%
5479 \def\plftitle{Kuvat}%
5480 \def\pltttitle{Taulukot}%
5481 %%
5482 \def\mtctitle{Sis\alt{o}}%

```

```

5483 \def\mlftitle{Kuvat}%
5484 \def\mltttitle{Taulukot}%
5485 %%
5486 \def\stctitle{Sis\alt{o}}%
5487 \def\slftitle{Kuvat}%
5488 \def\slttitle{Taulukot}%
5489 </finnish>

```

13.55 “Finnish2” language: `finnish2.mld`

The titles for the “finnish2” language are taken from a variant proposed by the `finnish.dtx` in the `babel` package [30, 31, 54]. See also section 13.54 on the preceding page.

```

5490 <{*finnish2>
5491 \ProvidesFile{finnish2.mld}[2006/01/13]
5492 %% Finnish titles for minitoc.sty (variant)
5493 %% from finnish.dtx (babel)
5494 %% Kanerva, Mikko and Reino, Keranen
5495 \def\ptctitle{Sis"allys}%
5496 \def\plftitle{Kuvat}%
5497 \def\plttitle{Taulukot}%
5498 %%
5499 \def\mtctitle{Sis"allys}%
5500 \def\mlftitle{Kuvat}%
5501 \def\mltttitle{Taulukot}%
5502 %%
5503 \def\stctitle{Sis"allys}%
5504 \def\slftitle{Kuvat}%
5505 \def\slttitle{Taulukot}%
5506 </finnish2>

```

13.56 “Français” language: `francais.mld`

The “français”²³ language is a synonym for the “french” language, so we load the file `french.mld` (see section 13.57 on the next page):

```

5507 <{*francais>
5508 \ProvidesFile{francais.mld}[2004/12/14]
5509 \mtcselectlanguage{french}%
5510 </francais>

```

²³ The right spelling is “français”, but we do not dare using a cedilla in a file name.

13.57 “French” language: french.mld

The titles for the “french” language are taken from the frenchb.dtx file in the babel package [30, 31, 49]. See also sections 13.1 on page 416, 13.2 on page 417, 13.32 on page 431, 13.56 on the page before, 13.60 on page 445, 13.61 on page 445, and 13.62 on page 445.

```

5511 (*french)
5512 \ProvidesFile{french.mld}[2006/03/21]
5513 %% French titles for minitoc.sty
5514 %% from frenchb.dtx (babel)
5515 %% Flipo, Daniel
5516 \def\ptctitle{Table des mati\`eres}%
5517 \def\plftitle{Liste des figures}%
5518 \def\plttitle{Liste des tableaux}%
5519 %%
5520 \def\mtctitle{Sommaire}%
5521 \def\mlftitle{Figures}%
5522 \def\mlttitle{Tableaux}%
5523 %%
5524 \def\stctitle{Sommaire}%
5525 \def\slftitle{Figures}%
5526 \def\slttitle{Tableaux}%
5527 \end{french}

```

13.58 “French1” language: french1.mld

The titles for the “french1” language are taken from the frenchb.dtx file in the babel package [30, 31, 49], with some adaptations for the part-level titles.

```

5528 (*french1)
5529 \ProvidesFile{french1.mld}[2006/03/29]
5530 %% French titles for minitoc.sty
5531 %% from frenchb.dtx (babel) Flipo, Daniel
5532 %% ptctitle, plftitle and plttitle modified (JPFD)
5533 \def\ptctitle{\ifnum\value{part}=1\relax
5534   Sommaire de la premi\`ere partie\relax
5535   \else Sommaire de la partie~\Roman{part}\fi}%
5536 \def\plftitle{\ifnum\value{part}=1\relax
5537   Liste des figures de la premi\`ere partie\relax
5538   \else Liste des figures de la partie~\Roman{part}\fi}%
5539 \def\plttitle{\ifnum\value{part}=1\relax
5540   Liste des tableaux de la premi\`ere partie\relax
5541   \else Liste des tableaux de la partie~\Roman{part}\fi}%
5542 %%
5543 \def\mtctitle{Sommaire}%
5544 \def\mlftitle{Figures}%
5545 \def\mlttitle{Tableaux}%

```

```

5546 %%
5547 \def\stctitle{Sommaire}%
5548 \def\slftitle{Figures}%
5549 \def\sltttitle{Tableaux}%
5550 </french1>

```

13.59 “French2” language: french2.mld

The titles for the “french2” language are taken from the frenchb.dtx file in the babel package [30, 31, 49], with some adaptations for the part-level titles. See also section 9.5.6 on page 240, for the subtle distinction between “deuxième” and “seconde”. See the second.tex example file in section 4.5 on page 106.

```

5551 <{*french2>
5552 \ProvidesFile{french2.mld}[2006/07/07]
5553 %% French titles for minitoc.sty
5554 %% from frenchb.dtx (babel) Flipo, Daniel
5555 %% ptctitle, plftitle and pltttitle modified (JPFD)
5556 \def\mtcFrenchIIpart{\ifcase\value{part}%
5557 \or premi\`ere partie\or
5558 {\ifmtcsecondpart seconde\else deuxi\`eme\fi} partie\or
5559 troisi\`eme partie\or quatri\`eme partie\or cinqui\`eme partie\or
5560 sixi\`eme partie\or septi\`eme partie\or huiti\`eme partie\or
5561 neuvi\`eme partie\or dixi\`eme partie\or onzi\`eme partie\or
5562 douzi\`eme partie\or treizi\`eme partie\or quatorzi\`eme partie\or
5563 quinzi\`eme partie\or seizi\`eme partie\or dix-septi\`eme partie\or
5564 dix-huiti\`eme partie\or dix-neuvi\`eme partie\or
5565 vingti\`eme partie\else partie~\Roman{part}\fi}
5566 \def\ptctitle{\ifnum\value{part}<1\relax
5567 Sommaire \else
5568 Sommaire de la \mtcFrenchIIpart\fi}
5569 \def\plftitle{\ifnum\value{part}<1\relax
5570 Liste des figures\else
5571 Liste des figures de la \mtcFrenchIIpart\fi}
5572 \def\pltttitle{\ifnum\value{part}<1\relax
5573 Liste des tableaux\else
5574 Liste des tableaux de la \mtcFrenchIIpart}
5575 %%
5576 \def\mtctitle{Sommaire}%
5577 \def\mlftitle{Figures}%
5578 \def\mltttitle{Tableaux}%
5579 %%
5580 \def\stctitle{Sommaire}%
5581 \def\slftitle{Figures}%
5582 \def\sltttitle{Tableaux}%
5583 </french2>

```

13.60 “Frenchb” language: frenchb.mld

The “frenchb” language is a synonym for the “french” language, so we load the french.mld file. See section 13.57 on page 443.

```
5584 \*frenchb>
5585 \ProvidesFile{frenchb.mld}[2003/02/11]
5586 \mtcselectlanguage{french}%
5587 </frenchb>
```

13.61 “Frenchle” language: frenchle.mld

The “frenchle” language is a synonym for the “french” language, so we load the french.mld file. See section 13.57 on page 443.

```
5588 \*frenchle>
5589 \ProvidesFile{frenchle.mld}[2003/02/20]
5590 \mtcselectlanguage{french}%
5591 </frenchle>
```

13.62 “Frenchpro” language: frenchpro.mld

The “frenchpro” language is a synonym for the “french” language, so we load the french.mld file. See section 13.57 on page 443.

```
5592 \*frenchpro>
5593 \ProvidesFile{frenchpro.mld}[2003/02/20]
5594 \mtcselectlanguage{french}%
5595 </frenchpro>
```

13.63 “Galician” language: galician.mld

The titles for the “galician” language²⁴ are taken from the galician.dtx file in the babel package [30, 31, 45]:

```
5596 \*galician>
5597 \ProvidesFile{galician.mld}[2006/01/13]
5598 %% Galician titles for minitoc.sty
```

²⁴Spoken in Galice, in the north-west part of Spain, around Santiago de Compostela.

```

5599%% from galician.dtx (babel)
5600%% Carriba, Manuel
5601\def\ptctitle{\'Indice Xeral}%
5602\def\plftitle{\'Indice de Figuras}%
5603\def\plttitle{\'Indice de T\'aboas}%
5604%%
5605\def\mtctitle{\'Indice Xeral}%
5606\def\mlftitle{\'Indice de Figuras}%
5607\def\mlttitle{\'Indice de T\'aboas}%
5608%%
5609\def\stctitle{\'Indice Xeral}%
5610\def\slftitle{\'Indice de Figuras}%
5611\def\slttitle{\'Indice de T\'aboas}%
5612</galician>

```

13.64 “German” language: `german.mld`

The titles for the “german” language are taken from the babel package [30, 31]. See also the section 13.13 on page 421.

```

5613<*german>
5614\ProvidesFile{german.mld}[1999/03/16]
5615%% German titles for minitoc.sty
5616\def\ptctitle{Inhaltsangabe}%
5617\def\plftitle{Figuren}%
5618\def\plttitle{Tabellen}%
5619%%
5620\def\mtctitle{Inhaltsangabe}%
5621\def\mlftitle{Figuren}%
5622\def\mlttitle{Tabellen}%
5623%%
5624\def\stctitle{Inhaltsangabe}%
5625\def\slftitle{Figuren}%
5626\def\slttitle{Tabellen}%
5627</german>

```

13.65 “Germanb” language: `germanb.mld`

The “germanb” language is a variant for the “german” language. The titles are taken from the `germanb.dtx` file in the babel package [30, 31, 65]:

```

5628<*germanb>
5629\ProvidesFile{germanb.mld}[2006/01/13]
5630%% German titles (variant) for minitoc.sty
5631%% from germanb.dtx (babel)
5632%% Braams, Johannes and Raichle, Bernd

```

```

5633 \def\ptctitle{Inhaltsverzeichnis}%
5634 \def\plftitle{Abbildungsverzeichnis}%
5635 \def\plttitle{Tabellenverzeichnis}%
5636 %%
5637 \def\mtctitle{Inhaltsverzeichnis}%
5638 \def\mlftitle{Abbildungsverzeichnis}%
5639 \def\mlttitle{Tabellenverzeichnis}%
5640 %%
5641 \def\stctitle{Inhalt}%
5642 \def\slftitle{Abbildungen}%
5643 \def\slttitle{Tabellen}%
5644 \</germanb>

```

13.66 “Germanb2” language: `germanb2.mld`

The “germanb2” language is a variant for the “german” language, with short titles. See also section 13.65 on the page before. The titles are taken from the babel package [30, 31]:

```

5645 \<germanb2>
5646 \ProvidesFile{germanb2.mld}[2005/09/27]
5647 %% German titles (variant) for minitoc.sty
5648 \def\ptctitle{Inhalt}%
5649 \def\plftitle{Abbildungen}%
5650 \def\plttitle{Tabellen}%
5651 %%
5652 \def\mtctitle{Inhalt}%
5653 \def\mlftitle{Abbildungen}%
5654 \def\mlttitle{Tabellenverzeichnis}%
5655 %%
5656 \def\stctitle{Inhalt}%
5657 \def\slftitle{Abbildungen}%
5658 \def\slttitle{Tabellen}%
5659 \</germanb2>

```

13.67 “Greek” language: `greek.mld`

The titles for the “greek” language (modern greek) are taken from the `greek.dtx` file in the babel package [30, 31, 72]. Greek fonts are required.

```

5660 \<greek>
5661 \ProvidesFile{greek.mld}[2006/01/13]
5662 %% Greek titles for minitoc.sty
5663 %% from greek.dtx (babel)
5664 %% Syropoulos, Apostolos

```

```

5665 %% Needs greek fonts.
5666 \def\ptctitle{Perieq'omena}%
5667 \def\plftitle{Kat'alogoc Sqhm'atwn}%
5668 \def\plttitle{Kat'alogoc Pin'akwn}%
5669 %%
5670 \def\mtctitle{Perieq'omena}%
5671 \def\mlftitle{Kat'alogoc Sqhm'atwn}%
5672 \def\mlttitle{Kat'alogoc Pin'akwn}%
5673 %%
5674 \def\stctitle{Perieq'omena}%
5675 \def\slftitle{Kat'alogoc Sqhm'atwn}%
5676 \def\slttitle{Kat'alogoc Pin'akwn}%
5677 </greek>

```

13.68 “Greek-mono” language: greek-mono.mld

The titles for the “greek-mono” language²⁵ are taken from the ANTOMEGA project [127]:

```

5678 <*greek – mono>
5679 \ProvidesFile{greek-mono.mld}[2005/02/08]
5680 %% from omega-greek.ldf (Antomega project)
5681 %% Needs Omega
5682 %% Alexej M. Kryokov
5683 %% Dmitry Ivanov
5684 %%
5685 \def\ptctitle{\localgreek%
5686 {^03a0^03b5^03c1^03b9^03b5^03c7^03cc^03bc%
5687 ^03b5^03bd^03b1}}%
5688 \def\plftitle{\localgreek%
5689 {^039a^03b1^03c4^03ac^03bb^03bf^03b3^03bf%
5690 ^03c2 ^03c3^03c7^03b7^03bc^03ac^03c4^03c9%
5691 ^03bd}}%
5692 \def\plftitle{\localgreek%
5693 {^039a^03b1^03c4^03ac^03bb^03bf^03b3^03bf%
5694 ^03c2 ^03c0^03b9^03bd^03ac^03ba^03c9^03bd}}%
5695 %%
5696 \def\mtctitle{\localgreek%
5697 {^03a0^03b5^03c1^03b9^03b5^03c7^03cc^03bc%
5698 ^03b5^03bd^03b1}}%
5699 \def\mlftitle{\localgreek%
5700 {^039a^03b1^03c4^03ac^03bb^03bf^03b3^03bf%
5701 ^03c2 ^03c3^03c7^03b7^03bc^03ac^03c4^03c9%
5702 ^03bd}}%
5703 \def\mlftitle{\localgreek%
5704 {^039a^03b1^03c4^03ac^03bb^03bf^03b3^03bf%
5705 ^03c2 ^03c0^03b9^03bd^03ac^03ba^03c9^03bd}}%
5706 %%

```

²⁵ Monotonic greek, from a recent (1982) but strongly contested – and contestable – reform of the greek language.


```

5707 \def\stctitle{\localgreek%
5708 {^03a0^03b5^03c1^03b9^03b5^03c7^03cc^03bc%
5709 ^03b5^03bd^03b1}}%
5710 \def\slftitle{\localgreek%
5711 {^039a^03b1^03c4^03ac^03bb^03bf^03b3^03bf%
5712 ^03c2 ^03c3^03c7^03b7^03bc^03ac^03c4^03c9%
5713 ^03bd}}%
5714 \def\slftitle{\localgreek%
5715 {^039a^03b1^03c4^03ac^03bb^03bf^03b3^03bf%
5716 ^03c2 ^03c0^03b9^03bd^03ac^03ba^03c9^03bd}}%
5717 </greek – mono>

```

13.69 “Greek-polydemo” language: greek-polydemo.mld

The titles for the “greek-polydemo” language²⁶ are taken from the ANTOMEGA project [127]:

```

5718 <*greek – polydemo>
5719 \ProvidesFile{greek-polydemo.mld}[2005/02/08]
5720 %% from omega-greek.ldf (Antomega project)
5721 %% Needs Omega
5722 %% Alexej M. Kryokov
5723 %% Dmitry Ivanov
5724 %%
5725 \def\ptctitle{\localgreek%
5726 {^03a0^03b5^03c1^03b9^03b5^03c7^1f79^03bc%
5727 ^03b5^03bd^03b1}}%
5728 \def\plftitle{\localgreek%
5729 {^039a^03b1^03c4^1f71^03bb^03bf^03b3^03bf%
5730 ^03c2 ^03c3^03c7^03b7^03bc^1f71^03c4^03c9%
5731 ^03bd}}%
5732 \def\plttitle{\localgreek%
5733 {^039a^03b1^03c4^1f71^03bb^03bf^03b3^03bf%
5734 ^03c2 ^03c0^03b9^03bd^1f71^03ba^03c9^03bd}}%
5735 %%
5736 \def\mtctitle{\localgreek%
5737 {^03a0^03b5^03c1^03b9^03b5^03c7^1f79^03bc%
5738 ^03b5^03bd^03b1}}%
5739 \def\mlftitle{\localgreek%
5740 {^039a^03b1^03c4^1f71^03bb^03bf^03b3^03bf%
5741 ^03c2 ^03c3^03c7^03b7^03bc^1f71^03c4^03c9%
5742 ^03bd}}%
5743 \def\mlttitle{\localgreek%
5744 {^039a^03b1^03c4^1f71^03bb^03bf^03b3^03bf%
5745 ^03c2 ^03c0^03b9^03bd^1f71^03ba^03c9^03bd}}%
5746 %%
5747 \def\stctitle{\localgreek%

```

²⁶Polytonic demotic (popular) greek, for classical greek.

```

5748 {^03a0^03b5^03c1^03b9^03b5^03c7^1f79^03bc%
5749 ^03b5^03bd^03b1}}%
5750 \def\slftitle{\localgreek%
5751 {^039a^03b1^03c4^1f71^03bb^03bf^03b3^03bf%
5752 ^03c2 ^03c3^03c7^03b7^03bc^1f71^03c4^03c9%
5753 ^03bd}}%
5754 \def\sltttitle{\localgreek%
5755 {^039a^03b1^03c4^1f71^03bb^03bf^03b3^03bf%
5756 ^03c2 ^03c0^03b9^03bd^1f71^03ba^03c9^03bd}}%
5757 \greek – polydemo)

```

13.70 “Greek-polykatha” language: greek-polykatha.mld

The titles for the “greek-polykatha” language²⁷ are taken from the ANTOMEGA project [127]:

```

5758 (*greek – polykatha)
5759 \ProvidesFile{greek-polykatha.mld}[2005/02/08]
5760 %% from omega-greek.ldf (Antomega project)
5761 %% Needs Omega
5762 %% Alexej M. Kryokov
5763 %% Dmitry Ivanov
5764 %%
5765 \def\ptctitle{\localgreek%
5766 {^03a0^03b5^03c1^03b9^03b5^03c7^1f79^03bc%
5767 ^03b5^03bd^03b1}}%
5768 \def\plftitle{\localgreek%
5769 {^039a^03b1^03c4^1f71^03bb^03bf^03b3^03bf%
5770 ^03c2 ^03c3^03c7^03b7^03bc^1f71^03c4^03c9%
5771 ^03bd}}%
5772 \def\plttitle{\localgreek%
5773 {^039a^03b1^03c4^1f71^03bb^03bf^03b3^03bf%
5774 ^03c2 ^03c0^03b9^03bd^1f71^03ba^03c9^03bd}}%
5775 %%
5776 \def\mtctitle{\localgreek%
5777 {^03a0^03b5^03c1^03b9^03b5^03c7^1f79^03bc%
5778 ^03b5^03bd^03b1}}%
5779 \def\mlftitle{\localgreek%
5780 {^039a^03b1^03c4^1f71^03bb^03bf^03b3^03bf%
5781 ^03c2 ^03c3^03c7^03b7^03bc^1f71^03c4^03c9%
5782 ^03bd}}%
5783 \def\mltttitle{\localgreek%
5784 {^039a^03b1^03c4^1f71^03bb^03bf^03b3^03bf%
5785 ^03c2 ^03c0^03b9^03bd^1f71^03ba^03c9^03bd}}%

```

²⁷ Polytonic greek, « kathaverousa » (purified) style, a form of the Greek language created during the early 19th century by Adamantios KORAI, 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.

```

5786 %%
5787 \def\stctitle{\localgreek%
5788 {^^^^03a0^^^^03b5^^^^03c1^^^^03b9^^^^03b5^^^^03c7^^^^1f79^^^^03bc%
5789 ^^^^^03b5^^^^03bd^^^^03b1}}}%
5790 \def\slftitle{\localgreek%
5791 {^^^^039a^^^^03b1^^^^03c4^^^^1f71^^^^03bb^^^^03bf^^^^03b3^^^^03bf%
5792 ^^^^^03c2 ^^^^^03c3^^^^03c7^^^^03b7^^^^03bc^^^^1f71^^^^03c4^^^^03c9%
5793 ^^^^^03bd}}}%
5794 \def\sltttitle{\localgreek%
5795 {^^^^039a^^^^03b1^^^^03c4^^^^1f71^^^^03bb^^^^03bf^^^^03b3^^^^03bf%
5796 ^^^^^03c2 ^^^^^03c0^^^^03b9^^^^03bd^^^^1f71^^^^03ba^^^^03c9^^^^03bd}}}%
5797 </greek – polykatha>

```

13.71 “Guarani” language: guarani.mld

The “guarani” language is the main language spoken in Paraguay. Very often, a mixture of Guarani and Spanish, known as Jopara, is spoken. The titles are taken from the guarani.ldf file by Javier BEZOS [26]. A special input encoding (win-gn.def) is needed. These files are available on the CTAN archives.

```

5798 <*guarani>
5799 \ProvidesFile{guarani.mld}[2005/08/26]
5800 %% Guarani titles for minitoc.sty
5801 %% from guarani.ldf by Javier Bezos.
5802 %% Input encoding win-gn.def is needed.
5803 %%
5804 \def\ptctitle{\'Indice general}%
5805 \def\plftitle{\'Indice de figuras}%
5806 \def\pltttitle{\'Indice de cuadros}%
5807 %%
5808 \def\mtctitle{\'Indice general}%
5809 \def\mlftitle{\'Indice de figuras}%
5810 \def\mltttitle{\'Indice de cuadros}%
5811 %%
5812 \def\stctitle{\'Indice general}%
5813 \def\slftitle{\'Indice de figuras}%
5814 \def\sltttitle{\'Indice de cuadros}%
5815 </guarani>

```

13.72 “Hangul1” language: hangul1.mld and hangul1.mlo

The Korean language was originally written using the Chinese characters; it is now mainly written in Hang l, the Korean writing system, optionally incorporating Hanja to write Sino-Korean words [199]. See [107, page 150] and [170].

The titles for the “hangul1” language (korean in hangŭl script, first variant) are taken from the file `hangul.cap` of the CJK system [143, 144]. Special fonts are needed, of course. See also sections 13.73, 13.74 on the next page, 13.75 on the following page, 13.76 on page 454, 13.77 on page 454, 13.78 on page 455, and 13.79 on page 455.

The titles for the “hangul1” language contain characters that cannot be easily generated, hence we load `hangul1.mlo`.

```

5816 (*hangul1)
5817 \ProvidesFile{hangul1.mld}[2005/01/28]
5818 %% From the file hangul.cap of the CJK package
5819 %% for using Asian logographs (Chinese/Japanese/Korean) with LaTeX2e
5820 %% created by Werner Lemberg <wl@gnu.org>
5821 %%
5822 %% Version 4.5.2 (28-Mar-2003)
5823 %% Hangul captions
5824 %% character set: KS X 1001:1992 (=KS C 5601-1992), encoding: EUC (=Wansung)
5825 %%
5826 \mtcloadmlo{hangul1}%
5827 </hangul1>

```

13.73 “Hangul2” language: `hangul2.mld` and `hangul2.mlo`

The titles for the “hangul2” language (korean in hangŭl script, second variant) are taken from the file `hangul.cpx` of the CJK system [143, 144]. Special fonts are needed, of course. See also sections 13.72 on the page before, 13.74 on the following page, 13.75 on the next page, 13.76 on page 454, 13.77 on page 454, 13.78 on page 455, and 13.79 on page 455.

The titles for the “hangul2” language contain characters that cannot be easily generated, hence we load `hangul2.mlo`.

```

5828 (*hangul2)
5829 \ProvidesFile{hangul2.mld}[2005/01/28]
5830 %% From the file hangul.cpx of the CJK package
5831 %% for using Asian logographs (Chinese/Japanese/Korean) with LaTeX2e
5832 %% created by Werner Lemberg <wl@gnu.org>
5833 %%
5834 %% Version 4.5.2 (28-Mar-2003)
5835 %% Hangul captions
5836 %% character set: KS X 1001:1992 (=KS C 5601-1992),
5837 %% encoding: EUC (=Wansung), preprocessed
5838 %%
5839 \mtcloadmlo{hangul2}%
5840 </hangul2>

```

13.74 “Hangul3” language: hangul3.mld and hangul3.mlo

The titles for the “hangul3” language (korean in hangûl script, third variant) are taken from the file hangul2.cap of the CJK system [143, 144]. Special fonts are needed, of course. See also sections 13.72 on page 451, 13.73 on the page before, 13.75, 13.76 on the next page, 13.77 on the following page, 13.78 on page 455, and 13.79 on page 455.

The titles for the “hangul3” language contain characters that cannot be easily generated, hence we load hangul3.mlo.

```

5841 (*hangul3)
5842 \ProvidesFile{hangul3.mld}[2005/01/28]
5843 %% From the file hangul2.cap of the CJK package
5844 %% for using Asian logographs (Chinese/Japanese/Korean) with LaTeX2e
5845 %% created by Werner Lemberg <wl@gnu.org>
5846 %%
5847 %% Version 4.5.2 (28-Mar-2003)
5848 %% Hangul captions set 2
5849 %% character set: KS X 1001:1992 (=KS C 5601-1992), encoding: EUC (=Wansung)
5850 %%
5851 \mtcloadmlo{hangul3}%
5852 \hangul3

```

13.75 “Hangul4” language: hangul4.mld and hangul4.mlo

The titles for the “hangul4” language (korean in hangûl script, fourth variant) are taken from the file hangul2.cpx of the CJK system [143, 144]. Special fonts are needed, of course. See also sections 13.72 on page 451, 13.73 on the preceding page, 13.74, 13.76 on the following page, 13.77 on the next page, 13.78 on page 455, and 13.79 on page 455.

The titles for the “hangul4” language contain characters that cannot be easily generated, hence we load hangul4.mlo.

```

5853 (*hangul4)
5854 \ProvidesFile{hangul4.mld}[2005/01/28]
5855 %% From the file hangul2.cpx of the CJK package
5856 %% for using Asian logographs (Chinese/Japanese/Korean) with LaTeX2e
5857 %% created by Werner Lemberg <wl@gnu.org>
5858 %%
5859 %% Version 4.5.2 (28-Mar-2003)
5860 %% Hangul captions set 2, character set: KS X 1001:1992 (=KS C 5601-1992),
5861 %% encoding: EUC (=Wansung), preprocessed
5862 %%
5863 \mtcloadmlo{hangul4}%

```

```
5864 </hangul4>
```

13.76 “Hangul-u8” language: `hangul-u8.mld` and `hangul-u8.mlo`

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 [124, in korean] by KOAUNGHl Un. Special fonts are needed, of course. Input encoding is UTF-8. See also sections 13.72 on page 451, 13.73 on page 452, 13.74 on the preceding page, 13.75 on the page before, 13.77, 13.78 on the next page, and 13.79 on the following page. See [107, page 150] and [170].

The titles for the “hangul-u8” language contain characters that cannot be easily generated, hence we load `hangul-u8.mlo`.

```
5865 < *hangul - u8 >
5866 \ProvidesFile{hangul-u8.mld}[2006/02/21]
5867 %% From the file u8hangul.tex of the HLaTeX package
5868 %% by Koaunghi Un (koaunghi@kornet.net)
5869 %%
5870 %% Hangul captions for Lambda
5871 %%
5872 \mctloadmlo{hangul-u8}%
5873 </hangul - u8 >
```

13.77 “Hanja1” language: `hanja1.mld` and `hanja1.mlo`

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 [143, 144]. Special fonts are needed, of course. See also sections 13.72 on page 451, 13.73 on page 452, 13.74 on the page before, 13.75 on the preceding page, 13.76, 13.78 on the following page, and 13.79 on the next page.

The titles for the “hanja1” language contain characters that cannot be easily generated, hence we load `hanja1.mlo`.

```
5874 < *hanja1 >
5875 \ProvidesFile{hanja1.mld}[2005/01/28]
5876 %% From the file hanja.cpx of the CJK package
5877 %% for using Asian logographs (Chinese/Japanese/Korean) with LaTeX2e
5878 %% created by Werner Lemberg <wl@gnu.org>
5879 %%
```

```

5880%% Version 4.5.2 (28-Mar-2003)
5881%% Hanja captions, character set: KS X 1001:1992 (=KS C 5601-1992),
5882%% encoding: EUC (=Wansung), preprocessed
5883%%
5884\mtcloadmlo{hanja1}%
5885</hanja1>

```

13.78 “Hanja2” language: hanja2.mld and hanja2.mlo

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 [143, 144]. Special fonts are needed, of course. See also sections 13.72 on page 451, 13.73 on page 452, 13.74 on page 453, 13.75 on page 453, 13.76 on the preceding page, 13.77 on the page before, and 13.79.

The titles for the “hanja2” language contain characters that cannot be easily generated, hence we load `hanja2.mlo`.

```

5886<(*hanja2)
5887\ProvidesFile{hanja2.mld}[2005/01/28]
5888%% From the file hanja.cap of the CJK package
5889%% for using Asian logographs (Chinese/Japanese/Korean) with LaTeX2e
5890%% created by Werner Lemberg <a7971428@unet.univie.ac.at>
5891%%
5892%% Version 4.1.3 (20-Jun-1997)
5893%% Hanja captions, character set: KS X 1001:1992 (=KS C 5601-1992),
5894%% encoding: EUC (=Wansung)
5895%%
5896\mtcloadmlo{hanja2}%
5897</hanja2>

```

13.79 “Hanja-u8” language: hanja-u8.mld and hanja-u8.mlo

The titles for the “hanja-u8” language (korean in hanja script, for Lambda Λ) are taken from the file `u8hanja.tex` of the H_ΛT_EX system [in korean][124] by KOAUNGHI Un. Special fonts are needed, of course. Input encoding is UTF-8. See also section 13.72 on page 451, 13.73 on page 452, 13.74 on page 453, 13.75 on page 453, 13.76 on the page before, 13.77 on the preceding page, and 13.78. See [107, page 150] and [170].

The titles for the “hanja-u8” language contain characters that cannot be easily generated, hence we load `hanja-u8.mlo`.

```

5898 <*hanja – u8>
5899 \ProvidesFile{hanja-u8.mld}[2006/02/21]
5900 %% From the file hanja-u8.tex of the HLaTeX package
5901 %% by Koaunghi Un (koaunghi@kornet.net)
5902 %%
5903 %% Hanja captions for Lambda
5904 %%
5905 \mtcloadmlo{hanja-u8}%
5906 </hanja – u8>

```

13.80 “Hebrew” language: hebrew.mld

The titles for the “hebrew” language are taken from the ArabTeX package [130, 131], which should be used, with the associated fonts. See also section 13.81 on the following page.

```

5907 <*hebrew>
5908 \ProvidesFile{hebrew.mld}[2001/02/28]
5909 %% Hebrew titles for minitoc.sty
5910 %% Need hebrew fonts (see arabtex documentation)
5911 \def\ptctitle{\tav\vav\kaf\finalnun\
5912             \ayin\nun\yod\nun\yod\finalmem}%
5913 \def\plftitle{\resh\shin\yod\mem\tav\
5914             \alef\yod\vav\resh\yod\finalmem}%
5915 \def\plttitle{\resh\shin\yod\mem\tav\
5916             \tet\bet\lamed\alef\vav\tav}%
5917 %%
5918 \def\mtctitle{\tav\vav\kaf\finalnun\
5919             \ayin\nun\yod\nun\yod\finalmem}%
5920 \def\mlftitle{\resh\shin\yod\mem\tav\
5921             \alef\yod\vav\resh\yod\finalmem}%
5922 \def\mlttitle{\resh\shin\yod\mem\tav\
5923             \tet\bet\lamed\alef\vav\tav}%
5924 %%
5925 \def\stctitle{\tav\vav\kaf\finalnun\
5926             \ayin\nun\yod\nun\yod\finalmem}%
5927 \def\slftitle{\resh\shin\yod\mem\tav\
5928             \alef\yod\vav\resh\yod\finalmem}%
5929 \def\slttitle{\resh\shin\yod\mem\tav\
5930             \tet\bet\lamed\alef\vav\tav}%
5931 </hebrew>

```


13.81 “Hebrew2” language: hebrew2.mld

The titles for the “hebrew2” language are taken from the file `hebrew.dtx` (by Boris LAVVA) in the Babel package [30, 31, 60], which should be used, with the associated fonts and encodings. See also section 13.80 on the page before.

```

5932 (*hebrew2)
5933 \ProvidesFile{hebrew2.mld}[2006/01/11]
5934 %% From hebrew.dtx in the Babel package.
5935 %% Boris Lavva (lavva@tx.technion.ac.il)
5936 %% Need hebrew fonts.
5937 \def\ptctitle{\@ensure@R{%
5938   \hebtav\hebvav\hebkafe\hebfinalnun\ %
5939   \hebayin\hebnun\hebyod\hebyod\hebnun\hebyod\hebfinalmem}}%
5940 \def\plftitle{\@ensure@R{%
5941   \hebrehe\hebshin\hebyod\hebmeme\hebtav\ %
5942   \hebalef\hebyod\hebvav\hebrehe\hebyod\hebfinalmem}}%
5943 \def\plttitle{\@ensure@R{%
5944   \hebrehe\hebshin\hebyod\hebmeme\hebtav\
5945   \hebtet\hebbet\heblamed\hebalef\hebvav\hebtav}}%
5946 %%
5947 \def\mtctitle{\@ensure@R{%
5948   \hebtav\hebvav\hebkafe\hebfinalnun\ %
5949   \hebayin\hebnun\hebyod\hebyod\hebnun\hebyod\hebfinalmem}}%
5950 \def\mlftitle{\@ensure@R{%
5951   \hebrehe\hebshin\hebyod\hebmeme\hebtav\ %
5952   \hebalef\hebyod\hebvav\hebrehe\hebyod\hebfinalmem}}%
5953 \def\mlttitle{\@ensure@R{%
5954   \hebrehe\hebshin\hebyod\hebmeme\hebtav\
5955   \hebtet\hebbet\heblamed\hebalef\hebvav\hebtav}}%
5956 %%
5957 \def\stctitle{\@ensure@R{%
5958   \hebtav\hebvav\hebkafe\hebfinalnun\ %
5959   \hebayin\hebnun\hebyod\hebyod\hebnun\hebyod\hebfinalmem}}%
5960 \def\slftitle{\@ensure@R{%
5961   \hebrehe\hebshin\hebyod\hebmeme\hebtav\ %
5962   \hebalef\hebyod\hebvav\hebrehe\hebyod\hebfinalmem}}%
5963 \def\slttitle{\@ensure@R{%
5964   \hebrehe\hebshin\hebyod\hebmeme\hebtav\
5965   \hebtet\hebbet\heblamed\hebalef\hebvav\hebtav}}%
5966 /hebrew2)

```

13.82 “Hungarian” language: hungarian.mld

The “hungarian” language is a synonym of the “magyar” language, so we load `magyar.mld`. See section 13.104 on page 468.

```

5967 <(*hungarian)
5968 \ProvidesFile{hungarian.mld}[2004/12/14]
5969 \mtcselectlanguage{magyar}%
5970 </hungarian>

```

13.83 “Icelandic” language: `icelandic.mld`

The titles for the “icelandic” language are taken from the `icelandic.dtx` file in the `babel` package [30, 31, 38]:

```

5971 <(*icelandic)
5972 \ProvidesFile{icelandic.mld}[2006/01/13]
5973 %% Icelandic titles for minitoc.sty
5974 %% from icelandic.dtx (babel)
5975 %% Árnason, Einar
5976 %% need inputenc with 8-bits encoding
5977 \def\ptctitle{Efnisyfirlit}%
5978 \def\plftitle{Myndaskrá}%
5979 \def\plttitle{Töfluskrá}%
5980 %%
5981 \def\mtctitle{Efnisyfirlit}%
5982 \def\mlftitle{Myndaskrá}%
5983 \def\mlttitle{Töfluskrá}%
5984 %%
5985 \def\stctitle{Efnisyfirlit}%
5986 \def\slftitle{Myndaskrá}%
5987 \def\slttitle{Töfluskrá}%
5988 </icelandic>

```

13.84 “Indon” language: `indon.mld`

The “indon” language is just like “bahasai”, so we just load `bahasai.mld` (see section 13.15 on page 422):

```

5989 <(*indon)
5990 \ProvidesFile{indon.mld}[2006/01/13]
5991 \mtcselectlanguage{bahasai}%
5992 </indon>

```

13.85 “Indonesian” language: `indonesian.mld`

The “indonesian” language is just like “bahasai”, so we just load `bahasai.mld` (see section 13.15 on page 422):

```

5993 <*indonesian>
5994 \ProvidesFile{indonesian.mld}[2006/01/13]
5995 \mtcselectlanguage{bahasai}%
5996 </indonesian>

```

13.86 “Interlingua” language: `interlingua.mld`

The titles for the “interlingua” language are taken from the `interlingua.dtx` file in the `babel` package [30, 31, 55]. 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. 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. You can have a look at the grammar at <http://www.geocities.com/linguablau>

```

5997 <*interlingua>
5998 \ProvidesFile{interlingua.mld}[2006/01/13]
5999 %% Interlingua titles for minitoc.sty
6000 %% from interlingua.dtx (babel)
6001 %% Kleiweg, Peter
6002 \def\ptctitle{Contento}%
6003 \def\plftitle{Lista de Figuras}%
6004 \def\pltttitle{Lista de Tabellas}%
6005 %%
6006 \def\mtctitle{Contento}%
6007 \def\mlftitle{Figuras}%
6008 \def\mltttitle{Tabellas}%
6009 %%
6010 \def\stctitle{Contento}%
6011 \def\slftitle{Figuras}%
6012 \def\sltttitle{Tabellas}%
6013 </interlingua>

```

13.87 “Irish” language: `irish.mld`

The titles for the “irish” language are taken from the `irish.dtx` file in the `babel` package [30, 31, 34]:

```

6014 <*irish>
6015 \ProvidesFile{irish.mld}[2006/02/28]
6016 %% Irish titles for minitoc.sty
6017 %% from irish.dtx (babel)
6018 %% Braams, Johannes and Gunn, Marion and Grant, Fraser
6019 \def\ptctitle{Cl\'ar \'Abhair}%
6020 \def\plftitle{L\'ear\'aid\'{\i}}%

```

```

6021 \def\pltttitle{T\'abla\'\{i}}%
6022 %%
6023 \def\mtctitle{Cl\'ar \'Abhair}%
6024 \def\mlfttitle{L\'ear\'aid\'\{i}}%
6025 \def\mltttitle{T\'abla\'\{i}}%
6026 %%
6027 \def\stctitle{Cl\'ar \'Abhair}%
6028 \def\slfttitle{L\'ear\'aid\'\{i}}%
6029 \def\slttitle{T\'abla\'\{i}}%
6030 </irish>

```

13.88 “Italian” language: `italian.mld`

The titles for the “italian” language are taken from the file `italian.dtx` in the `babel` package [30, 31, 47]. See also section 13.89.

```

6031 <*italian>
6032 \ProvidesFile{italian.mld}[2006/01/13]
6033 %% Italian titles for minitoc.sty
6034 %% from italian.dtx (babel)
6035 %% Maurizio Codogno, (mau@beatles.cselt.stet.it)
6036 %% and Claudio Beccari, (beccari@polito.it)
6037 \def\ptctitle{Indice}%
6038 \def\plfttitle{Elenco delle figure}%
6039 \def\pltttitle{Elenco delle tabelle}%
6040 %%
6041 \def\mtctitle{Indice}%
6042 \def\mlfttitle{Elenco delle figure}%
6043 \def\mltttitle{Elenco delle tabelle}%
6044 %%
6045 \def\stctitle{Indice}%
6046 \def\slfttitle{Elenco delle figure}%
6047 \def\slttitle{Elenco delle tabelle}%
6048 </italian>

```

13.89 “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 13.88.

```

6049 <*italian2>
6050 \ProvidesFile{italian2.mld}[2006/01/13]
6051 %% Italian titles for minitoc.sty. Variant.
6052 %% from italian.dtx (babel)
6053 %% Maurizio Codogno, (mau@beatles.cselt.stet.it)
6054 %% and Claudio Beccari, (beccari@polito.it)

```

```

6055 \def\ptctitle{Contenuto}%
6056 \def\plftitle{Elenco delle figure}%
6057 \def\plttitle{Elenco delle tabelle}%
6058 %%
6059 \def\mtctitle{Contenuto}%
6060 \def\mlftitle{Elenco delle figure}%
6061 \def\mlttitle{Elenco delle tabelle}%
6062 %%
6063 \def\stctitle{Contenuto}%
6064 \def\slftitle{Elenco delle figure}%
6065 \def\slttitle{Elenco delle tabelle}%
6066 </italian2>

```

13.90 “Japanese” language: `japanese.mld` and `japanese.mlo`

They 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 KUMAZAWA²⁸. See also sections 13.91, 13.92 on the following page, 13.93 on the next page, and 13.94 on page 463. The titles for the “japanese” language contain characters that cannot be easily generated, hence we load `japanese.mlo`.

```

6067 < *japanese>
6068 \ProvidesFile{japanese.mld}[2006/01/13]
6069 %% Japanese titles for minitoc.sty
6070 %% Needs japanese fonts (CJK) and special input encoding.
6071 %% From Kumazawa Toshiki
6072 %% kumazawa@biwako.shiga-u.ac.jp
6073 %% http://www.biwako.shiga-u.ac.jp/sensei/kumazawa/tex/minitoc.html
6074 \mtcloadmlo{japanese}%
6075 </japanese>

```

13.91 “Japanese2” language: `japanese2.mld` and `japanese2.mlo`

The titles for the “japanese2” language (japanese, second variant) are taken from file JIS.cap of the CJK system [143, 144]. Special fonts are needed, of course. See also sections 13.90, 13.92 on the following page, 13.93 on the next page, and 13.94 on page 463. The titles for the “japanese2” language contain characters that cannot be easily generated, hence we load `japanese2.mlo`.

²⁸ <http://www.biwako.shiga-u.ac.jp/sensei/kumazawa/tex/minitoc.html>

```

6076 <*japanese2>
6077 \ProvidesFile{japanese2.mld}[2006/01/13]
6078 %% From the file JIS.cap of the CJK package
6079 %% for using Asian logographs (Chinese/Japanese/Korean) with LaTeX2e
6080 %% created by Werner Lemberg <wl@gnu.org>
6081 %%
6082 %% Version 4.5.2 (28-Mar-2003)
6083 %% Japanese captions, character set: JIS X 0208:1997 (or JIS X 0208-1990),
6084 %% encoding: EUC
6085 %%
6086 \mtcloadmlo{japanese2}%
6087 </japanese2>

```

13.92 “Japanese3” language: `japanese3.mld` and `japanese3.mlo`

The titles for the “japanese3” language (japanese, third variant) are taken from file `JIS.cpx` of the CJK system [143, 144]. Special fonts are needed, of course. See also sections 13.90 on the page before, 13.91 on the preceding page, 13.93, and 13.94 on the following page. The titles for the “japanese3” language contain characters that cannot be easily generated, hence we load `japanese3.mlo`.

```

6088 <*japanese3>
6089 \ProvidesFile{japanese3.mld}[2006/01/13]
6090 %% From the file JIS.cpx of the CJK package
6091 %% for using Asian logographs (Chinese/Japanese/Korean) with LaTeX2e
6092 %% created by Werner Lemberg <wl@gnu.org>
6093 %%
6094 %% Version 4.5.2 (28-Mar-2003)
6095 %% Japanese captions, character set: JIS X 0208:1997 (or JIS X 0208-1990)
6096 %% encoding: EUC, preprocessed
6097 %%
6098 \mtcloadmlo{japanese3}%
6099 </japanese3>

```

13.93 “Japanese4” language: `japanese4.mld` and `japanese4.mlo`

The titles for the “japanese4” language (japanese, fourth version) are taken from file `SJIS.cap` of the CJK system [143, 144]. Special fonts are needed, of course. See also sections 13.90 on the page before, 13.91 on the preceding page, 13.92, and 13.94 on the following page. The titles for the “japanese4” language contain characters that cannot be easily generated, hence we load `japanese4.mlo`.

```

6100 <*japanese4>
6101 \ProvidesFile{japanese4.mld}[2006/01/13]
6102 %% From the file SJIS.cap of the CJK package
6103 %% for using Asian logographs (Chinese/Japanese/Korean) with LaTeX2e
6104 %% created by Werner Lemberg <wl@gnu.org>
6105 %%
6106 %% Version 4.5.2 (28-Mar-2003)
6107 %% Japanese captions
6108 %% character set: JIS X 0208:1997 (or JIS X 0208-1990), encoding: SJIS
6109 %%
6110 \mtcloadmlo{japanese4}%
6111 </japanese4>

```

13.94 “Japanese5” language: `japanese5.mld` and `japanese.mlo`

The titles for the “japanese5” (japanese, fifth variant) language are taken from file `SJIS.cpx` of the CJK system [143, 144]. Special fonts are needed, of course. See also sections 13.90 on page 461, 13.91 on page 461, 13.92 on the page before, and 13.93 on the preceding page. The titles for the “japanese5” language contain characters that cannot be easily generated, hence we load `japanese5.mlo`.

```

6112 <*japanese5>
6113 \ProvidesFile{japanese5.mld}[2006/01/13]
6114 %% From the file SJIS.cpx of the CJK package
6115 %% for using Asian logographs (Chinese/Japanese/Korean) with LaTeX2e
6116 %% created by Werner Lemberg <wl@gnu.org>
6117 %%
6118 %% Version 4.5.2 (28-Mar-2003)
6119 %% Japanese captions
6120 %% character set: JIS X 0208:1997 (or JIS X 0208-1990),
6121 %% encoding: SJIS, preprocessed
6122 %%
6123 \mtcloadmlo{japanese5}%
6124 </japanese5>

```

13.95 “Kannada” language: `kannada.mld`

The Kannada (or Kannara) is a dravidian language spoken in the Karnataka state of India. The titles are taken from the `kanlel.sty` package file from the Kannadatex project²⁹ by C.S. YOGANANDA. Specific fonts are required.

²⁹ <http://Sarovar.org/projects/kannadatex>

```

6125 <*kannada>
6126 \ProvidesFile{kannada.mld}[2006/02/14]
6127 %% from kanlel.sty of Kannadatex (C.S. Yogananda yoga@math.iisc.ernet.in)
6128 %%
6129 \def\ptctitle{pariviDi}
6130 \def\plftitle{citarxgaLa paTiTx}
6131 \def\plftitle{koVSaTxkagaLa paTiTx}
6132 %%
6133 \def\mtctitle{pariviDi}
6134 \def\mlftitle{citarxgaLa paTiTx}
6135 \def\mlftitle{koVSaTxkagaLa paTiTx}
6136 %%
6137 \def\stctitle{pariviDi}
6138 \def\slftitle{citarxgaLa paTiTx}
6139 \def\slftitle{koVSaTxkagaLa paTiTx}
6140 </kannada>

```

13.96 “Khalka” language: khalkha.mld

“khalkha” is a synonym for “xalx”, so we just load xalx.mld (see sections 13.169 on page 500, 13.170 on page 500, and 13.171 on page 501):

```

6141 <*khalkha>
6142 \ProvidesFile{khalkha.mld}[2005/11/16]
6143 \mtcselectlanguage{xalx}%
6144 </khalkha>

```

13.97 “Latin” language: latin.mld

The titles for the “latin” language are taken from the latin.dtx file in the babel package [30, 31, 40]. See also section 13.98 on the following page.

```

6145 <*latin>
6146 \ProvidesFile{latin.mld}[2006/01/13]
6147 %% Latin titles for minitoc.sty
6148 %% from latin.dtx (babel)
6149 %% Beccari, Claudio
6150 \def\ptctitle{Index}%
6151 \def\plftitle{Conspectus descriptionum}%
6152 \def\pltttitle{Conspectus tabularum}%
6153 %%
6154 \def\mtctitle{Index}%
6155 \def\mlftitle{Conspectus descriptionum}%
6156 \def\mltttitle{Conspectus tabularum}%
6157 %%
6158 \def\stctitle{Index}%

```



```

6159 \def\slftitle{Conspectus descriptionum}%
6160 \def\sltttitle{Conspectus tabularum}%
6161 \</latin>

```

13.98 “Latin2” language: latin2.mld

The titles for the “latin2” language (latin, abbreviated variant) are taken from the latin.dtx file in the babel package [30, 31, 40], but abbreviated. See also section 13.97 on the page before.

```

6162 \<latin2>
6163 \ProvidesFile{latin2.mld}[2006/01/13]
6164 %% Latin titles (shortened) for minitoc.sty
6165 %% from latin.dtx (babel)
6166 %% Beccari, Claudio
6167 \def\ptctitle{Index}%
6168 \def\plftitle{Conspectus descriptionum}%
6169 \def\pltttitle{Conspectus tabularum}%
6170 %%
6171 \def\mtctitle{Index}%
6172 \def\mlftitle{Descriptiones}%
6173 \def\mltttitle{Tabulae}%
6174 %%
6175 \def\stctitle{Index}%
6176 \def\slftitle{Descriptiones}%
6177 \def\sltttitle{Tabulae}%
6178 \</latin2>

```

13.99 “Latvian” language: latvian.mld

The titles for the “latvian” language³⁰ are taken from the ANTOMEGA project [127]. See also section 13.100 on the following page:

```

6179 \<latvian>
6180 \ProvidesFile{latvian.mld}[2005/02/08]
6181 %% from latvian.ldf (Antomega project)
6182 %% Needs Omega
6183 %% Alexej M. Kryokov
6184 %% Dmitry Ivanov
6185 %%
6186 \def\ptctitle{\locallatvian{Saturs}}%
6187 \def\plftitle{\locallatvian{Att^^^0113lu saraksts}}%
6188 \def\pltttitle{\locallatvian{Tabulu saraksts}}%
6189 %%

```

³⁰“Latvian” is the original name for “letton”.

```

6190 \def\mtctitle{\locallatvian{Saturdays}}%
6191 \def\mlftitle{\locallatvian{Att113lu saraksts}}%
6192 \def\mltttitle{\locallatvian{Tabulu saraksts}}%
6193 %%
6194 \def\stctitle{\locallatvian{Saturdays}}%
6195 \def\slftitle{\locallatvian{Att113lu saraksts}}%
6196 \def\sltttitle{\locallatvian{Tabulu saraksts}}%
6197 </latvian>

```

13.100 “Letton” language: `letton.mld`

The “letton” language is a synonym for the “latvian” language, so we just load `latvian.mld`. See section 13.99 on the page before.

```

6198 <*\letton>
6199 \ProvidesFile{letton.mld}[2005/02/08]
6200 \mtcselectlanguage{latvian}%
6201 </letton>

```

13.101 “Lithuanian” language: `lithuanian.mld`

The titles for the “lithuanian” language are taken from the `lithuanian.ldf` file³¹ for the `babel` package [30, 31]:

```

6202 <*\lithuanian>
6203 \ProvidesFile{lithuanian.mld}[2006/01/13]
6204 %% Lithuanian titles for minitoc.sty
6205 %% from lithuanian.ldf
6206 %% in http://www.vtex.lt/tex/download/texinput/babel/babel.zip
6207 %% Tolusis, Sigitas (sigitas@vtex.lt)
6208 \def\ptctitle{Turinys}%
6209 \def\plftitle{Paveiksl\protect\c u s\protect\c ara\protect\v sas}%
6210 \def\pltttitle{Lentel\protect\es}%
6211 %%
6212 \def\mtctitle{Turinys}%
6213 \def\mlftitle{Paveiksl\protect\c u s\protect\c ara\protect\v sas}%
6214 \def\mltttitle{Lentel\protect\es}%
6215 %%
6216 \def\stctitle{Turinys}%
6217 \def\slftitle{Paveiksl\protect\c u s\protect\c ara\protect\v sas}%
6218 \def\sltttitle{Lentel\protect\es}%
6219 </lithuanian>

```

³¹Found in <http://www.vtex.lt/tex/download/texinput/babel/babel.zip>.

13.102 “Lowersorbian” language: lowersorbian.mld

The titles for the “lowsorbian” language³² are taken from the `lsorbian.dtx` file in the `babel` package [30, 31, 73]. See also section 13.163 on page 497. A shorter language name is `lsorbian` (see section 13.103).

```

6220 (*lowsorbian)
6221 \ProvidesFile{lowsorbian.mld}[2006/02/28]
6222 %% Lower sorbian titles for minitoc.sty
6223 %% from lsorbian.dtx (babel)
6224 %% Werner, Eduard
6225 \def\ptctitle{Wop\'simje\'se}%
6226 \def\plftitle{Zapis wobrazow}%
6227 \def\pltttitle{Zapis tabulkow}%
6228 %%
6229 \def\mtctitle{Wop\'simje\'se}%
6230 \def\mlftitle{Zapis wobrazow}%
6231 \def\mltttitle{Zapis tabulkow}%
6232 %%
6233 \def\stctitle{Wop\'simje\'se}%
6234 \def\slftitle{Zapis wobrazow}%
6235 \def\slttitle{Zapis tabulkow}%
6236 \end{lowsorbian}

```

13.103 “Lsorbian” language: lsorbian.mld

The “lsorbian” language is a synonym for “lowsorbian”, so we just load `lowsorbian.mld`. See section 13.102.

```

6237 (*lsorbian)
6238 \ProvidesFile{lsorbian.mld}[2006/01/23]
6239 %% Lower sorbian titles for minitoc.sty
6240 %% from lsorbian.dtx (babel)
6241 %% Werner, Eduard
6242 \selectlanguage{lowsorbian}%
6243 \end{lsorbian}

```

³²Lower sorbian. Sorbian, or wendisch, is a member of the west slavic subgroup of indo-european languages spoken in Lower 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 (VIth century).

13.104 “Magyar” language: magyar.mld

The titles for the “magyar” language are taken from the magyar.dtx file in the babel package [30, 31, 41]. A synonym of “magyar” is “hungarian” (see section 13.82 on page 457). See also sections 13.105 and 13.106 on the next page for variants (it seems that magyar.dtx might have evolved).

```

6244 <*magyar>
6245 \ProvidesFile{magyar.mld}[2006/03/08]
6246 %% Magyar titles for minitoc.sty
6247 %% from magyar.dtx (babel)
6248 %% Bíró, Árpád and Bérces, József
6249 \def\ptctitle{Tartalom}%
6250 \def\plftitle{\`Abr\`ak}%
6251 \def\pltttitle{T\`abl\`azatok}%
6252 %%
6253 \def\mtctitle{Tartalom}%
6254 \def\mlftitle{\`Abr\`ak}%
6255 \def\mltttitle{T\`abl\`azatok}%
6256 %%
6257 \def\stctitle{Tartalom}%
6258 \def\slftitle{\`Abr\`ak}%
6259 \def\slttitle{T\`abl\`azatok}%
6260 </magyar>

```

13.105 “Magyar2” language: magyar2.mld

The titles for the “magyar2” language are taken from a variant proposed in the babel package [30, 31]. See also sections 13.104 and 13.106 on the next page.

```

6261 <*magyar2>
6262 \ProvidesFile{magyar.mld}[2006/03/08]
6263 %% Magyar2 titles for minitoc.sty (variant)
6264 %% from magyar.dtx (babel)
6265 %% Bíró, Árpád and Bérces, József
6266 \def\ptctitle{Tartalom}%
6267 \def\plftitle{\`Abr\`ak list\`aja}%
6268 \def\pltttitle{T\`abl\`azatok list\`aja}%
6269 %%
6270 \def\mtctitle{Tartalom}%
6271 \def\mlftitle{\`Abr\`ak list\`aja}%
6272 \def\mltttitle{T\`abl\`azatok list\`aja}%
6273 %%
6274 \def\stctitle{Tartalom}%
6275 \def\slftitle{\`Abr\`ak list\`aja}%
6276 \def\slttitle{T\`abl\`azatok list\`aja}%
6277 </magyar2>

```

13.106 “Magyar3” language: magyar.mld

The titles for the “magyar3” language (third variant of magyar) are taken from the magyar.dtx file in the babel package [30, 31, 41]. See also sections 13.104 on the preceding page and 13.105 on the page before.

```

6278 <*magyar3>
6279 \ProvidesFile{magyar3.mld}[2006/03/08]
6280 %% Magyar3 titles for minitoc.sty (variant)
6281 %% from magyar.dtx (babel)
6282 %% Bíró, Árpád and Bérces, József
6283 \def\ptctitle{Tartalomjegyz\’ek}%
6284 \def\plftitle{\’Abr\’ak jegyz\’eke}%
6285 \def\pltttitle{T\’abl\’azatok jegyz\’eke}%
6286 %%
6287 \def\mtctitle{Tartalomjegyz\’ek}%
6288 \def\mlftitle{\’Abr\’ak jegyz\’eke}%
6289 \def\mltttitle{T\’abl\’azatok jegyz\’eke}%
6290 %%
6291 \def\stctitle{Tartalomjegyz\’ek}%
6292 \def\slftitle{\’Abr\’ak jegyz\’eke}%
6293 \def\slttitle{T\’abl\’azatok jegyz\’eke}%
6294 </magyar3>

```

13.107 “Malay” language: malay.mld

The “malay” language is just like “bahasam”, so we just load bahasam.mld (see section 13.16 on page 422):

```

6295 <*malay>
6296 \ProvidesFile{malay.mld}[2006/01/11]
6297 \mtcselectlanguage{bahasam}%
6298 </malay>

```

13.108 “Malayalam-keli” language: malayalam-keli.mld

The titles for the “malayalam” language³³, with the “Keli” fonts, are taken from the malayalam package [3]. This language requires specific fonts. See also sections 13.110 on the next page and 13.111 on page 471.

³³The 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.

```

6299 <*malayalam – keli>
6300 \ProvidesFile{malayalam-keli.mld}[2006/01/13]
6301 %%
6302 %% Malayalam: Keli fonts
6303 %%
6304 \def\ptctitle{\mm \X{\<68>}\X{\<197>}\X{\<83>}\X{\<161>}\<119>}%
6305 \def\plftitle{\mm \X{\<78>\<110>}\X{\<123>\<88>}\X{\<167>}\X{\<196>}}%
6306 \def\pltttitle{\mm \X{\<116>\<83>}\X{\<95>\<110>}\X{\<102>\<112>}\X{\<73>}\X{\<196>}}%
6307 %%
6308 \def\mtctitle{\mm \X{\<68>}\X{\<197>}\X{\<83>}\X{\<161>}\<119>}%
6309 \def\mlftitle{\mm \X{\<78>\<110>}\X{\<123>\<88>}\X{\<167>}\X{\<196>}}%
6310 \def\mltttitle{\mm \X{\<116>\<83>}\X{\<95>\<110>}\X{\<102>\<112>}\X{\<73>}\X{\<196>}}%
6311 %%
6312 \def\stctitle{\mm \X{\<68>}\X{\<197>}\X{\<83>}\X{\<161>}\<119>}%
6313 \def\slftitle{\mm \X{\<78>\<110>}\X{\<123>\<88>}\X{\<167>}\X{\<196>}}%
6314 \def\sltttitle{\mm \X{\<116>\<83>}\X{\<95>\<110>}\X{\<102>\<112>}\X{\<73>}\X{\<196>}}%
6315 </malayalam – keli>

```

13.109 “Malayalam-omega” language: malayalam-omega.mld and malayalam-omega.mlo

This is the Malayalam language implementation when you use Lambda (Λ) (the version of L^AT_EX for Omega) via the `omal` package [4] of the Malayalam-Omega project³⁴. 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 `omal` package.

```

6316 <*malayalam – omega>
6317 \ProvidesFile{malayalam-omega.mld}[2006/02/13]
6318 % from omal.sty (Alex A.J. indicTeX@gmail.com)
6319 \mteclloadmlo{malayalam-omega}
6320 %
6321 </malayalam – omega>

```

13.110 “Malayalam-rachana” language: malayalam-rachana.mld

The titles for the “malayalam” language, with the traditionnal “Rachana” fonts (old lipi), are taken from the `malayalam` package [3]. This language requires specific fonts. See also sections 13.108 on the page before and 13.111 on the following page.

³⁴ <http://Sarovar.org/projects/malayalam>

```

6322 (*malayalam – rachana)
6323 \ProvidesFile{malayalam-rachana.mld}[2005/06/07]
6324 %%
6325 %% Malayalam: Rachana fonts, traditionnal.
6326 %%
6327 \def\ptctitle{\mm \X{<68>}\X{<201>}\X{<83>}\X{<183>}\<119>}%
6328 \def\plftitle{\mm \X{<78>}\<111>}\X{<C<94>}\X{<186>}\X{<179>}}%
6329 \def\pltttitle{\mm \X{<117>}\<83>}\X{<95>}\<111>}\X{<F<59>}\X{<73>}\X{<179>}}%
6330 %%
6331 \def\mtctitle{\mm \X{<68>}\X{<201>}\X{<83>}\X{<183>}\<119>}%
6332 \def\mlftitle{\mm \X{<78>}\<111>}\X{<C<94>}\X{<186>}\X{<179>}}%
6333 \def\mltttitle{\mm \X{<117>}\<83>}\X{<95>}\<111>}\X{<F<59>}\X{<73>}\X{<179>}}%
6334 %%
6335 \def\stctitle{\mm \X{<68>}\X{<201>}\X{<83>}\X{<183>}\<119>}%
6336 \def\slftitle{\mm \X{<78>}\<111>}\X{<C<94>}\X{<186>}\X{<179>}}%
6337 \def\sltttitle{\mm \X{<117>}\<83>}\X{<95>}\<111>}\X{<F<59>}\X{<73>}\X{<179>}}%
6338 (/malayalam – rachana)

```

13.111 “Malayalam-rachana2” language: malayalam-rachana2.mld

The titles for the “malayalam” language, with the reformed “Rachana” fonts (new lipi), are taken from the malayalam package [3]. This language requires specific fonts. See also sections 13.108 on page 469 and 13.110 on the preceding page.

```

6339 (*malayalam – rachana2)
6340 \ProvidesFile{malayalam-rachana2.mld}[2006/01/13]
6341 %%
6342 %% Malayalam: Rachana fonts, reformed.
6343 %%
6344 \def\ptctitle{\mm \X{<68>}\X{<201>}\X{<83>}\X{<183>}\<119>}%
6345 \def\plftitle{\mm \X{<78>}\<111>}\X{<125>}\<88>}\X{<186>}\X{<179>}}%
6346 \def\pltttitle{\mm \X{<117>}\<83>}\X{<95>}\<111>}\X{<106>}\<113>}\X{<73>}\X{<179>}}%
6347 %%
6348 \def\mtctitle{\mm \X{<68>}\X{<201>}\X{<83>}\X{<183>}\<119>}%
6349 \def\mlftitle{\mm \X{<78>}\<111>}\X{<125>}\<88>}\X{<186>}\X{<179>}}%
6350 \def\mltttitle{\mm \X{<117>}\<83>}\X{<95>}\<111>}\X{<106>}\<113>}\X{<73>}\X{<179>}}%
6351 %%
6352 \def\stctitle{\mm \X{<68>}\X{<201>}\X{<83>}\X{<183>}\<119>}%
6353 \def\slftitle{\mm \X{<78>}\<111>}\X{<125>}\<88>}\X{<186>}\X{<179>}}%
6354 \def\sltttitle{\mm \X{<117>}\<83>}\X{<95>}\<111>}\X{<106>}\<113>}\X{<73>}\X{<179>}}%
6355 (/malayalam – rachana2)

```

13.112 “Manju” language: manju.mld

“Manju” is a synonym for “bithe”, so we just load bithe.mld (see section 13.22 on page 426):

```
6356 <*manju>
6357 \ProvidesFile{manju.mld}[2005/11/16]
6358 \mtcselectlanguage{bithe}%
6359 </manju>
```

13.113 “Meyalu” language: meyalu.mld

The “meyalu” language is just like “bahasam”, so we just load bahasam.mld (see section 13.16 on page 422):

```
6360 <*meyalu>
6361 \ProvidesFile{meyalu.mld}[2006/01/13]
6362 \mtcselectlanguage{bahasam}%
6363 </meyalu>
```

13.114 “Mongol” language: mongol.mld

The titles for the “mongol” language are taken from the MonTeX package [87, 89]. This language requires specific fonts. See also sections 13.19 on page 424, 13.20 on page 424, 13.21 on page 425, 13.22 on page 426, 13.29 on page 429, 13.30 on page 430, 13.169 on page 500, 13.170 on page 500, and 13.171 on page 501.

```
6364 <*mongol>
6365 \ProvidesFile{mongol.mld}[1999/03/16]
6366 %% Mongol (xalx) titles for minitoc.sty
6367 %% Needs mongol fonts
6368 \def\ptctitle{{\mnr Garqig}}%
6369 \def\plftitle{{\mnr Zurgi"in jagsaalt}}%
6370 \def\plttitle{{\mnr X"usn"agti"in jagsaalt}}%
6371 %%
6372 \def\mtctitle{{\mnr Garqig}}%
6373 \def\mlftitle{{\mnr Zurgi"in jagsaalt}}%
6374 \def\mlttitle{{\mnr X"usn"agti"in jagsaalt}}%
6375 %%
6376 \def\stctitle{{\mnr Garqig}}%
6377 \def\slftitle{{\mnr Zurgi"in jagsaalt}}%
6378 \def\slttitle{{\mnr X"usn"agti"in jagsaalt}}%
6379 </mongol>
```


13.115 “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 load `ngermanb.mld`. See also section 13.118.

```
6380 < *naustrian>
6381 \ProvidesFile{naustrian.mld}[2004/12/14]
6382 \mtcselectlanguage{ngermanb}%
6383 < /naustrian>
```

13.116 “Newzealand” language: `newzealand.mld`

The “newzealand” language is just like “english”, so we just load `english.mld` (see section 13.42 on page 435):

```
6384 < *newzealand>
6385 \ProvidesFile{newzealand.mld}[2006/01/11]
6386 \mtcselectlanguage{english}%
6387 < /newzealand>
```

13.117 “Ngerman” language: `ngerman.mld`

The “ngerman” language is a synonym of the “ngermanb” language³⁵, so we load `ngermanb.mld`. See also section 13.118.

```
6388 < *ngerman>
6389 \ProvidesFile{ngerman.mld}[2004/12/14]
6390 \mtcselectlanguage{ngermanb}%
6391 < /ngerman>
```

13.118 “Ngermanb” language: `ngermanb.mld`

The titles for the “ngermanb” language³⁶ are taken from the file `ngermanb.dtx` file in the `babel` package [30, 31, 66]. See also sections 13.115, and 13.117.

³⁵A revised version of the `germanb` variant of the german language.

³⁶A variant of the german language, with revised spelling.

```

6392 <*ngermanb>
6393 \ProvidesFile{ngermanb.mld}[2006/01/13]
6394 %% New german (B) titles for minitoc.sty
6395 %% from ngermanb.dtx (babel)
6396 %% Raichle, Bernd and Schmidt, Walter
6397 \def\ptctitle{Inhaltsverzeichnis}%      % oder nur: Inhalt
6398 \def\plftitle{Abbildungsverzeichnis}%
6399 \def\plttitle{Tabellenverzeichnis}%
6400 %%
6401 \def\mtctitle{Inhalt}%
6402 \def\mlftitle{Abbildungsverzeichnis}%
6403 \def\mlttitle{Tabellenverzeichnis}%
6404 %%
6405 \def\stctitle{Inhalt}%
6406 \def\slftitle{Abbildungsverzeichnis}%
6407 \def\slttitle{Tabellenverzeichnis}%
6408 </ngermanb>

```

13.119 “Ngermanb2” language: ngermanb2.mld

The titles for the “ngermanb2” language³⁷ are taken from the babel package [30, 31, 66]. See also section 13.118 on the page before.

```

6409 <*ngermanb2>
6410 \ProvidesFile{ngermanb2.mld}[2005/09/27]
6411 %% New german (B) short (2) titles for minitoc.sty
6412 \def\ptctitle{Inhalt}%
6413 \def\plftitle{Abbildungen}%
6414 \def\plttitle{Tabellen}%
6415 %%
6416 \def\mtctitle{Inhalt}%
6417 \def\mlftitle{Abbildungen}%
6418 \def\mlttitle{Tabellen}%
6419 %%
6420 \def\stctitle{Inhalt}%
6421 \def\slftitle{Abbildungen}%
6422 \def\slttitle{Tabellen}%
6423 </ngermanb2>

```

³⁷ A variant of the german language, with revised spelling and short titles.

13.120 “Norsk” language: norsk.mld

The titles for the “norsk” language (or bokmål, language of the kingdom) are taken from the norsk.dtx file in the babel package [30, 31, 35], with help from Dag LANGMYHR. See also section 13.122 on the following page.

```

6424 ⟨*norsk⟩
6425 \ProvidesFile{norsk.mld}[2006/01/13]
6426 %% Norsk titles for minitoc.sty
6427 %% from noesk.dtx (babel)
6428 %% Braams, Johannes and Helstrup, Haavard and Holme, Alv Kjetil and
6429 %% Iversen, Per Steinar and Petterst, Terje Engeset and Kleveland, Rune
6430 %% Thanks to Dag Langmyhr (dag@ifi.uio.no)
6431 \def\ptctitle{Innhold}%
6432 \def\plftitle{Figurer}%
6433 \def\plttitle{Tabeller}%
6434 %%
6435 \def\mtctitle{Innhold}%
6436 \def\mlftitle{Figurer}%
6437 \def\mlttitle{Tabeller}%
6438 %%
6439 \def\stctitle{Innhold}%
6440 \def\slftitle{Figurer}%
6441 \def\slttitle{Tabeller}%
6442 ⟨/norsk⟩

```

13.121 “Norsk2” language: norsk2.mld

The titles for the “norsk2” language (or bokmål, language of the kingdom) are taken from the babel package [30, 31, 35], with help from Dag LANGMYHR. These are variants of the titles of section 13.120.

```

6443 ⟨*norsk2⟩
6444 \ProvidesFile{norsk2.mld}[2005/09/27]
6445 %% Short norsk titles for minitoc.sty
6446 %% Thanks to Dag Langmyhr (dag@ifi.uio.no)
6447 \def\ptctitle{Innhold}%
6448 \def\plftitle{Figurliste}%
6449 \def\plttitle{Tabelliste}%
6450 %%
6451 \def\mtctitle{Innhold}%
6452 \def\mlftitle{Figurliste}%
6453 \def\mlttitle{Tabelliste}%
6454 %%
6455 \def\stctitle{Innhold}%
6456 \def\slftitle{Figurliste}%
6457 \def\slttitle{Tabelliste}%
6458 ⟨/norsk2⟩

```

13.122 “Nynorsk” language: nynorsk.mld

The titles for the “nynorsk” language³⁸ are taken from norsk.dtx file in the babel package [30, 31, 35], with help from Dag LANGMYHR. See also section 13.120 on the preceding page.

```

6459 <*nynorsk>
6460 \ProvidesFile{nynorsk.mld}[2006/01/13]
6461 %% Nynorsk titles for minitoc.sty
6462 %% from norsk.dtx (babel)
6463 %% Braams, Johannes and Helstrup, Haavard and Holme, Alv Kjetil and
6464 %% Iversen, Per Steinar and Petterst, Terje Engeset and Kleveland, Rune
6465 %% Thanks to Dag Langmyhr (dag@ifi.uio.no)
6466 \def\mtctitle{Innhald}%
6467 \def\mlftitle{Figurar}%
6468 \def\mltttitle{Tabellar}%
6469 %%
6470 \def\ptctitle{Innhald}%
6471 \def\plftitle{Figurar}%
6472 \def\pltttitle{Tabellar}%
6473 %%
6474 \def\stctitle{Innhald}%
6475 \def\slftitle{Figurar}%
6476 \def\sltttitle{Tabellar}%
6477 </nynorsk>

```

13.123 “Nynorsk2” language: nynorsk2.mld

The titles for the “nynorsk2” language are variants of the titles of the “nynorsk” language. See also section 13.122.

```

6478 <*nynorsk2>
6479 \ProvidesFile{nynorsk.mld}[1999/03/16]
6480 %% Nynorsk titles for minitoc.sty
6481 %% Thanks to Dag Langmyhr (dag@ifi.uio.no)
6482 \def\mtctitle{Innhald}%
6483 \def\mlftitle{Figurliste}%
6484 \def\mltttitle{Tabelliste}%
6485 %%
6486 \def\ptctitle{Innhald}%
6487 \def\plftitle{Figurliste}%
6488 \def\pltttitle{Tabelliste}%
6489 %%
6490 \def\stctitle{Innhald}%
6491 \def\slftitle{Figurliste}%

```

³⁸Created around 1800 by Ivar ÅSSEN to make a real independent and national norwegian language, in reaction to danish, from the various dialect spoken in the country.

```
6492 \def\sltttitle{Tabelliste}%
6493 </nynorsk2>
```

13.124 “Polish” language: polish.mld

The titles for the “polish” language are taken from the polish.dtx file in the babel package [30, 31, 70]. See also sections 13.125 and 13.126 on the next page.

```
6494 <polish>
6495 \ProvidesFile{polish.mld}[2006/01/13]
6496 %% Polish titles for minitoc.sty
6497 %% from polish.dtx (babel)
6498 %% Schalueck, Elmar and Janich, Michael
6499 \def\ptctitle{Spis rzeczy}%
6500 \def\plfttitle{Spis rysunk\'ow}%
6501 \def\pltttitle{Spis tablic}%
6502 %%
6503 \def\mtctitle{Spis rzeczy}%
6504 \def\mlfttitle{Spis rysunk\'ow}%
6505 \def\mltttitle{Spis tablic}%
6506 %%
6507 \def\stctitle{Spis rzeczy}%
6508 \def\slfttitle{Spis rysunk\'ow}%
6509 \def\sltttitle{Spis tablic}%
6510 </polish>
```

13.125 “Polish2” language: polish2.mld

The titles for the “polish2” language³⁹ are taken from the ANTOMEGA project [127]. See also sections 13.124 and 13.126 on the next page.

```
6511 <polish2>
6512 \ProvidesFile{polish2.mld}[2005/02/08]
6513 %% from omega-polish.ldf (Antomega project)
6514 %% Needs Omega
6515 %% Alexej M. Kryokov
6516 %% Dmitry Ivanov
6517 %%
6518 \def\ptctitle{\localpolish{Spis tre^^^^00b1ci}}%
6519 \def\plfttitle{\localpolish{Spis rysunk^^^^00adw}}%
6520 \def\pltttitle{\localpolish{Spis tablic}}%
6521 %%
6522 \def\mtctitle{\localpolish{Spis tre^^^^00b1ci}}%
6523 \def\mlfttitle{\localpolish{Spis rysunk^^^^00adw}}%
```

³⁹ “Polish2” is a variant of “polish”.

```

6524 \def\mltttitle{\localpolish{Spis tablic}}%
6525 %%
6526 \def\stcttitle{\localpolish{Spis tre^^^^00b1ci}}%
6527 \def\slfttitle{\localpolish{Spis rysunk^^^^00adw}}%
6528 \def\slttitle{\localpolish{Spis tablic}}%
6529 </polish2>

```

13.126 “Polski” language: polski.mld

The titles for the “polski” language (variant for polish) are taken from the `polski.dtx` file in the \LaTeX package [168, 204]. See also sections 13.124 on the preceding page and 13.125 on the page before.

```

6530 <{*polski}>
6531 \ProvidesFile{polski.mld}[2006/02/28]
6532 %% Polski titles for minitoc.sty
6533 %% from polski.dtx (PLaTeX)
6534 %% Mariusz Olko, Marcin Woli'nski.
6535 %%
6536 \def\ptcttitle{Spis tre\'sci}%
6537 \def\plfttitle{Spis rysunk\'ow}%
6538 \def\pltttitle{Spis tabel}%
6539 %%
6540 \def\mtcttitle{Spis rysunk\'ow}%
6541 \def\mlfttitle{Spis rysunk\'ow}%
6542 \def\mltttitle{Spis tabel}%
6543 %%
6544 \def\stcttitle{Spis rysunk\'ow}%
6545 \def\slfttitle{Spis rysunk\'ow}%
6546 \def\slttitle{Spis tabel}%
6547 </polski>

```

13.127 “Portuges” language: portuges.mld

This is another spelling for “portuguese” (see section 13.128 on the following page), so we just load `portuguese.mld`:

```

6548 <{*portuges}>
6549 \ProvidesFile{portuges.mld}[2005/06/07]
6550 \mtcselectlanguage{portuguese}%
6551 </portuges>

```

13.128 “Portuguese” language: `portuguese.mld`

The titles for the “portuguese” language are taken from the `portuges.dtx` file in the `babel` package [30, 31, 67]. See also section 13.23 on page 426, because the titles are different in Brazil, even if the language is also portuguese.

```

6552 (*portuguese)
6553 \ProvidesFile{portuguese.mld}[2006/01/13]
6554 %% Portuguese titles for minitoc.sty
6555 %% from portuges.dtx (babel)
6556 %% Ramalhete, Jose Pedro
6557 \def\ptctitle{Conte\'udo}%
6558 \def\plftitle{Lista de Figuras}%
6559 \def\pltttitle{Lista de Tabelas}%
6560 %%
6561 \def\mtctitle{Conte\'udo}%
6562 \def\mlftitle{Lista de Figuras}%
6563 \def\mltttitle{Lista de Tabelas}%
6564 %%
6565 \def\stctitle{Conte\'udo}%
6566 \def\slftitle{Lista de Figuras}%
6567 \def\slttitle{Lista de Tabelas}%
6568 \end{portuguese}

```

13.129 “Romanian” language: `romanian.mld`

The titles for the “romanian” language are taken from the `romanian.dtx` file in the `babel` package [30, 31, 52].

```

6569 (*romanian)
6570 \ProvidesFile{romanian.mld}[2006/01/13]
6571 %% Romanian titles for minitoc.sty
6572 %% from romanian.dtx (babel)
6573 %% Horst, Umstatter and Juhasz, Robert
6574 \def\ptctitle{Cuprins}%
6575 \def\plftitle{List\u{a} de figuri}%
6576 \def\pltttitle{List\u{a} de tabele}%
6577 %%
6578 \def\mtctitle{Cuprins}%
6579 \def\mlftitle{List\u{a} de figuri}%
6580 \def\mltttitle{List\u{a} de tabele}%
6581 %%
6582 \def\stctitle{Cuprins}%
6583 \def\slftitle{List\u{a} de figuri}%
6584 \def\slttitle{List\u{a} de tabele}%
6585 \end{romanian}

```

13.130 “Romanian2” language: `romanian2.mld`

The titles for the “romanian2” language are taken from the `romanian.dtx` file in the RomanianTeX package [182]. See also sections 13.129 on the preceding page and 13.131. Alas, RomanianTeX is not compatible with the `babel` package [30, 31].

```

6586 \(*romanian2)
6587 \ProvidesFile{romanian2.mld}[2006/08/03]
6588 %% Romanian titles for minitoc.sty
6589 %% from RomanianTeX (romanian.dtx)
6590 %% Adrian Rezus (adriaan@cs.kun.nl)
6591 %% Bernd Raichle
6592 \def\ptctitle{Cuprins}%
6593 \def\plftitle{Lista de figuri}%
6594 \def\plttitle{Lista de tabele}%
6595 %%
6596 \def\mtctitle{Cuprins}%
6597 \def\mlftitle{Lista de figuri}%
6598 \def\mlttitle{Lista de tabele}%
6599 %%
6600 \def\stctitle{Cuprins}%
6601 \def\slftitle{Lista de figuri}%
6602 \def\slttitle{Lista de tabele}%
6603 \(/romanian2)

```

13.131 “Romanian3” language: `romanian3.mld`

The titles for the “romanian3” language are taken from the `romanian.dtx` file in the RomanianTeX package [182]. See also sections 13.129 on the preceding page and 13.130. Alas, RomanianTeX is not compatible with the `babel` package [30, 31].

```

6604 \(*romanian3)
6605 \ProvidesFile{romanian3.mld}[2006/08/03]
6606 %% Romanian titles for minitoc.sty
6607 %% from RomanianTeX (romanian.dtx) variant.
6608 %% Adrian Rezus (adriaan@cs.kun.nl)
6609 %% Bernd Raichle
6610 \def\ptctitle{Tabla de materii}%
6611 \def\plftitle{Indice de figuri}%
6612 \def\plttitle{Tabele}%
6613 %%
6614 \def\mtctitle{Tabla de materii}%
6615 \def\mlttitle{Indice de figuri}%
6616 \def\mlttitle{Tabele}%
6617 %%
6618 \def\stctitle{Tabla de materii}%
6619 \def\slttitle{Indice de figuri}%
6620 \def\slttitle{Tabele}%

```



```
6621 </romanian3>
```

13.132 “Russian” language: russian.mld

The titles for the “russian” language are taken from the babel package [30, 31]. Specific cyrillic fonts are required.

```
6622 <*russian>
6623 \ProvidesFile{russian.mld}[1999/03/16]
6624 %% Russian titles for minitoc.sty
6625 \def\ptctitle{Oglavlenie}%
6626 \def\plftitle{Pere{\cz}en{\mz} risunkov}%
6627 \def\plttitle{Pere{\cz}en{\mz} tablic}%
6628 %%
6629 \def\mtctitle{Oglavlenie}%
6630 \def\mlftitle{Pere{\cz}en{\mz} risunkov}%
6631 \def\mlttitle{Pere{\cz}en{\mz} tablic}%
6632 %%
6633 \def\stctitle{Oglavlenie}%
6634 \def\slftitle{Pere{\cz}en{\mz} risunkov}%
6635 \def\slttitle{Pere{\cz}en{\mz} tablic}%
6636 </russian>
```

13.133 “Russian2m” language: russian2m.mld

The titles for the “russian2m” language (“russian2m” is a modern variant of “russian”) are taken from the ANTOMEGA project [127]. Specific cyrillic fonts are required. See also section 13.132.

```
6637 <*russian2m>
6638 \ProvidesFile{russian2m.mld}[2005/02/08]
6639 %% from russian2m.ldf (Antomega project, russian modern)
6640 %% Needs Omega and cyrillic fonts
6641 %% Alexej M. Kryokov
6642 %% Dmitry Ivanov
6643 %%
6644 \def\ptctitle{\localrussian%
6645 {^041e^0433^043b^0430^0432^043b^0435^043d%
6646 ^0438^0435}}%
6647 \def\plftitle{\localrussian%
6648 {^0421^043f^0438^0441^043e^043a ^0438^043b%
6649 ^043b^044e^0441^0442^0440^0430^0446^0438%
6650 ^0439}}%
6651 \def\plttitle{\localrussian%
6652 {^0421^043f^0438^0441^043e^043a ^0442^0430%
```

```

6653  ^^^^0431^^^^043b^^^^0438^^^^0446}}}%
6654 %%
6655 \def\mtctitle{\localrussian%
6656 {^^^^041e^^^^0433^^^^043b^^^^0430^^^^0432^^^^043b^^^^0435^^^^043d%
6657  ^^^^0438^^^^0435}}}%
6658 \def\mlftitle{\localrussian%
6659 {^^^^0421^^^^043f^^^^0438^^^^0441^^^^043e^^^^043a  ^^^^0438^^^^043b%
6660  ^^^^043b^^^^044e^^^^0441^^^^0442^^^^0440^^^^0430^^^^0446^^^^0438%
6661  ^^^^0439}}}%
6662 \def\mltttitle{\localrussian%
6663 {^^^^0421^^^^043f^^^^0438^^^^0441^^^^043e^^^^043a  ^^^^0442^^^^0430%
6664  ^^^^0431^^^^043b^^^^0438^^^^0446}}}%
6665 %%
6666 \def\stctitle{\localrussian%
6667 {^^^^041e^^^^0433^^^^043b^^^^0430^^^^0432^^^^043b^^^^0435^^^^043d%
6668  ^^^^0438^^^^0435}}}%
6669 \def\slftitle{\localrussian%
6670 {^^^^0421^^^^043f^^^^0438^^^^0441^^^^043e^^^^043a  ^^^^0438^^^^043b%
6671  ^^^^043b^^^^044e^^^^0441^^^^0442^^^^0440^^^^0430^^^^0446^^^^0438%
6672  ^^^^0439}}}%
6673 \def\sltttitle{\localrussian%
6674 {^^^^0421^^^^043f^^^^0438^^^^0441^^^^043e^^^^043a  ^^^^0442^^^^0430%
6675  ^^^^0431^^^^043b^^^^0438^^^^0446}}}%
6676 </russian2m>

```

13.134 “Russian2o” language: russian2o.mld

The titles for the “russian2o” language (“russian2o” is an old variant of “russian”) are taken from the ANTOMEGA project [127]. Specific cyrillic fonts are required. See also section 13.132 on the page before.

```

6677 <*russian2o>
6678 \ProvidesFile{russian2o.mld}[2005/02/08]
6679 %% from russian2o.mld (Antomega project - russian old)
6680 %% Needs Omega and cyrillic fonts
6681 %% Alexej M. Kryokov
6682 %% Dmitry Ivanov
6683 %%
6684 \def\ptctitle{\localrussian%
6685 {^^^^041e^^^^0433^^^^043b^^^^0430^^^^0432^^^^043b^^^^0435^^^^043d%
6686  ^^^^0456^^^^0435}}}%
6687 \def\plftitle{\localrussian%
6688 {^^^^0421^^^^043f^^^^0438^^^^0441^^^^043e^^^^043a^^^^044a  ^^^^0438%
6689  ^^^^043b^^^^043b^^^^044e^^^^0441^^^^0442^^^^0440^^^^0430^^^^0446%
6690  ^^^^0456^^^^0439}}}%
6691 \def\pltttitle{\localrussian%
6692 {^^^^0421^^^^043f^^^^0438^^^^0441^^^^043e^^^^043a^^^^044a  ^^^^0442%
6693  ^^^^0430^^^^0431^^^^043b^^^^0438^^^^0446^^^^044a}}}%
6694 %%
6695 \def\mtctitle{\localrussian%

```

```

6696 {^041e^0433^043b^0430^0432^043b^0435^043d%
6697 ^0456^0435}}%
6698 \def\mlftitle{\localrussian%
6699 {^0421^043f^0438^0441^043e^043a^044a ^0438%
6700 ^043b^043b^044e^0441^0442^0440^0430^0446%
6701 ^0456^0439}}%
6702 \def\mltttitle{\localrussian%
6703 {^0421^043f^0438^0441^043e^043a^044a ^0442%
6704 ^0430^0431^043b^0438^0446^044a}}%
6705 %
6706 \def\stctitle{\localrussian%
6707 {^041e^0433^043b^0430^0432^043b^0435^043d%
6708 ^0456^0435}}%
6709 \def\slftitle{\localrussian%
6710 {^0421^043f^0438^0441^043e^043a^044a ^0438%
6711 ^043b^043b^044e^0441^0442^0440^0430^0446%
6712 ^0456^0439}}%
6713 \def\sltttitle{\localrussian%
6714 {^0421^043f^0438^0441^043e^043a^044a ^0442%
6715 ^0430^0431^043b^0438^0446^044a}}%
6716 </russian2o>

```

13.135 “Russianb” language: russianb.mld

The titles for the “russianb” language (“russianb” is a variant of “russian”) are taken from the russianb.dtx file in the babel package [30, 31, 58, 136]. Specific cyrillic fonts are required. See also section 13.132 on page 481. The parttoc title varies depending if chapters are defined or not by the document class.

```

6717 <{*russianb>
6718 \ProvidesFile{russianb.mld}[2006/02/15]
6719 %% Russian (russianb) titles for minitoc.sty
6720 %% from russianb.dtx (babel)
6721 %% Lapko, Olga and Volovitch, Vladimir and Lemberg, Werner
6722 \expandafter\ifx\csname chapter\endcsname\relax
6723 \def\ptctitle{%
6724   {\cyr\CYRS\cyro\cyrd\cyre\cyrr\cyrzh\cyra\cyrn\cyri\cyre}}%
6725 \else
6726 \def\ptctitle{%
6727   {\cyr \CYRO\CYRg\CYRl\CYRa\CYRv\CYRl\CYRe\CYRn\CYRi\CYRe}}%
6728 \fi
6729 \def\plftitle{%
6730   {\cyr \CYRS\CYRp\CYRi\CYRs\CYRo\CYRk\space
6731     \CYRi\CYRl\CYRl\CYRyu\CYRs\CYRt\CYRr\CYRa\CYRc\CYRi\CYRishrt}}%
6732 \def\pltttitle{%
6733   {\cyr \CYRS\CYRp\CYRi\CYRs\CYRo\CYRk\space
6734     \CYRt\CYRa\CYRb\CYRl\CYRi\CYRc}}%
6735 %%
6736 \def\mtctitle{%
6737   {\cyr \CYRO\CYRg\CYRl\CYRa\CYRv\CYRl\CYRe\CYRn\CYRi\CYRe}}%

```

```

6738 \def\mlfttitle{%
6739   {\cyr \CYRS\CYRp\CYRi\CYRs\CYRo\CYRk\space
6740     \CYRi\CYRl\CYRl\CYRyu\CYRs\CYRt\CYRr\CYRa\CYRc\CYRi\CYRishrt}}%
6741 \def\mltttitle{%
6742   {\cyr \CYRS\CYRp\CYRi\CYRs\CYRo\CYRk\space
6743     \CYRt\CYRa\CYRb\CYRl\CYRi\CYRc}}%
6744 %%
6745 \def\stctitle{%
6746   {\cyr \CYRO\CYRg\CYRl\CYRa\CYRv\CYRl\CYRe\CYRn\CYRi\CYRe}}%
6747 \def\slfttitle{%
6748   {\cyr \CYRS\CYRp\CYRi\CYRs\CYRo\CYRk\space
6749     \CYRi\CYRl\CYRl\CYRyu\CYRs\CYRt\CYRr\CYRa\CYRc\CYRi\CYRishrt}}%
6750 \def\slttitle{%
6751   {\cyr \CYRS\CYRp\CYRi\CYRs\CYRo\CYRk\space
6752     \CYRt\CYRa\CYRb\CYRl\CYRi\CYRc}}%
6753 </russianb>

```

13.136 “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 [87, 89]. Specific cyrillic fonts are required. See also section 13.132 on page 481.

```

6754 <*russianc>
6755 \ProvidesFile{russianc.mld}[1999/03/16]
6756 %% Russian titles for minitoc.sty
6757 %% Needs cyrillic fonts
6758 \def\ptctitle{\xalx{Oglawlenie}}%
6759 \def\plfttitle{\xalx{Spisok risunkow}}%
6760 \def\pltttitle{\xalx{Spisok tablic}}%
6761 %%
6762 \def\mtctitle{\xalx{Soderjanie}}%
6763 \def\mlfttitle{\xalx{Spisok risunkow}}%
6764 \def\mltttitle{\xalx{Spisok tablic}}%
6765 %%
6766 \def\stctitle{\xalx{Soderjanie}}%
6767 \def\slfttitle{\xalx{Spisok risunkow}}%
6768 \def\slttitle{\xalx{Spisok tablic}}%
6769 </russianc>

```

13.137 “Russian-cca” language: `russian-cca.mld` and `russian-cca.mlo`

They are several variants for the russian titles with the cmcyralt fonts. The titles for a first variant of the “russian-cca” are taken from the `russian.sty` file in the cmcyralt package [28].

The titles for the “russian-cca” language contain characters that cannot be easily generated, hence we load `russian-cca.mlo`.

```
6770 <*russian – cca>
6771 \ProvidesFile{russian-cca.mld}[2006/03/08]
6772 %% Russian-cca titles for minitoc.sty
6773 %% From russian.sty in the cmcyralt package
6774 %% Vadim Maslov (vadik@cs.umd.edu)
6775 %% Victor Boyko (vb1890@cs.nyu.edu)
6776 %% Needs cmcyralt fonts and special input encoding.
6777 \mtcloadmlo{russian-cca}%
6778 </russian – cca>
```

13.138 “Russian-cca1” language: `russian-cca1.mld` and `russian-cca1.mlo`

They are several variants for the russian titles with the cmcyralt fonts. The titles for the “russian-cca1” language are taken from the `cmcyralt.sty` file in the cmcyralt package [112].

The titles for the “russian-cca1” language contain characters that cannot be easily generated, hence we load `russian-cca1.mlo`.

```
6779 <*russian – cca1>
6780 \ProvidesFile{russian-cca1.mld}[2006/03/08]
6781 %% Russian-cca1 titles for minitoc.sty
6782 %% From cmcyralt.sty in the cmcyralt package
6783 %% cmcyr fonts in alt encoding
6784 %% Vadim Maslov (vadik@cs.umd.edu)
6785 %% Alexander Harin (harin@lourie.und.ac.za)
6786 %% and Vadim V. Zhytnikov (vvzhy@phy.ncu.edu.tw)
6787 \mtcloadmlo{russian-cca1}%
6788 </russian – cca1>
```

13.139 “Russian-lh” language: `russian-lh.mld` and `russian-lh.mlo`

The russian titles for the LH fonts (“russian-lh” language) are taken from the `russian.sty` file in the LH package [163].

The titles for the “russian-lh” language contain characters that cannot be easily generated, hence we load `russian-lh.mlo`.

```
6789 < *russian – lh>
6790 \ProvidesFile{russian-lh.mld}[2006/03/08]
6791 %% Russian-lh titles for minitoc.sty
6792 %% From russian.sty in the LH package
6793 %% LH fonts in special encoding
6794 %% Sergei O. Naoumov (serge@astro.unc.edu)
6795 \mtcloadmlo{russian-lh}%
6796 < /russian – lh>
```

13.140 “Russian-lhcyralt” language: `russian-lhcyralt.mld` and `russian-lhcyralt.mlo`

The russian titles for the LHCYRALT fonts (“russian-lhcyralt” language) are taken from the `lhcyralt.sty` file in the lhcyr package [217].

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).

```
6797 < *russian – lhcyralt>
6798 \ProvidesFile{russian-lhcyralt.mld}[2006/03/10]
6799 %% Russian-lhcyralt titles for minitoc.sty
6800 %% From lhcyralt.sty in the LHCYR package
6801 %% LHCYRALT fonts in special encoding ALT (CP866)
6802 %% Vadim V. Zhytnikov (vvzhy@td.lpi.ac.ru)
6803 \mtcloadmlo{russian-lhcyralt}%
6804 < /russian – lhcyralt>
```

13.141 “Russian-lhcyrkoi” language: russian-lhcyrkoi.mld and russian-lhcyrkoi.mlo

The russian titles for the LHCYRKOI fonts (“russian-lhcyrkoi” language) are taken from the lhcyrkoi.sty file in the lhcyr package [217].

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.

```
6805 ⟨*russian – lhcyrkoi⟩
6806 \ProvidesFile{russian-lhcyrkoi.mld}[2006/03/13]
6807 %% Russian-lhcyrkoi titles for minitoc.sty
6808 %% From lhcyrkoi.sty in the LHCYR package
6809 %% LHCYRKOI fonts in special encoding KOI-8
6810 %% Vadim V. Zhytnikov (vvzhy@td.lpi.ac.ru)
6811 \mtcloadmlo{russian-lhcyrkoi}%
6812 ⟨/russian – lhcyrkoi⟩
```

13.142 “Russian-lhcyrwin” language: russian-lhcyrwin.mld and russian-lhcyrwin.mlo

The russian titles for the LHCYRWIN fonts (“russian-lhcyrwin” language) are taken from the lhcyrwin.sty file in the lhcyr package [217].

The titles for the “russian-lhcyrwin” language contain characters that cannot be easily generated, hence we load russian-lhcyrwin.mlo. The input encoding is the code page CP1251.

```
6813 ⟨*russian – lhcyrwin⟩
6814 \ProvidesFile{russian-lhcyrwin.mld}[2006/03/13]
6815 %% Russian-lhcyrwin titles for minitoc.sty
6816 %% From lhcyrwin.sty in the LHCYR package
6817 %% LHCYRWIN fonts in special encoding CP1251
6818 %% Vadim V. Zhytnikov (vvzhy@td.lpi.ac.ru)
6819 \mtcloadmlo{russian-lhcyrwin}%
6820 ⟨/russian – lhcyrwin⟩
```

13.143 “Samin” language: `samin.mld`

The titles for the “samin” language⁴⁰ are taken from the `samin.dtx` file in the `babel` package [30, 31, 53]. Specific fonts are required.

```

6821 <*samin>
6822 \ProvidesFile{samin.mld}[2006/01/13]
6823 %% North Sami (samin) titles for minitoc.sty
6824 %% from samin.dtx (babel)
6825 %% Jernsletten, Regnor
6826 \def\ptctitle{Sisdoallu}%
6827 \def\plftitle{Govvosat}%
6828 \def\plttitle{Tabeallat}%
6829 %%
6830 \def\mtctitle{Sisdoallu}%
6831 \def\mlftitle{Govvosat}%
6832 \def\mlttitle{Tabeallat}%
6833 %%
6834 \def\stctitle{Sisdoallu}%
6835 \def\slftitle{Govvosat}%
6836 \def\slttitle{Tabeallat}%
6837 </samin>

```

13.144 “Scottish” language: `scottish.mld`

The titles for the “scottish” language are taken from the `scottish.dtx` file in the `babel` language [30, 31, 50]:

```

6838 <*scottish>
6839 \ProvidesFile{scottish.mld}[2006/01/13]
6840 %% Scottish titles for minitoc.sty
6841 %% from scottish.dtx (babel)
6842 %% Grant, Fraser
6843 \def\ptctitle{Cl\‘ar-obraich}%
6844 \def\plftitle{LiostaDhealbh}%
6845 \def\plttitle{LiostaChl\‘ar}%
6846 %%
6847 \def\mtctitle{Cl\‘ar-obraich}%
6848 \def\mlftitle{LiostaDhealbh}%
6849 \def\mlttitle{LiostaChl\‘ar}%
6850 %%
6851 \def\stctitle{Cl\‘ar-obraich}%
6852 \def\slftitle{LiostaDhealbh}%
6853 \def\slttitle{LiostaChl\‘ar}%

```

⁴⁰Several Sami 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 Sami. Hence the name `samin.dtx` (and not `sami.dtx` or the like) in the North Sami case. This note is copied from the `samin.dtx` file.


```
6854 </scottish>
```

13.145 “Serbian” language: `serbian.mld`

The titles for the “serbian” (serbocroatian) language are taken from the `serbian.dtx` file in the `babel` package [30, 31, 63]. Serbocroatian is spoken by Serbs, Croats and Chernogors, but only Serbs and Chernogors use the cyrillic alphabet (a variant). See also section 13.146.

```
6855 <*serbian>
6856 \ProvidesFile{serbian.mld}[2006/01/13]
6857 %% Serbian titles for minitoc.sty
6858 %% from serbian.dtx (babel)
6859 %% Muhamedagi\`{c}, Dejan and Slobodan, Jankovic
6860 \def\ptctitle{Sadr\v{z}aj}%
6861 \def\plftitle{Slike}%
6862 \def\pltttitle{Tabele}%
6863 %%
6864 \def\mtctitle{Sadr\v{z}aj}%
6865 \def\mlftitle{Slike}%
6866 \def\mltttitle{Tabele}%
6867 %%
6868 \def\stctitle{Sadr\v{z}aj}%
6869 \def\slftitle{Slike}%
6870 \def\sltttitle{Tabele}%
6871 </serbian>
```

13.146 “Serbianc” language: `serbianc.mld`

The titles for the “serbianc” language⁴¹ have been gently provided by Marko ÈEHAJA 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 13.145.

```
6872 <*serbianc>
6873 \ProvidesFile{serbianc.mld}[2006/01/13]
6874 %% Provides titles for minitoc.sty in Serbian Cyrillic
6875 %%
6876 %% Marko Èehaja Internut@Thetaworld.Org
6877 %% Frank Küster, Biozentrum der Univ. Basel, frank@kuesterei.ch
6878 %% Abt. Biophysikalische Chemie
6879 \def\ptctitle{%
6880   {\cyr\CYRS\cyra\cyrd\cyrr\cyrzh\cyra\cyrje}}%
```

⁴¹ The “serbianc” language is written with cyrillic characters.

```

6881 \def\plftitle{%
6882   {\cyr\CYRS\cyrl\cyri\cyrk\cyre}}%
6883 \def\plttitle{\CYRT\cyra\cyrb\cyrl\cyri\cyrc\cyre}%
6884 %%
6885 \def\mtctitle{%
6886   {\cyr\CYRS\cyra\cyrd\cyrr\cyrzh\cyra\cyrje}}%
6887 \def\mlftitle{%
6888   {\cyr\CYRS\cyrl\cyri\cyrk\cyre}}%
6889 \def\mlttitle{\cyr\CYRT\cyra\cyrb\cyrl\cyri\cyrc\cyre}%
6890 %%
6891 \def\stctitle{%
6892   {\cyr\CYRS\cyra\cyrd\cyrr\cyrzh\cyra\cyrje}}%
6893 \def\slftitle{%
6894   {\cyr\CYRS\cyrl\cyri\cyrk\cyre}}%
6895 \def\slttitle{\CYRT\cyra\cyrb\cyrl\cyri\cyrc\cyre}%
6896 </serbianc>

```

13.147 “Slovak” language: slovak.mld

The titles for the “slovak” language are taken from the slovak.dtx file in the babel package [30, 31, 46]. Using the T1 font encoding is recommended.

```

6897 <*slovak>
6898 \ProvidesFile{slovak.mld}[2006/01/13]
6899 %% Slovak titles for minitoc.sty
6900 %% from slovak.dtx (babel)
6901 %% Chlebikova, Jana and Schlemmer, Tobias
6902 \def\ptctitle{Obsah}%
6903 \def\plftitle{Zoznam obr\'azkov}%
6904 \def\plttitle{Zoznam tabuliek}%
6905 %%
6906 \def\mtctitle{Obsah}%
6907 \def\mlftitle{Zoznam obr\'azkov}%
6908 \def\mlttitle{Zoznam tabuliek}%
6909 %%
6910 \def\stctitle{Obsah}%
6911 \def\slftitle{Zoznam obr\'azkov}%
6912 \def\slttitle{Zoznam tabuliek}%
6913 </slovak>

```

13.148 “Slovene” language: slovene.mld

The titles for the “slovene” language are taken from the slovene.dtx file in the babel package [30, 31, 76]:

```

6914 <*slovene>
6915 \ProvidesFile{slovene.mld}[2006/01/13]
6916 %% Slovene titles for minitoc.sty
6917 %% from slovene.dtx (babel)
6918 %% Zavrtanik, Danilo and \v{Z}lajpah, Leon
6919 \def\ptctitle{Kazalo}%
6920 \def\plftitle{Slike}%
6921 \def\plttitle{Tabele}%
6922 %%
6923 \def\mtctitle{Kazalo}%
6924 \def\mlftitle{Slike}%
6925 \def\mlttitle{Tabele}%
6926 %%
6927 \def\stctitle{Kazalo}%
6928 \def\slftitle{Slike}%
6929 \def\slttitle{Tabele}%
6930 </slovene>

```

13.149 “Spanish” language: spanish.mld

The titles for the “spanish” language are taken from the `spanish.dtx` file in the `babel` package [30, 31, 25]. Note that the “spanish” is in fact “castillan” (see section 13.33 on page 431). Other languages are spoken in Spain: “basque” (section 13.18 on page 423), “catalan” (section 13.35 on page 431), and “galician” (section 13.63 on page 445). “Spanish2” is a version of “spanish” with shorter titles (see section 13.150 on the following page). “Spanish3” (see section 13.151 on the next page) is a version for the ANTOMEGA [127] project; some titles are different. “Spanish4” is a variant of “spanish” where `\ptctitle` is shorter for articles (section 13.152 on page 493).

```

6931 <*spanish>
6932 \ProvidesFile{spanish.mld}[2006/02/15]
6933 %% Spanish titles for minitoc.sty
6934 %% from spanish.dtx (babel)
6935 %% Bezos, Javier
6936 \expandafter\ifx\csname chapter\endcsname\relax
6937 \def\ptctitle{\'Indice}%
6938 \else
6939 \def\ptctitle{\'Indice General}%
6940 \fi
6941 \def\plftitle{\'Indice de Figuras}%
6942 \def\plttitle{\'Indice de Tablas}%
6943 %%
6944 \def\mtctitle{\'Indice}%
6945 \def\mlftitle{\'Indice de Figuras}%
6946 \def\mlttitle{\'Indice de Tablas}%
6947 %%
6948 \def\stctitle{\'Indice}%
6949 \def\slftitle{\'Indice de Figuras}%
6950 \def\slttitle{\'Indice de Tablas}%

```

```
6951 </spanish>
```

13.150 “Spanish2” language: spanish2.mld

The titles for the “spanish2” language are taken from the spanish.dtx file in the babel package [30, 31, 25], but made shorter for chapter and section levels. See section 13.149 on the page before.

```
6952 <*spanish2>
6953 \ProvidesFile{spanish2.mld}[2005/03/31]
6954 %% Spanish titles for minitoc.sty
6955 \def\ptctitle{\`Indice General}%
6956 \def\plftitle{\`Indice de Figuras}%
6957 \def\plttitle{\`Indice de Tablas}%
6958 %%
6959 \def\mtctitle{Contenido}%
6960 \def\mlftitle{Figuras}%
6961 \def\mltttitle{Tablas}%
6962 %%
6963 \def\stctitle{Contenido}%
6964 \def\slftitle{Figuras}%
6965 \def\slttitle{Tablas}%
6966 </spanish2>
```

13.151 “Spanish3” language: spanish3.mld

The titles for the “spanish3” language are taken from the omega-spanish.ldf file of the ANTOMEGA project [127]. See section 13.149 on the preceding page.

```
6967 <*spanish3>
6968 \ProvidesFile{spanish3.mld}[2005/09/06]
6969 %% Spanish titles for minitoc.sty
6970 %% from omega-spanish.ldf of the ANTOMEGA project.
6971 \def\ptctitle{\localspanish{^^^00cdndice general}}%
6972 \def\plftitle{\localspanish{^^^00cdndice de figuras}}%
6973 \def\plttitle{\localspanish{^^^00cdndice de cuadros}}%
6974 %%
6975 \def\mtctitle{\localspanish{^^^00cdndice general}}%
6976 \def\mlftitle{\localspanish{^^^00cdndice de figuras}}%
6977 \def\mltttitle{\localspanish{^^^00cdndice de cuadros}}%
6978 %%
6979 \def\stctitle{\localspanish{^^^00cdndice general}}%
6980 \def\slftitle{\localspanish{^^^00cdndice de figuras}}%
6981 \def\slttitle{\localspanish{^^^00cdndice de cuadros}}%
6982 </spanish3>
```

13.152 “Spanish4” language: spanish4.mld

The titles for the “spanish4” language are taken from the `spanish.dtx` file from `CervanTeX` [24]. The title of the parttocs is shorter for articles. See also section 13.149 on page 491.

```

6983 <*spanish4>
6984 \ProvidesFile{spanish4.mld}[2006/01/19]
6985 %% Spanish titles for minitoc.sty
6986 %% from spanish.dtx (CervanTeX)
6987 %% Bezos, Javier
6988 \expandafter\ifx\csname chapter\endcsname\relax
6989 \def\ptctitle{\`Indice}%
6990 \else
6991 \def\ptctitle{\`Indice general}%
6992 \fi
6993 \def\plftitle{\`Indice de figuras}%
6994 \def\plttitle{\`Indice de cuadros}%
6995 %%
6996 \def\mtctitle{\`Indice}%
6997 \def\mlftitle{\`Indice de figuras}%
6998 \def\mlttitle{\`Indice de cuadros}%
6999 %%
7000 \def\stctitle{\`Indice}%
7001 \def\slftitle{\`Indice de figuras}%
7002 \def\slttitle{\`Indice de cuadros}%
7003 </spanish4>

```

13.153 “Swedish” language: swedish.mld

The titles for the “swedish” language are taken from the `swedish.dtx` file in the `babel` package [30, 31, 51]. See also section 13.154 on the following page.

```

7004 <*swedish>
7005 \ProvidesFile{swedish.mld}[2006/01/13]
7006 %% Swedish titles for minitoc.sty
7007 %% from swedish.dtx (babel)
7008 %% Hellman, Sten and Östhols, Erik
7009 \def\ptctitle{Inneh\csname aa\endcsname ll}%
7010 \def\plftitle{Figurer}%
7011 \def\plttitle{Tabeller}%
7012 %%
7013 \def\mtctitle{Inneh\csname aa\endcsname ll}%
7014 \def\mlftitle{Figurer}%
7015 \def\mlttitle{Tabeller}%
7016 %%
7017 \def\stctitle{Inneh\csname aa\endcsname ll}%
7018 \def\slftitle{Figurer}%

```

```

7019 \def\sltttitle{Tabeller}%
7020 </swedish>

```

13.154 “Swedish2” language: `swedish2.mld`

The titles for the “swedish2” language (variant for `swedish`) are taken from the `rapport.doc` file in the `SLATEX` package [152]. See also section 13.153 on the page before.

```

7021 <*swedish2>
7022 \ProvidesFile{swedish2.mld}[2006/04/04]
7023 %% Swedish2 titles for minitoc.sty
7024 %% from rapport.doc (slatex)
7025 %% Mattisson, Sven
7026 \def\ptctitle{Inneh\csname aa\endcsname ll}%
7027 \def\plftitle{Figurf\"orteckning}%
7028 \def\plttitle{Tabellf\"orteckning}%
7029 %%
7030 \def\mtctitle{Inneh\csname aa\endcsname ll}%
7031 \def\mlftitle{Figurf\"orteckning}%
7032 \def\mlttitle{Tabellf\"orteckning}%
7033 %%
7034 \def\stctitle{Inneh\csname aa\endcsname ll}%
7035 \def\slftitle{Figurf\"orteckning}%
7036 \def\sltttitle{Tabellf\"orteckning}%
7037 </swedish2>

```

13.155 “Thai” language: `thai.mld` and `thai.mlo`

The titles for the “thai” language are taken from the `thaicjk.ldf` file and use fonts of the CJK system [143, 144]. The `thailatex` package [188] provides the same titles in its `thai.ldf` file.

The titles for the “thai” language contain characters that cannot be easily generated, hence we load `thai.mlo`.

```

7038 <*thai>
7039 \ProvidesFile{thai.mld}[2005/01/28]
7040 %% from thaicjk.ldf CJK 4.5.2 Thai support for the babel system
7041 %% by Werner Lemberg <wl@gnu.org>
7042 %%
7043 \mtcloadmlo{thai}%
7044 </thai>

```

13.156 “Turkish” language: `turkish.mld`

The titles for the “turkish” language are taken from the `turkish.dtx` file in the `babel` package [30, 31, 43]:

```

7045 <*turkish>
7046 \ProvidesFile{turkish.mld}[2006/01/13]
7047 %% Turkish titles for minitoc.sty
7048 %% from turkish.dtx (babel)
7049 %% Burc, Mustafa
7050 \def\ptctitle{\.I\c cindekiler}%
7051 \def\plftitle{\c Sekiller Listesi}%
7052 \def\pltttitle{Tablolar\in Listesi}%
7053 %%
7054 \def\mtctitle{\.I\c cindekiler}%
7055 \def\mlftitle{\c Sekiller Listesi}%
7056 \def\mltttitle{Tablolar\in Listesi}%
7057 %%
7058 \def\stctitle{\.I\c cindekiler}%
7059 \def\slftitle{\c Sekiller Listesi}%
7060 \def\slttitle{Tablolar\in Listesi}%
7061 </turkish>

```

13.157 “Uighur” language: `uighur.mld`

The “uighur” and “bicig” languages are synonyms, so we just load the `bicig.mld` file (see section 13.19 on page 424):

```

7062 <*uighur>
7063 \ProvidesFile{uighur.mld}[2006/05/31]
7064 \mtcselectlanguage{bicig}%
7065 </uighur>

```

13.158 “Uighur2” language: `uighur2.mld`

The “uighur2” and “bicig2” languages are synonyms, so we just load the `bicig2.mld` file (see section 13.20 on page 424):

```

7066 <*uighur2>
7067 \ProvidesFile{uighur2.mld}[2006/05/31]
7068 \mtcselectlanguage{bicig2}%
7069 </uighur2>

```

13.159 “Uighur3” language: uighur3.mld

The “uighur3” and “bicig3” languages are synonyms, so we just load the bicig3.mld file (see section 13.19 on page 424):

```
7070 < *uighur3 >
7071 \ProvidesFile{uighur3.mld}[2006/05/31]
7072 \mtcselectlanguage{bicig3}%
7073 < /uighur3 >
```

13.160 “UKenglish” language: UKenglish.mld

The “UKenglish” language is just like “english” (“UK” is for “United Kingdom”), so we just load english.mld (see section 13.42 on page 435):

```
7074 < *UKenglish >
7075 \ProvidesFile{UKenglish.mld}[2005/07/11]
7076 \mtcselectlanguage{english}%
7077 < /UKenglish >
```

13.161 “Ukraineb” language: ukraineb.mld

The “ukraineb” language is a synonym for “ukrainian”, so we just load ukrainian.mld. See section 13.162.

```
7078 < *ukraineb >
7079 \ProvidesFile{ukraineb.mld}[2006/01/23]
7080 %% Ukranian titles for minitoc.sty
7081 %% from ukraineb.dtx (babel)
7082 %% Werner, Eduard
7083 \selectlanguage{ukrainian}%
7084 < /ukraineb >
```

13.162 “Ukrainian” language: ukrainian.mld

The titles for the “ukrainian” language are taken from the ukraineb.dtx file in the babel package [30, 31, 71]. Cyrillic fonts are required. Another language name is ukraineb (see section 13.161).


```

7085 <*ukrainian>
7086 \ProvidesFile{ukrainian.mld}[2006/01/33]
7087 %% Ukraine titles for minitoc.sty
7088 %% from ukraineb.dtx (babel)
7089 %% Shvaika, Andriy
7090 %% Needs cyrillic fonts
7091 \def\mtctitle{{\cyr\CYZ\cyrm\cyrii\cyrs\cyrt}}%
7092 \def\mlftitle{{\cyr\CYRP\cyre\cyrr\cyre\cyrl\cyrii\cyrk
7093 \ \cyrii\cyrl\cyryu\cyrs\cyrt\cyrr\cyra\cyrc\cyrii\cyrishrt}}%
7094 \def\mltttitle{{\cyr\CYRP\cyre\cyrr\cyre\cyrl\cyrii\cyrk
7095 \ \cyrt\cyra\cyrb\cyrl\cyri\cyrc\cyrsfts}}%
7096 %%
7097 \def\ptctitle{{\cyr\CYZ\cyrm\cyrii\cyrs\cyrt}}%
7098 \def\plftitle{{\cyr\CYRP\cyre\cyrr\cyre\cyrl\cyrii\cyrk
7099 \ \cyrii\cyrl\cyryu\cyrs\cyrt\cyrr\cyra\cyrc\cyrii\cyrishrt}}%
7100 \def\pltttitle{{\cyr\CYRP\cyre\cyrr\cyre\cyrl\cyrii\cyrk
7101 \ \cyrt\cyra\cyrb\cyrl\cyri\cyrc\cyrsfts}}%
7102 %%
7103 \def\stctitle{{\cyr\CYZ\cyrm\cyrii\cyrs\cyrt}}%
7104 \def\slftitle{{\cyr\CYRP\cyre\cyrr\cyre\cyrl\cyrii\cyrk
7105 \ \cyrii\cyrl\cyryu\cyrs\cyrt\cyrr\cyra\cyrc\cyrii\cyrishrt}}%
7106 \def\sltttitle{{\cyr\CYRP\cyre\cyrr\cyre\cyrl\cyrii\cyrk
7107 \ \cyrt\cyra\cyrb\cyrl\cyri\cyrc\cyrsfts}}%
7108 </ukrainian>

```

13.163 “Uppersorbian” language: `uppersorbian.mld`

The titles for the “uppersorbian” language⁴² are taken from the `usorbian.dtx` file in the `babel` package [30, 31, 74]. See also section 13.102 on page 467. A shorter language name is `usorbian` (see section 13.165 on the next page).

```

7109 <*uppersorbian>
7110 \ProvidesFile{uppersorbian.mld}[2006/02/38]
7111 %% Upper sorbian titles for minitoc.sty
7112 %% from usorbian.dtx (babel)
7113 %% Werner, Eduard
7114 %% Needs cyrillic fonts
7115 \def\ptctitle{Wobsah}%
7116 \def\plftitle{Zapis wobrazow}%
7117 \def\pltttitle{Zapis tabulkow}%
7118 %%
7119 \def\mtctitle{Wobsah}%
7120 \def\mlftitle{Zapis wobrazow}%
7121 \def\mltttitle{Zapis tabulkow}%
7122 %%
7123 \def\stctitle{Wobsah}%

```

⁴²Upper 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 descendants 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 (VIth century).

```

7124 \def\slfttitle{Zapis wobrazow}%
7125 \def\sltttitle{Zapis tabulkow}%
7126 \end{uppersorbian}

```

13.164 “USenglish” language: USenglish.mld

The “USenglish” language “US” is for “United States (of America)” is just like “english”⁴³, so we just load english.mld (see section 13.42 on page 435):

```

7127 \begin{USenglish}
7128 \ProvidesFile{USenglish.mld}[2005/07/11]
7129 \mtcselectlanguage{english}%
7130 \end{USenglish}

```

13.165 “Usorbian” language: usorbian.mld

The “usorbian” language is a synonym for “uppersorbian”, so we just load uppersorbian.mld. See section 13.163 on the page before.

```

7131 \begin{usorbian}
7132 \ProvidesFile{usorbian.mld}[2006/01/23]
7133 %% Upper sorbian titles for minitoc.sty
7134 %% from usorbian.dtx (babel)
7135 %% Werner, Eduard
7136 \selectlanguage{uppersorbian}%
7137 \end{usorbian}

```

13.166 “Vietnam” language: vietnam.mld

The titles for the “vietnam” language are taken from the vietnam.sty package. Vietnamese fonts are required. See also section 13.167 on the following page.

```

7138 \begin{vietnam}
7139 \ProvidesFile{vietnam.mld}[1999/03/16]
7140 %% vietnamese titles for minitoc.sty
7141 %%
7142 \def\ptctitle{M\dc l\dc}%
7143 \def\plfttitle{Danh s\ach h\inh v\~e}%
7144 \def\pltttitle{Danh s\ach b\h{a}ng}%

```

⁴³ It should be true for the mini-table titles; the languages themselves have some differences, like the hyphenation rules.

```

7145 %%
7146 \def\mtctitle{M\d{u}c l\d{u}c}%
7147 \def\mlftitle{Danh s\'ach h\'inh v\~e}%
7148 \def\mltttitle{Danh s\'ach b\h{a}ng}%
7149 %%
7150 \def\stctitle{M\d{u}c l\d{u}c}%
7151 \def\slftitle{Danh s\'ach h\'inh v\~e}%
7152 \def\slttitle{Danh s\'ach b\h{a}ng}%
7153 </vietnam>

```

13.167 “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 13.166 on the page before.

```

7154 <*vietnamese>
7155 \ProvidesFile{vietnamese.mld}[2004/12/14]
7156 \mtcselectlanguage{vietnam}%
7157 </vietnamese>

```

13.168 “Welsh” language: welsh.mld

The titles for the “welsh” language are taken from the welsh.dtx file in the babel package [30, 31, 36]:

```

7158 <*welsh>
7159 \ProvidesFile{welsh.mld}[1999/12/06]
7160 %% Welsh titles for minitoc.sty
7161 %% from welsh.dtx (babel)
7162 %% Braams, Johannes
7163 \def\ptctitle{Cynnwys}%
7164 \def\plftitle{Rhestr Ddarluniau}%
7165 \def\pltttitle{Rhestr Dablau}%
7166 %%
7167 \def\mtctitle{Cynnwys}%
7168 \def\mlftitle{Rhestr Ddarluniau}%
7169 \def\mltttitle{Rhestr Dablau}%
7170 %%
7171 \def\stctitle{Cynnwys}%
7172 \def\slftitle{Rhestr Ddarluniau}%
7173 \def\slttitle{Rhestr Dablau}%
7174 </welsh>

```

13.169 “Xalx” language: xalx.mld

The titles for the “xalx” language are taken from the MonT_EX package [87, 89]. 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. See also sections 13.96 on page 464, 13.170, and 13.171 on the following page.

```

7175 <*xalx>
7176 \ProvidesFile{xalx.mld}[2005/11/16]
7177 %% Mongol (xalx) titles for minitoc.sty
7178 \def\ptctitle{{\mnr Garqig}}%
7179 \def\plftitle{{\mnr Zurgi"in jagsaalt}}%
7180 \def\plttitle{{\mnr X"usn"agti"in jagsaalt}}%
7181 %%
7182 \def\mtctitle{{\mnr Garqig}}%
7183 \def\mlftitle{{\mnr Zurgi"in jagsaalt}}%
7184 \def\mlttitle{{\mnr X"usn"agti"in jagsaalt}}%
7185 %%
7186 \def\stctitle{{\mnr Garqig}}%
7187 \def\slftitle{{\mnr Zurgi"in jagsaalt}}%
7188 \def\slttitle{{\mnr X"usn"agti"in jagsaalt}}%
7189 </xalx>

```

13.170 “Xalx2” language: xalx2.mld

The titles for the “xalx2” language are taken from the MonT_EX package [87, 89]. This is a variant for the “xalx” language (see section 13.169).

```

7190 <*xalx2>
7191 \ProvidesFile{xalx2.mld}[2006/03/31]
7192 %% Mongol (xalx2) titles for minitoc.sty
7193 \def\ptctitle{{\mnr Aguulga}}%
7194 \def\plftitle{{\mnr Zurgi"in jagsaalt}}%
7195 \def\plttitle{{\mnr X"usn"agti"in jagsaalt}}%
7196 %%
7197 \def\mtctitle{{\mnr Aguulga}}%
7198 \def\mlftitle{{\mnr Zurgi"in jagsaalt}}%
7199 \def\mlttitle{{\mnr X"usn"agti"in jagsaalt}}%
7200 %%
7201 \def\stctitle{{\mnr Aguulga}}%
7202 \def\slftitle{{\mnr Zurgi"in jagsaalt}}%
7203 \def\slttitle{{\mnr X"usn"agti"in jagsaalt}}%
7204 </xalx2>

```

13.171 “Xalx3” language: xalx3.mld

The titles for the “xalx3” language are taken from the MonTeX package [87, 89]. This is an other variant for the “xalx” language (see section 13.169 on the preceding page).

```

7205 <*xalx3>
7206 \ProvidesFile{xalx3.mld}[2006/03/31]
7207 %% Mongol (xalx3) titles for minitoc.sty
7208 \def\ptctitle{\xalx{Soderjanie}}%
7209 \def\plftitle{\xalx{Spisok risunkow}}%
7210 \def\plttitle{\xalx{Spisok tablic}}%
7211 %%
7212 \def\mtctitle{\xalx{Soderjanie}}%
7213 \def\mlftitle{\xalx{Spisok risunkow}}%
7214 \def\mlttitle{\xalx{Spisok tablic}}%
7215 %%
7216 \def\stctitle{\xalx{Soderjanie}}%
7217 \def\slftitle{\xalx{Spisok risunkow}}%
7218 \def\slttitle{\xalx{Spisok tablic}}%
7219 </xalx3>

```

Part III

Complements

Contents of the Third Part

Bibliography	504
Changes History	522
Index	549

Bibliography

- [1] Paul W. Abrahams, Karl Berry, and Kathryn A. Hargreaves. \TeX for the Impatient, September 2003. <mailto:impatient@tug.org>. Available from: <http://www.dante.de/tex-archive/info/impatient/book.pdf>. 504
- [2] Paul W. Abrahams, Karl Berry, and Kathryn A. Hargreaves. \TeX pour l'Impatient, September 2004. Translation of [1] by Marc Chaudemanche. Available from: <http://www.dante.de/tex-archive/info/impatient/fr/fbook.pdf>.
- [3] Alex AJ. Typesetting Malayalam Using \LaTeX 2 ϵ , April 2003. <mailto:alexaj@myrealbox.com>. Available from: <http://www.dante.de/tex-archive/language/malayalam/doc/mm-usr.pdf>. 41, 184, 416, 469, 470, 471
- [4] Alex AJ. Typesetting Malayalam Using Ω , January 2006. <mailto:indicTeX@gmail.com>. Available from: <http://www.dante.de/tex-archive/macros/omega/latex/contrib/malayalam-omega/doc/manual-malayalam.pdf>. 41, 184, 416, 470
- [5] American Mathematical Society. Instructions for Preparation of Papers and Monographs: $\mathcal{A}\mathcal{M}\mathcal{S}$ - \LaTeX . Providence, Rhode Island, 1999. Available from: <http://www.dante.de/tex-archive/macros/latex/required/amslatex/classes/instr-l.pdf>.
- [6] Donald Arseneau. The `wrapfig` package, January 2003. <mailto:asnd@triumf.ca>. Available from: <http://www.dante.de/tex-archive/contrib/macros/latex/contrib/wrapfig/>. 133, 203, 213
- [7] Donald Arseneau. `chapterbib.sty`, February 2004. <mailto:asnd@reg.triumf.ca>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/cite/chapterbib.sty>. 57, 191, 203
- [8] Donald Arseneau. The `notoccite` package, July 2000. <mailto:asnd@triumf.ca>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/misc/notoccite.sty>. 49, 206, 222, 241, 266, 269, 272, 364, 392
- [9] Donald Arseneau. The `placeins` package, June 2002. <mailto:asnd@triumf.ca>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/placeins/>. 29, 195, 197, 208, 222, 371, 377, 391
- [10] Helmer Aslaksen. Chinese \TeX Using the CJK \LaTeX Package, Unicode TrueType Fonts and `pdf\TeX` under Windows [online]. February 2006. Available from: <http://www.math.nus.edu.sg/aslaksen/cs/cjk.html>. <mailto:aslaksen@math.nus.edu.sg>.

- [11] Helmer Aslaksen. Reading and Writing Chinese Characters and Pinyin on the Web Using Unicode [online]. January 2006. Available from: <http://www.math.nus.edu.sg/aslaksen/read.html>. <mailto:aslaksen@math.nus.edu.sg>.
- [12] Pierre Basso and Stephan Ulrich. `bibtopic.sty`, August 2002. <mailto:Pierre.Basso@lim.univ-mrs.fr>, <mailto:stefanulrich@users.sourceforge.net>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/bibtopic/bibtopic.pdf>. 57, 189
- [13] Benjamin Bayart. *Joli manuel pour L^AT_EX 2_ε*. Guide local pour l'ESIEE, December 1995. Available from: <http://www.dante.de/tex-archive/info/JMPL.ps.gz>. 46
- [14] Benjamin Bayart, Thierry Bayet, Prakash Countcham, Éric Depardieu, Jean-Pierre F. Drucbert, Mathieu Goutelle, Yvon Henel, Florence Henry, Loïc Joly, Christophe Jorssen, Erwan Le Pennec, Nicolas Markey, Françoise Marre-Fournier, Sébastien Mengin, Josselin Noirel, Bruno Piguët, Éric Streit, Thomas van Oudenhove, and Damien Wyart. Foire aux questions du groupe `fr.comp.text.tex` (version 3.0 α) [online]. November 2004. Available from: <http://faqfctt.fr.eu.org/>. <mailto:bayartb@edgard.fdn.fr>.
- [15] Claudio Beccari and Apostolos Syropoulos. New Greek Fonts and the greek option of the babel package. *TUGboat*, 19(4):419–425, December 1998. Available from: <http://www.tug.org/TUGboat/Articles/tb19-4/tb61becc.pdf>.
- [16] Emmanuel Beffara. Rubber Manual 1.0 [online]. July 2005. Available from: <http://www.pps.jussieu.fr/~beffara/soft/rubber/doc.html>. 58, 209
- [17] Alexander Berdnikov and Olga A. Grineva. Some Problems with Accents in T_EX: Letters with Multiple Accents and Accents Varying for Uppercase/Lowercase Letters. In *Congrès EuroT_EX'98*, volume 28-29 of *Cahiers GUTenberg*, pages 44–55, St. Malo, March 1998. Available from: <http://www.gutenberg.eu.org/pub/GUTenberg/publicationsPDF/28-29-berdnikovc.pdf>.
- [18] Alexander Berdnikov, Olga G. Lapko, Mikhail Kolodin, Andrew Janishevsky, and Alexey Burykin. Alphabets Necessary for Various Cyrillic Writing Systems (Towards X2 and T2 Encodings). In *Congrès EuroT_EX'98*, volume 28-29 of *Cahiers GUTenberg*, pages 33–43, St. Malo, March 1998. Available from: <http://www.gutenberg.eu.org/pub/GUTenberg/publicationsPDF/28-29-berdnikobv.pdf>.
- [19] Alexander Berdnikov, Olga G. Lapko, Mikhail Kolodin, Andrew Janishevsky, and Alexey Burykin. Cyrillic encoding for L^AT_EX 2_ε multi-language documents. *TUGboat*, 19(4):403–416, 1998. Available from: <http://www.tug.org/TUGboat/Articles/tb19-4/tb61berd.pdf>.
- [20] Alexander Berdnikov, Olga G. Lapko, Mikhail Kolodin, Andrew Janishevsky, and Alexey Burykin. The Encoding Paradigm in L^AT_EX 2_ε and the Projected X2 Encoding for Cyrillic Texts. In *Congrès EuroT_EX*, volume 28-29 of *Cahiers GUTenberg*, pages 17–31, March 1998. Available from: <http://www.gutenberg.eu.org/pub/GUTenberg/publicationsPDF/28-29-berdnikova.pdf>.
- [21] Karl Berry. Filenames for fonts. *TUGboat*, 11(4):517–520, November 1990. Available from: <http://www.tug.org/TUGboat/Articles/tb11-4/tb30berry.pdf>. 516

- [22] Karl Berry. Fontname, September 2005. Available from: <http://www.tug.org/fontname/fontname.pdf>.
- [23] Berhanu Beyene, Manfred Kudlek, Olaf Kummer, and Jochen Metzinger. Ethiopian Language Support for the babel Package, December 1998. Universität Hamburg. Available from: <http://www.dante.de/tex-archive/language/ethiopia/ethiop/doc/ethiodoc.ps>. 41, 194, 416, 438, 439
- [24] Javier Bezos. Estilo spanish para el sistema babel, September 2003. <mailto:jbezos@wanadoo.es>. Available from: <http://filemon.mecanica.upm.es/CervanTeX/spanish.dtx>. 493
- [25] Javier Bezos. Spanish support from the babel system, February 2004. <mailto:jbezos@wanadoo.es> or <mailto:babel@braams.cistron.nl>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/spanish.dtx>. 491, 492
- [26] Javier Bezos. Typesetting Guaraní with T_EX, July 2004. Available from: <http://www.dante.de/tex-archive/language/guarani/guarani.pdf>. 41, 196, 416, 451
- [27] Javier Bezos. The titlesec and titletoc packages, January 2005. <http://www.texytipografia.com/contact.php>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/titlesec/titlesec.pdf>. 211, 212, 371
- [28] Victor Boyko. The russian.sty file, January 1995. Available from: <ftp://ftp.dante.de/tex-archive/macros/latex/contrib/cmcyalt/russian.sty>. 485
- [29] Johannes Braams. Babel, a multilingual style-option system for use with L^AT_EX's standard document styles. *TUGboat*, 12(2):291–301, June 1991. Available from: <http://www.tug.org/TUGboat/Articles/tb12-2/tb32braa.pdf>. 416
- [30] Johannes Braams. Babel, a multilingual package for use with L^AT_EX's standard document classes, June 2004. <mailto:babel@braams.cistron.nl>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/babel.pdf>. 36, 41, 72, 189, 194, 219, 416, 417, 418, 422, 423, 426, 427, 428, 429, 431, 433, 434, 435, 436, 437, 438, 441, 442, 443, 444, 445, 446, 447, 457, 458, 459, 460, 464, 465, 466, 467, 468, 469, 473, 474, 475, 476, 477, 479, 480, 481, 483, 488, 489, 490, 491, 492, 493, 495, 496, 497, 499
- [31] Johannes Braams. Babel, a multilingual package for use with L^AT_EX's standard document classes (user), June 2004. <mailto:babel@braams.cistron.nl>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/user.pdf>. 36, 41, 72, 189, 194, 219, 416, 417, 418, 422, 423, 426, 427, 428, 429, 431, 433, 434, 435, 436, 437, 438, 441, 442, 443, 444, 445, 446, 447, 457, 458, 459, 460, 464, 465, 466, 467, 468, 469, 473, 474, 475, 476, 477, 479, 480, 481, 483, 488, 489, 490, 491, 492, 493, 495, 496, 497, 499
- [32] Johannes Braams. Dutch language support from the babel system, March 2005. <mailto:babel@braams.cistron.nl>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/dutch.dtx>. 417, 434
- [33] Johannes Braams. English support from the babel system, March 2005. <mailto:babel@braams.cistron.nl>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/english.dtx>. 435, 436

- [34] Johannes Braams. Irish support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/irish.dtx>. 459
- [35] Johannes Braams. Norsk support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/norsk.dtx>. 475, 476
- [36] Johannes Braams. Welsh support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/welsh.dtx>. 499
- [37] Johannes Braams, Juan M. Aguirregabiria, and Julio Sanchez. Basque support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>, <mailto:lg.ehu.es>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/basque.dtx>. 423
- [38] Johannes Braams and Einar Árnason. Icelandic support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>, <mailto:einar@lif.hi.is>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/icelandic.dtx>. 458
- [39] Johannes Braams and Goncal Badenes. Catalan language support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>, <mailto:badenes@imec.be>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/catalan.dtx>. 431
- [40] Johannes Braams and Claudio Beccari. Latin support from the babel system, November 2005. <mailto:babel@bramms.cistron.nl>, <mailto:claudio.beccari@polito.it>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/latin.dtx>. 464, 465
- [41] Johannes Braams, Árpád Bíró, and József Bérces. Magyar support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/magyar.dtx>. 468, 469
- [42] Johannes Braams and Georgi Boshnakov. Bulgarian language support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>, <mailto:georgi.boshnakov@umist.ac.uk>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/bulgarian.dtx>. 428
- [43] Johannes Braams and Mustafa Burc. Turkish support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>, <mailto:rz6001@rziris01.rz.uni-hamburg.de>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/turkish.dtx>. 495
- [44] Johannes Braams, David Carlisle, Alan Jeffrey, Leslie Lamport, Frank Mittelbach, Chris Rowley, and Rainer Schöpf. The L^AT_EX 2_ε sources, 2003. Available from: <http://www.dante.de/tex-archive/macros/latex/base>.
- [45] Johannes Braams and Manuel Carriba. Galician support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>, <mailto:mcarriba@eunetcom.net>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/galician.dtx>. 445

- [46] Johannes Braams, Jana Chlebikova, and Tobias Schlemmer. Slovak support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>, <mailto:chlebikj@mff.uniba.cs>, <mailto:Tobias.Schlemmer@web.de>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/slovak.dtx>. 490
- [47] Johannes Braams, Maurizio Codogno, and Claudio Beccari. Italian support from the babel system, November 2005. <mailto:babel@bramms.cistron.nl>, <mailto:mau@beatles.cselt.stet.it>, <mailto:beccari@polito.it>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/italian.dtx>. 460
- [48] Johannes Braams, Victor Eijkhout, and Nico Poppelier. The development of national L^AT_EX styles. *TUGboat*, 10(3):401–406, November 1989. Available from: <http://www.tug.org/TUGboat/Articles/tb10-3/tb25braams.pdf>. 416
- [49] Johannes Braams and Daniel Flipo. French support from the babel system, February 2005. <mailto:babel@bramms.cistron.nl>, <mailto:Daniel.Flipo@univ-lille1.fr>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/frenchb.dtx>. 443, 444
- [50] Johannes Braams and Fraser Grant. Scottish support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>, <mailto:fraser@cernvm>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/scottish.dtx>. 488
- [51] Johannes Braams and Sten Hellman. Swedish support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>, <mailto:hellman@cernvm.cern.ch>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/swedish.dtx>. 493
- [52] Johannes Braams, Umstatter Horst, and Robert Juhasz. Romanian support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>, <mailto:hhu@cernvm.cern.ch>, <mailto:robertj@uni-paderborn.de>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/romanian.dtx>. 479
- [53] Johannes Braams and Regnor Jernsletten. North Sami support from the babel system, February 2004. <mailto:babel@bramms.cistron.nl>, <mailto:Regnor.Jernsletten@sami.uit.no> or <mailto:Regnor.Jernsletten@eunet.no>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/samin.dtx>. 488
- [54] Johannes Braams, Mikko Kanerva, and Keranen Reino. Finnish support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>, <mailto:kanerva@cernvm>, <mailto:keranen@cernvm>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/finnish.dtx>. 441, 442
- [55] Johannes Braams and Peter Kleiweg. Interlingua support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>, <mailto:kleiweg@let.rug.nl>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/interlingua.dtx>. 459
- [56] Johannes Braams, Jörg Knappen, and Terry Mart. Bahasa indonesia support from the babel system, November 2005. <mailto:babel@bramms.cistron.nl>, <mailto:>

- joerg.knappen@alpha.ntp.springer.de, <mailto:mart@vkpmzd.kph.uni-mainz.de>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/bahasa.dtx>. 422
- [57] Johannes Braams, Jörg Knappen, Terry Mart, and Bob Margolis. Bahasa malaysia support from the babel system, November 2005. <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://www.dante.de/tex-archive/macros/latex/required/babel/bahasam.dtx>. 422
- [58] Johannes Braams, Olga G. Lapko, Vladimir Volovich, and Werner Lemberg. Russian support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>, <mailto:cyrtug@mir.msk.su>, <mailto:TeX@vvv.vsu.ru> et <mailto:wl@gnu.org>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/russianb.dtx>. 429, 483
- [59] Johannes Braams and Henning Larsen. Danish language support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>, <mailto:larsen@cernvm.cern.ch>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/danish.dtx>. 434
- [60] Johannes Braams and Boris Lavva. Hebrew language support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/hebrew.dtx>. 457
- [61] Johannes Braams and Milos Lokajicek. Czech language support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>, <mailto:lokajick@cernvm>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/czech.dtx>. 433
- [62] Johannes Braams, Ruiz-Altava Marti, and Knappen. Jörg. Esperanto support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>, <mailto:ruizaltb@cernvm.cern.ch>, <mailto:knappen@vkpmzd.kph.uni-mainz.de>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/esperanto.dtx>. 437
- [63] Johannes Braams, Dejan Muhamedagić, and Jankovic Slobodan. Serbocroatian support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>, <mailto:dejan@unix.com>, <mailto:slobodan@archimed.filfak.ac.ni.yu>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/serbian.dtx>. 489
- [64] Johannes Braams and Alan Paić. Croatian language support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>, <mailto:paica@cernvm.cern.ch>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/croatian.dtx>. 433
- [65] Johannes Braams and Bernd Raichle. German support from the babel system, February 2004. <mailto:babel@bramms.cistron.nl>, azu.Informatik.Uni-Stuttgart.de. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/germanb.dtx>. 446

- [66] Johannes Braams, Bernd Raichle, and Walter Schmidt. New german support from the babel system, February 2004. <mailto:babel@bramms.cistron.nl>, <mailto:raichle@azu.Informatik.Uni-Stuttgart.de>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/ngermanb.dtx>. 473, 474
- [67] Johannes Braams and Jose Pedro Ramalhete. Portuguese support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>, <mailto:jramalhe@cernvm> or mailto:Jose-Pedro_Ramalhete@macmail. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/portuges.dtx>. 426, 479
- [68] Johannes Braams and Christian Rolland. Breton language support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>, <mailto:Christian.Rolland@univ-brest.fr>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/breton.dtx>. 427
- [69] Johannes Braams and Enn Saar. Estonian support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>, <mailto:saar@aai.ee>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/estonian.dtx>. 438
- [70] Johannes Braams, Elmar Schalueck, and Michael Janich. Polish support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>, <mailto:elmar@uni-paderborn.de>, <mailto:massa@uni-paderborn.de>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/polish.dtx>. 477
- [71] Johannes Braams and Andrij Shvaika. Ukrainian support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>, <mailto:ashv@icmp.lviv.ua>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/ukraineb.dtx>. 496
- [72] Johannes Braams and Apostolos Syropoulos. Greek support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>, <mailto:apostolo@platon.ee.duth.gr> or <mailto:apostolo@obelix.ee.duth.gr>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/greek.dtx>. 447
- [73] Johannes Braams and Eduard Werner. Lower Sorbian support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>, <mailto:edi@kaihh.hanse.de>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/lisorbian.dtx>. 467
- [74] Johannes Braams and Eduard Werner. Upper Sorbian support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>, <mailto:edi@kaihh.hanse.de>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/usorbian.dtx>. 497
- [75] Johannes Braams and Adi Zaimi. Albanian support from the babel system, November 2005. <mailto:babel@bramms.cistron.nl>, <mailto:zami1st@yahoo.com>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/albaban.dtx>. 418
- [76] Johannes Braams, Danilo Zavrtanik, and Leon Žlajpah. Slovene support from the babel system, March 2005. <mailto:babel@bramms.cistron.nl>, <mailto:leon.zlajpah@ijs.si>. Available from: <http://www.dante.de/tex-archive/macros/latex/required/babel/slovene.dtx>. 490

- [77] Felix Braun. Die Documentklasse jura, July 1998. <mailto:fbraun@atdot.org>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/jura/>. 187, 197, 370
- [78] Peter Breitenlohner. *The ε -T_EX manual*. The \mathcal{N} -S Team, Max-Planck-Institut für Physik, München, February 1998. Available from: http://www.dante.de/tex-archive/systems/e-tex/v2/doc/etex_man.pdf. 192
- [79] David Carlisle. The xr package, May 1994. <mailto:carlisle@cs.man.ac.uk>. Available from: <http://www.dante.de/tex-archive/macros/required/tools/xr.dtx>. 26, 213, 278
- [80] David Carlisle. The xr-hyper package, March 2000. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/hyperref/xr-hyper.sty>. 213
- [81] David Carlisle. The color package, November 2005. <mailto:carlisle@cs.man.ac.uk>. Available from: <http://www.dante.de/tex-archive/macros/required/graphics/color.dtx>. 120
- [82] Otfried Cheong. Using TrueType Fonts in PdfL^AT_EX [online]. September 2005. Available from: <http://ipe.comp.geom.org/pdftex.html>. <mailto:otfried@tclab.kaist.ac.kr>.
- [83] Pai H. Chou. How to make L^AT_EX (teT_EX) handle unicode and CJK in MacOSX [online]. December 2004. Available from: <http://www.ece.uci.edu/~chou/unicode-tex.html>. <mailto:chou@ece.uci.edu>.
- [84] Steven Douglas Cochran. The subfigure package, March 1995. <mailto:sds@cs.cmu.edu>, <mailto:cochran@ieee.org>. Available from: <http://www.dante.de/tex-archive/obsolete/macros/latex/contrib/subfigure/subfigure.pdf>. 31, 114, 211
- [85] Steven Douglas Cochran. The captcont package, February 2002. <mailto:sds@cs.cmu.edu>, <mailto:cochran@ieee.org>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/captcont/>. 51, 73, 189, 225, 243
- [86] Steven Douglas Cochran. The subfig Package, January 2004. <mailto:sds@cs.cmu.edu>, <mailto:cochran@ieee.org>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/subfig/subfig.pdf>. 31, 116, 189, 211, 212, 356
- [87] Oliver Corff. MonT_EX – A Quick Guide (*draft*), July 2002. <mailto:corff@zedat.fu-berlin.de>. Available from: <http://www.dante.de/tex-archive/language/mongolian/montex/doc/mlsquick.pdf>. 41, 184, 201, 416, 424, 425, 426, 429, 430, 472, 484, 500, 501
- [88] Oliver Corff. Infosystem Mongolei [online]. July 2004. Available from: <http://userpage.fu-berlin.de/~corff/infomong.html>. An Internet-based Journal on Mongolian Affairs. 202
- [89] Oliver Corff and Dorjpalam Dorj. MonT_EX – mongolian for L^AT_EX 2_ε, Implementation Level System Documentation, July 2002. <mailto:corff@zedat.fu-berlin.de>. Available from: <http://www.dante.de/tex-archive/language/mongolian/montex/doc/montex.pdf>. 41, 184, 201, 202, 416, 424, 425, 426, 429, 430, 472, 484, 500, 501

- [90] Serguei Dachian, Arnak Dalalyan, and Vardan Vakopian. ArmT_EX: a System for Writing in Armenian with T_EX and L^AT_EX (written in armenian), June 1999. <mailto:Serguei.Dachian@univ-lemans.fr>. Available from: <http://www.dante.de/tex-archive/language/armtex/v2.0/manual.ps>. 41, 188, 416, 420
- [91] Bernard Desgraupes. *ET_EX, Apprentissage, guide et référence*. Vuibert, Paris, second edition, 2003.
- [92] Jean-Pierre F. Drucbert. The shorttoc package, August 2002. <mailto:jean-pierre.drucbert@onera.fr>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/shorttoc/shorttoc.dtx>. 19, 210
- [93] Jean-Pierre F. Drucbert. Le paquetage minitoc, August 2006. <mailto:jean-pierre.drucbert@onera.fr>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/minitoc/fminitoc.pdf>. 395
- [94] Jean-Pierre F. Drucbert. The minitoc package, August 2006. <mailto:jean-pierre.drucbert@onera.fr>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/minitoc/minitoc.pdf>. 395
- [95] Victor Eijkhout. The comment package, October 1999. <mailto:victor@eijkhout.net>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/comment/>.
- [96] Behdad Esfahbod and Roozbeh Pournader. FarsiT_EX and the Iranian T_EX Community. *TUGboat*, 22(3):41–44, 2003. Available from: <http://www.tug.org/TUGboat/Articles/tb23-1/farsitex.pdf>.
- [97] Mike Fabian. CJK Support in SuSe Linux, March 2005. Available from: <http://www.suse.de/~mfabian/suse-cjk.pdf>.
- [98] Robin Fairbairns. The UK T_EX FAQ. Your 396 Questions Answered, November 2005. Version 3.15a. Available from: <ftp://cam.ctan.org/tex-archive/help/uk-tex-faq/newfaq.pdf>.
- [99] Michael J. Ferguson. Report on multilingual activities. *TUGboat*, 11(4):514–516, November 1990.
- [100] Chuck Genschte and John Warnock. PDF Reference, November 2004. Available from: <http://partners.adobe.com/public/developer/en/pdf/PDFReference16.pdf>. 208
- [101] Helen Gilhooly and Rozenn Etienne. *Lire et écrire le japonais*. Langues orientales. Larousse, Paris, March 2006. Translation and adaptation to french by Cécile DESPRAIRIES.
- [102] Michel Goossens and Sebastian Rahtz. *The L^AT_EX Web Companion: Integrating T_EX, HTML, and XML*. Tools and Techniques for Computer Typesetting. Addison-Wesley, Reading, Massachusetts, May 1999. With Eitan M. GURARI, ROSS MOORE and Robert S. SUTOR.
- [103] Norman Gray. The urlbst package, June 2005. <http://www.astro.gla.ac.uk/users/norman/>. Available from: <http://www.tug.org/tex-archive/macros/latex/contrib/urlbst/urlbst.pdf>. 196, 208, 224, 543

- [104] Thorsten Hansen. The bibunits package, May 2004. <mailto:thorsten.hansen@psychol.uni-giessen.de>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/bibunits/bibunits.pdf>. 57, 189, 203
- [105] Thorsten Hansen. The multibib package, January 2004. <mailto:thorsten.hansen@psychol.uni-giessen.de>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/multibib/multibib.pdf>. 57, 203
- [106] Patrick Happel. The lipsum package, January 2005. <mailto:patrick.happel@rub.de>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/lipsum/lipsum.pdf>. 199, 547
- [107] Yannis Haralambous. *Fontes et codages*. O'Reilly France, April 2004. 451, 454, 455
- [108] Yannis Haralambous and John Plaice. First applications of Ω : Greek, Arabic, Khmer, Poetica, ISO 10646/Unicode, etc. *TUGboat*, 15(3):344–352, 1994.
- [109] Yannis Haralambous and John Plaice. Ω , a \TeX extension including Unicode and featuring lex-like filtering processes. In Wlodek Bzyl and Plata-Przechlewski, editors, *Proceedings of the European \TeX Conference*, pages 153–166, Gdnask, Poland, 1994. GUST. 513
- [110] Yannis Haralambous and John Plaice. The Design and Use of a Multiple-Alphabet Font with Omega. *MAPS*, 27:27–37, 2000. Available from: http://www.ntg.nl/maps/pdf/24_7.pdf.
- [111] Yannis Haralambous, John Plaice, and Éric Picheral. Ω , une extension de \TeX incluant UNICODE et des filtres de type lex. *Cahiers GUTenberg*, 20:55–80, June 1995. French translation of [109].
- [112] Alexander Harin and Vadim V. Zhytnikov. The cmcyralt.sty file, December 1994. Available from: <ftp://ftp.dante.de/tex-archive/macros/latex/contrib/cmcyralt/cmcyralt.sty>. 485
- [113] Stephan Hennig. The mcaption package, September 2005. <mailto:stephanhennig@arcor.de>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/mcaption/mcaption.pdf>. 51, 69, 200, 244
- [114] Alan Hoenig. *\TeX Unbound*. Oxford University Press, Oxford New York, 1998. \LaTeX & \TeX Strategies for Fonts, Graphics, & More.
- [115] Hiroya Ikeda. $\text{p}\text{\LaTeX} 2_{\epsilon}$, September 2001. In japanese, <mailto:ike@kobitosan.net>. Available from: <http://www.kobitosan.net/ike/latex/handhtml/index.html>. 208
- [116] Youssef Jabri. Typesetting Arabic and Farsi with the *Arabi* package. The Users Guide, February 2006. <mailto:yjabri@ensa.univ-oujda.ac.ma>. Available from: http://www.dante.de/tex-archive/language/arabic/arabi/doc/latex/arabi/user_guide.pdf. 36, 41, 188, 419, 441
- [117] Alan Jeffrey. PostScript font support in $\text{\LaTeX} 2_{\epsilon}$. *TUGboat*, 15(3):263–268, September 1994. Available from: <http://www.tug.org/TUGboat/Articles/tb15-3/tb44jeff.pdf>.

- [118] Zhuhan Jiang. The `vruler` package, October 1996. <mailto:zhuhan@neumann.une.edu.au>, and look at <http://www.dante.de/tex-archive/help/Catalogue/entries/vruler.html>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/misc/vruler.sty>.
- [119] David M. Jones. `hangcaption.sty`, August 1992. <mailto:dmjones@theory.lcs.mit.edu>. Available from: <http://www.dante.de/tex-archive/macros/latex209/contrib/misc/hangcaption.sty>. 196
- [120] Donald E. Knuth. *Computer Modern Typefaces*, volume E of *Computers and Typesetting*. Addison-Wesley, Reading, Massachusetts, 1986. 34, 192
- [121] Donald E. Knuth. *The T_EXbook*, volume A of *Computers and Typesetting*. Addison-Wesley, Reading, Massachusetts, 16th edition, 1989. Revised to cover T_EX3, 1991. 190, 198, 211, 212, 514
- [122] Donald E. Knuth. *The METAFONTbook*, volume C of *Computers and Typesetting*. Addison-Wesley, Reading, Massachusetts, 1993. With the final corrections of 1993. 192
- [123] Donald E. Knuth. *Le T_EXbook, Composition informatique*. Vuibert, Paris, 2003. Translation in french of [121], by Jean-Côme CHARPENTIER. 198
- [124] Un Koaunghi. H_LT_EX: a L_AT_EX package for the Korean language support, April 2005. Written in korean. Available from: <http://www.dante.de/tex-archive/language/korean/HLaTeX/hlguide.pdf>. 184, 196, 416, 454, 455
- [125] Helmut Kopka and Patrick W. Daly. *Guide to L_AT_EX*. Tools and Techniques for Computer Typesetting. Addison Wesley, Reading, Massachusetts, fourth edition, 2004. 207
- [126] Siep Kroonenberg. NFSS: using font families in L_AT_EX 2_ε. *MAPS*, 11:52–54, 1999. <mailto:siepo@cybercomm.nl>. Available from: http://www.ntg.nl/maps/pdf/22_13.pdf.
- [127] Alexej M. Kryukov. Typesetting multilingual documents with ANTOMEGA, September 2003. Available from: <http://www.dante.de/tex-archive/systems/omega/contrib/antomega/doc/omega/antomega/antomega.pdf>. 41, 187, 416, 448, 449, 450, 465, 477, 481, 482, 491, 492
- [128] Frank Küster. `alnumsec.sty`: Using alphanumeric section numbering with standard sectioning commands, February 2005. frank@kuesterei.ch. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/alnumsec/>. 51, 73, 187, 225, 243
- [129] Thomas Lachand-Robert. *La maîtrise de T_EX et L_AT_EX*. Masson, Paris, 1995.
- [130] Klaus Lagally. ArabT_EX—Typetting Arabic with vowels and ligatures. In *Proceedings of the 7th European T_EX Conference*, pages 153–172, Prague, 1992. CsTUG. <mailto:lagallyk@acm.org>. Available from: http://www.ntg.nl/maps/pdf/20_22.pdf. 36, 41, 188, 416, 418, 419, 456
- [131] Klaus Lagally. ArabT_EX Version 3, October 2003. <mailto:lagallyk@acm.org>. Available from: <http://www.dante.de/tex-archive/language/arabtex/doc/html/arabtex.htm>. 36, 41, 188, 416, 418, 419, 456

- [132] Leslie Lamport. *TEX: A Document Preparation System: User's Guide and Reference Manual*. Addison-Wesley, Reading, Massachusetts, second edition, 1994. 54, 198
- [133] Leslie Lamport, Frank Mittelbach, and Johannes Braams. Standard Document Classes for L^AT_EX version 2e, September 1999. Available from: <http://www.dante.de/tex-archive/macros/latex/base/classes.dtx>. 184
- [134] Olga G. Lapko. Full Cyrillic: How Many Languages? In M. Burbank and C. Thiele, editors, *Proceedings of TUG'96*, pages 164–170, Dubna, 1996. JINR. Available from: <http://www.tug.org/TUGboat/Articles/tb17-2/tb51olga.pdf>.
- [135] Olga G. Lapko. The floatrow package, July 2005. <mailto:Lapko.O@g23.relcom.ru>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/floatrow/floatrow.pdf>. 29
- [136] Olga G. Lapko and Irina Makhovaya. The Style russianb for Babel: Problems and solutions. *TUGboat*, 16(4):364–372, 1995. Available from: <http://www.tug.org/TUGboat/Articles/tb16-4/tb49olga.pdf>. 36, 429, 483
- [137] L^AT_EX3 Project Team. Default docstrip headers. *TUGboat*, 19(2):137–138, June 1998. Available from: <http://www.tug.org/TUGboat/Articles/tb19-2/tb59ltdocstrip.pdf>.
- [138] L^AT_EX3 Project Team. L^AT_EX 2_ε Kernel (Output Routine), July 2000. Available from: <http://www.dante.de/tex-archive/macros/latex/base/ltoutput.dtx>. 29, 208, 391
- [139] L^AT_EX3 Project Team. L^AT_EX 2_ε for authors, July 2001. Available from: <http://www.dante.de/tex-archive/macros/latex/doc/usrguide.pdf>. 184
- [140] L^AT_EX3 Project Team. L^AT_EX 2_ε for class and package writers, December 2003. Available from: <http://www.dante.de/tex-archive/macros/latex/doc/clsguide.pdf>. 184
- [141] L^AT_EX3 Project Team. L^AT_EX 2_ε font selection, February 2004. Available from: <http://www.dante.de/tex-archive/macros/latex/doc/fntguide.pdf>. 184, 203, 261, 339, 342
- [142] L^AT_EX3 Project Team. L^AT_EX font encodings, January 2006. Available from: <http://www.dante.de/tex-archive/macros/latex/doc/encguide.pdf>. 194, 204
- [143] Werner Lemberg. The CJK package for L^AT_EX 2_ε — Multilingual support beyond babel. *TUGboat*, 18(3):214–224, 1997. Available from: <http://www.tug.org/TUGboat/Articles/tb18-3/cjkintro600.pdf>. 36, 41, 191, 210, 211, 416, 432, 452, 453, 454, 455, 461, 462, 463, 494
- [144] Werner Lemberg. The CJK package [online]. August 2005. Available from: <http://cjk.ffii.org/>. 36, 41, 191, 210, 211, 416, 432, 452, 453, 454, 455, 461, 462, 463, 494
- [145] Knut Lickert. L^AT_EX 2_ε for people in associations: minutes.sty, January 2001. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/minutes/>. 201, 262, 363

- [146] Ulf A. Lindgren. FncyChap V1.33, August 2005. <mailto:ulf.a.lindgren@ericsson.com>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/fncychap/fncychap.pdf>. 50, 72, 195, 225, 242
- [147] Anselm Lingnau. An Improved Environment for Floats, November 2001. <mailto:anselm@strathspey.org>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/float/float.dtx>. 29
- [148] Maurizio Loreti. The sfheaders package, September 1997. <mailto:loreti@padova.infn.it> and <http://wwwcdf.pd.infn.it/~loreti/mlo.html>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/sfheaders/>. 51, 72, 210, 225, 243
- [149] Ken Lunde. *Understanding Japanese Information Processing*. O'Reilly & Associates, Inc, September 1993. 184, 192, 516
- [150] Ken Lunde. Online companion of “Understanding Japanese Information Processing”, 1996. See [149]. Available from: <ftp://ftp.ora.com/pub/examples/nutshell/ujip/doc/cjk.inf>. 184, 192
- [151] Nicolas Markey. Tame the BeaST (The B to X of BibT_EX). <mailto:markey@lsv.ens-cachan.fr>, June 2005. Available from: http://www.dante.de/tex-archive/info/bibtex/tamethebeast/ttb_en.pdf. 184, 189
- [152] Sven Mattisson. SI_AT_EX: swedish speaking version of L_AT_EX, November 1989. <mailto:sven@tde.lu.se>. Available from: <http://ftp.dante.de/tex-archive/language/swedish/slatex/slatex.1>. 41, 210, 494
- [153] Rowland McDonnell. The sectsty package v2.0.2, April 2002. <mailto:rowland.mcdonnell@physics.org>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/sectsty/>. 50, 67, 127, 210, 223, 242
- [154] Frank Mittelbach. ε -T_EX: Guidelines for future T_EX. *TUGboat*, 11(3):337–345, September 1990. Available from: <http://www.tug.org/TUGboat/Articles/tb11-3/tb29mitt.pdf>.
- [155] Frank Mittelbach. Comments on “Filenames for Fonts” [21]. *TUGboat*, 13(1):51–53, April 1992. Available from: <http://www.tug.org/TUGboat/Articles/tb13-1/tb34mittfont.pdf>.
- [156] Frank Mittelbach. An environment for multicolumn output, July 2000. <mailto:Frank.Mittelbach@latex-project.org>. Available from: <http://www.dante.de/tex-archive/macros/required/tools/multicol.pdf>. 119, 120
- [157] Frank Mittelbach. The doc and shortverb packages, January 2004. Available from: <http://www.tug.org/tex-archive/macros/latex/base/doc.dtx>.
- [158] Frank Mittelbach, Denys Duchier, Johannes Braams, Marcin Woliński, and Mark Wooding. The docstrip program, January 2004. Available from: <http://www.tug.org/tex-archive/macros/latex/base/docstrip.dtx>.
- [159] Frank Mittelbach, Michel Goossens, Johannes Braams, David Carlisle, and Chris Rowley. *The L_AT_EX Companion 2*. Tools and Techniques for Computer Typesetting. Addison-Wesley Professional, Reading, Massachusetts, 2004. 29, 132, 184, 199, 208, 212, 391

- [160] Frank Mittelbach, Michel Goossens, Johannes Braams, David Carlisle, and Chris Rowley. *ℒ_ᵀᵀ_ᵀ Companion*. Pearson Éducation France, Paris, second edition, 2005. French translation by Jacques ANDRÉ, Benoît BELET, Jean-Côme CHARPENTIER, Jean-Michel HUFFLEN, and Yves SOULET. 537
- [161] Frank Mittelbach and Chris Rowley. Language Information in Structured Documents: A Model for Mark-up and Rendering. *TUGboat*, 18(3):199–205, September 1997. Available from: <http://www.tug.org/TUGboat/Articles/tb18-3/tb56lang.pdf>.
- [162] Young Joon Moon. How to make and view a Japanese PDF file, June 2002. <mailto:director@research-j.org>. Available from: <http://e-japanese-online.com/english/japanese-computing/platex/Free-Japanese-PDF-authoring-EV003.pdf>.
- [163] Sergei O. Naoumov. LaTeX2e support for LH family of fonts, April 1995. Available from: <http://web.mit.edu/textsrc/source/latex/LH/russian.sty>. 486
- [164] Frank Neukam, Markus Kohm, Axel Kielhorn, and Jens-Uwe Morawski. The KOMA-Script Bundle, March 2005. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/koma-script/scrguien.pdf>. 39, 47, 59, 71, 147, 184, 189, 198, 223, 224, 370
- [165] Heiko Oberdiek. PDF information and navigation elements with hyperref, pdfT_ᵀᵀ_ᵀ, and thumbpdf. In *EuroT_ᵀᵀ_ᵀ'99 Proceedings*. TUG, 1999. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/hyperref/doc/paper.pdf>.
- [166] Tobias Oetiker, Hubert Partl, Helene Hyna, and Elisabeth Schlegl. Une courte (?) introduction à ℒ_ᵀᵀ_ᵀ 2_ᵀ. *Cahiers GUTenberg*, Hors-série numéro 2, February 2003. French translation of [167] by Matthieu Herrb, adaptation and complements by Daniel Flipo. Available from: <http://www.dante.de/tex-archive/info/lshort/french/flshort-3.20.pdf>. 184
- [167] Tobias Oetiker, Hubert Partl, Helene Hyna, and Elisabeth Schlegl. The Not So Short Introduction to ℒ_ᵀᵀ_ᵀ 2_ᵀ, May 2006. Available from: <http://www.dante.de/tex-archive/info/lshort/english/lshort.pdf>. 184, 192, 198, 517
- [168] Mariuz Olko and Marcin Woliński. ℒ_ᵀᵀ_ᵀ 2_ᵀ po polsku czyli PL_ᵀᵀ_ᵀ (wersja 1.3.1), May 2003. Available from: <http://www.tug.org/tex-archive/macros/latex/contrib/platex/polski.dtx>. 41, 208, 478
- [169] Palash Baran Pal. Bangtex: a package for typesetting documents in Bangla using the T_ᵀᵀ_ᵀ/ℒ_ᵀᵀ_ᵀ systems [online]. January 2001. Available from: <http://www.saha.ac.in/theory/palashbaran.pal/bangtex/bangtex.html>. 41, 189, 416, 423
- [170] Minje Byeng-sen Park. *Histoire de l'imprimerie coréenne des origines à 1910*. Maison-Neuve & Larose, Paris, 2003. 451, 454, 455
- [171] Oren Patashnik. B_ᵀᵀ_ᵀXing, February 1988. Documentation for general B_ᵀᵀ_ᵀX users. Available from: <http://www.dante.de/tex-archive/biblio/bibtex/contrib/doc/btxdoc.pdf>. 189
- [172] Oren Patashnik. Designing B_ᵀᵀ_ᵀX styles, February 1988. The part of B_ᵀᵀ_ᵀX's documentation that's not meant for general users. Available from: <http://www.dante.de/tex-archive/biblio/bibtex/contrib/doc/btxhak.pdf>. 189

- [173] Jonathan Pechta, Federico Zenith, Holger Danielsson, and Jeroen Wijnhout. CJK Support [online]. March 2006. Available from: <http://sourceforge.net/Documentation/html/cjk.html>. Part of [175].
- [174] Jonathan Pechta, Federico Zenith, Holger Danielsson, and Jeroen Wijnhout. Document Encoding [online]. March 2006. Available from: <http://sourceforge.net/Documentation/html/lang.html>. Part of [175].
- [175] Jonathan Pechta, Federico Zenith, Holger Danielsson, and Jeroen Wijnhout. The Kile Handbook [online]. March 2006. Available from: <http://sourceforge.net/Documentation/html/>. 518
- [176] Karel Píška. Cyrillic Alphabets. In M. Burbank and C. Thiele, editors, *Proceedings of TUG'96*, pages 1–7, Dubna, 1996. JINR. Available from: <http://www.tug.org/TUGboat/Articles/tb17-2/tb51pisk.pdf>.
- [177] John Plaice and Yannis Haralambous. The latest developments in Ω . *TUGboat*, 17(2):181–183, June 1996. Available from: <http://omega.enstb.org/papers/latest-omega9papers/latest-omega96.pdf>.
- [178] John Plaice and Yannis Haralambous. Typesetting French, German and English in Ω . In *Congrès EuroT_EX'98*, volume 28-29 of *Cahiers GUTenberg*, St. Malo, March 1998.
- [179] Sebastian Rahtz and Heiko Oberdiek. The `hyperref` package [online]. August 2003. Available from: <http://www.tug.org/applications/hyperref>. This page points to the package documentation and its distribution. 59, 69, 197, 201, 213, 221, 241, 374
- [180] Sebastian Rahtz and Heiko Oberdiek. Hypertext marks in \LaTeX : a manual for `hyperref`, February 2004. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/hyperref/doc/manual.pdf>. 121
- [181] Bernd Raichle, Rolf Niepraschk, and Thomas Hafner. Fragen und Antworten (FAQ) über das Textsatzsystem \TeX und DANTE, Deutschsprachige Anwendervereinigung \TeX e.V., September 2003. Version 72. Available from: <http://www.dante.de/faq/de-tex-faq/de-tex-faq.pdf>.
- [182] Adrian Rezuş and Bernd Raichle. Romanian \TeX , October 1996. <mailto:adriaan@cs.kun.nl>. Available from: <http://www.dante.de/tex-archive/language/romanian/RomanianTeX/romanian.dtx>. 480
- [183] Chris Rowley. Models and languages for formatted documents. *TUGboat*, 20(3):189–195, September 1999. Available from: <http://www.tug.org/TUGboat/Articles/tb29-3/tb64rowl.pdf>.
- [184] Young Ryu. The TX Fonts, December 2000. Available from: <http://www.dante.de/tex-archive/fonts/txfonts/doc/txfontsdotA4.pdf>. 34
- [185] Axel Sommerfeldt. The `caption2` package, November 2005. <mailto:caption@sommerfee.de>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/caption/caption2.pdf>. 51, 69, 189, 244

- [186] Axel Sommerfeldt. The `caption` package, January 2006. <mailto:caption@sommerfee.de>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/caption/caption.pdf>. 51, 69, 189, 244
- [187] Axel Sommerfeldt. Setzen von Abbildungs- und Tabellenbeschriftungen mit dem `caption`-Paket, January 2006. <mailto:caption@sommerfee.de>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/caption/anleitung.pdf>.
- [188] Meknavin Surapant, Karoonboonyanan Theppitak, Silpa-Anan Chanop, and Veerathanabutr Poonlap. The `thailatex` package, March 2006. <mailto:surapan@nectec.or.th>. Available from: <http://www.dante.de/tex-archive/language/thailatex/>. 211, 494
- [189] Daniel Taupin⁺. The `varsects` package, November 1998. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/taupin/varsects.sty>. 50, 69, 213, 224, 242
- [190] Thé Hàn Thành. The `pdfTEX` Program. In *EuroT_EX'98 Proceedings*, volume 28-29 of *Cahiers GUTenberg*, pages 197–219, March 1998. Available from: <http://www.gutenberg.eu.org/pub/GUTenberg/publicationsPDF/28-29-han.pdf>.
- [191] Thé Hàn Thành. Micro-typographic extensions to the `TEX` typesetting system. *TUGboat*, 21(4):317–434, October 2000. Doctoral dissertation (Faculty of Informatics, Masarik University, Brno, Czech Republic). Available from: <http://www.tug.org/TUGboat/Articles/tb21-4/tb69thanh.pdf>.
- [192] Thé Hàn Thành, Sebastian Rahtz, Hans Hagen, Hartmut Henkel, and Paweł Jackowski. *The pdfT_EX user manual*, February 2006. Available from: <http://www.dante.de/tex-archive/systems/pdftex/manual/pdftex-a.pdf>.
- [193] Karsten Tinnfeld. The `quotchap` document style, February 1998. tinnfeld@irb.cs.uni-dortmund.de. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/quotchap/>. 50, 72, 209, 225, 242
- [194] Mark Trettin. Une liste des péchés des utilisateurs de `LATEX 2ε` (ou Commandes et extensions obsolètes, et quelques autres erreurs). <mailto:Mark@Trettin@gmx.de>, <mailto:juergen.fen@gmx.de> and <mailto:Yvon.Henel@wanadoo.fr>, June 2004. English translation by Jürgen Fenn, french translation by Yvon Henel. Available from: <http://www.dante.de/tex-archive/info/l2tabu/french/l2tabufr-heavy.pdf>.
- [195] Mark Trettin. An essential guide to `LATEX 2ε` usage (Obsolete commands and packages). <mailto:Mark@Trettin@gmx.de> et <mailto:juergen.fen@gmx.de>, June 2006. English translation by Jürgen Fenn. Available from: <http://www.dante.de/tex-archive/info/l2tabu/english/l2tabuen.pdf>.
- [196] Hideo Umeki. The `geometry` package, July 2002. <mailto:hideo.umeki@toshiba.co.jp>. Available from: <http://www.dante.de/tex-archive/contrib/macros/latex/contrib/geometry/manual.pdf>. 120
- [197] The Unicode Standard [online]. February 2006. Available from: <http://unicode.org>.
- [198] Thomas van Oudenhove. Bibliographie avec `BibTEX`, 2003. Available from: http://www.enstimac.fr/~vanouden/LaTeX/contrib/Pres_Bibtex.tar.gz.

- [199] Suki K. Venkatesan. Moving from bytes to words to semantics. *TUGboat*, 26(2):165–169, August 2005. Proceedings of the 2005 Annual Meeting. 451
 - [200] Carl F. Voegelin and Florence M. Voegelin. *Classification and Index of the World Languages*. Academic Press, 1977.
 - [201] Vladimir Volovich, Werner Lemberg, and L^AT_EX3 Project Team. Cyrillic language support in L^AT_EX, March 1999. Available from: <http://www.dante.de/tex-archive/macros/latex/doc/cyrguide.pdf>. 184
 - [202] Stephan P. von Bechtolsheim. T_EX in practice: Comments on a 4-volumes, 1400-pages series on T_EX. *TUGboat*, 11(3):409–412, September 1990. <http://www.tug.org/TUGboat/Articles/tb11-3/tb29bechtolsheim.pdf>.
 - [203] Stephan P. von Bechtolsheim. *T_EX in Practice*. Springer Verlag, Berlin, Germany / Heidelberg, Germany / London, UK / etc., 1992.

1. Basics.	3. Tokens, Macros.
2. Paragraphs, Maths and Fonts.	4. Output Routines.
- (Une approche progressive des secrets de T_EX). 245
- [204] Staszek Wawrykiewicz. PL^AT_EX i pdf^LA_TE_X, May 2005. In polish, <mailto:staw@gust.org.pl>. Available from: <http://www.gust.org.pl/doc/poczatki/platex.html>. 41, 208, 478
 - [205] Peter R. Wilson. The stdclsdv package, July 2000. <mailto:herries.press@earthlink.net>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/stdclsdv/stdclsdv.pdf>. 209, 210
 - [206] Peter R. Wilson. The needspace package, January 2003. <mailto:herries.press@earthlink.net>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/misc/needspace.sty>. 54, 203
 - [207] Peter R. Wilson. The tocloft package, September 2003. <mailto:herries.press@earthlink.net>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/tocloft/tocloft.pdf>. 42, 61, 68, 120, 127, 189, 212, 371
 - [208] Peter R. Wilson. The abstract package, April 2004. <mailto:herries.press@earthlink.net>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/abstract/abstract.pdf>. 50, 67, 186, 223, 374
 - [209] Peter R. Wilson. The appendix package, April 2004. <mailto:herries.press@earthlink.net>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/appendix/appendix.pdf>. 60, 188, 222, 370
 - [210] Peter R. Wilson. The tocbibind package, May 2004. <mailto:herries.press@earthlink.net>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/tocbibind/tocbibind.pdf>. 47, 96, 101, 105, 118, 186, 187, 211, 212, 221, 370
 - [211] Peter R. Wilson. The tocvsec2 package, May 2004. <mailto:herries.press@earthlink.net>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/tocvsec2/tocvsec2.pdf>.

- [212] Peter R. Wilson. The `ccaption` package, March 2005. <mailto:herries.press@earthlink.net>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/ccaption/ccaption.pdf>. 51, 69, 190, 244
- [213] Peter R. Wilson. The Memoir Class, September 2005. <mailto:herries.press@earthlink.net>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/memoir/memman.pdf>. 62, 69, 200, 222, 223, 245, 254, 373
- [214] Peter R. Wilson. The Memoir Class (Addendum), December 2005. <mailto:herries.press@earthlink.net>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/memoir/memmanadd.pdf>. 62, 69, 200, 222, 223, 245, 254, 373
- [215] Peter R. Wilson. The `romannum` package, May 2005. <mailto:herries.press@earthlink.net>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/romannum/romannum.pdf>. 50, 72, 209, 225, 243
- [216] Peter R. Wilson and Donald Arseneau. The `ifmarg` package, March 2000. <mailto:herries.press@earthlink.net>, <mailto:asnd@triumf.ca>. Available from: <http://www.dante.de/tex-archive/macros/latex/contrib/misc/ifmarg.sty>. 197, 245
- [217] Vadim V. Zhytnikov. The LHCYR collection [online]. December 1998. Available from: <http://ftp.dante.de/tex-archive/macros/latex/contrib/supported/lhcyr/>. <mailto:vvzhy@td.lpi.ac.ru>. 486, 487

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 (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 `\minitableofcontents` command, so if someboby 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.sty` (external references). `xr.sty` version 5 is much more tolerant.

★ **version 02**

- 1993/07/05
 - Added compatibility with `hangcaption.sty` (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 file `mtcoeff.sty` allows you to use a \LaTeX document with `minitoc` commands and to make them transparent: just replace the `minitoc` option by `mtcoeff`.

★ **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 better) added compatibility with `toch.sty` (`toch.sty` makes a table of chapters; if used, must be loaded *before* `minitoc.sty`).

★ **version 07**

- 1993/07/22
 - (*major differences*) Completely rewritten, using tricks from `xr.sty` (the version 5, by David 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.

★ **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 `latex.tex` (`\@inputcheck` or `\reset@font` not defined).

★ version 11

- 1993/08/18
 - Added `\mtcSfont`, font for section entries, `\mtcSSfont` for subsection entries, `\mtcSSSfont` for subsubsection entries, `\mtcPfont` for paragraph entries, `\mtcSPfont` for subparagraph entries.

★ version 12

- 1993/12/16
 - Use `\kern` in place of `\vspace*`, and added penalties (`\nopagebreak`) to avoid a page break just before last `\mtc@rule`.
 - Also added a `samepage` environnement.
 - Removed old commented out lines from previous versions.

★ version 13

- 1993/12/17
 - Added `minilof` and `minilot` stuff. For MS-DOS, uncomment the definition of `\SHORTEXT`.

★ version 14

- 1994/01/03
 - Corrected space under `minitoc/lof/lot` and added a `\raggedright` setting to avoid “underfull” warnings.
 - Corrected some spacing problems (avoiding ~’s). `\mtifont` is changed from `\normalsize\bf` to `\large\bf`.
 - Some modifications suggested by Donald ARSENEAU (thanks): `\@newread` becomes `\newread`, not outer version of `\newread`; `\empty` replaced by `\relax` in the spare definition of `\reset@font`.
 - Removed the setting of `\clubpenalty` and `\widowpenalty` to 10 000 (done by `\samepage`), and `\noindent`.
 - Simplified processing of optional argument in `\minitoc`, `\minilof` and `\minilot`.

★ 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 “ $\text{\LaTeX 2}_{\epsilon}$ ”. Thanks to Denis ROEGEL (who found the problem) and Frank MITTELBACH (who gave the hints to solve).
- ★ **version 18**
 - 1994/06/26
 - Make `minitoc` really compatible with $\text{\LaTeX 2}_{\epsilon}$.
 - 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 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 `mtcswedish.sty` (Jan Michel 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.
 - 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.sty` and `minitoc.tex`.

★ version 22

- 1994/10/10
 - Corrected typos in `minitoc.sty`.

★ 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 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 L^AT_EX 2_ε release of December 1994. Thanks to Denis ROEGEL who did the work.

★ version 25

- 1996/09/13
 - Updated `mtcnorsk.sty` and added `mtcnynorsk.sty` on a suggestion from Dag LANGMYHR.

★ version 26

- 1996/11/14
 - Language specific files are now named `⟨language⟩.mld` (replacing `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.

★ version 27

- 1996/12/20
 - Corrections for starred sectionning commands.
 - `english.mld` loaded as default language.
 - Added `vietnam.mld` and `arab.mld`.
 - Renamed `minitocoff.sty` into `mtcoff.sty` 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`).
 - 1999/07/23
 - Added the dotted/undotted package options (default: dotted).
 - 1999/07/29
 - Added the lithuanian language.
- ★ version 30
 - 1999/12/06
 - Added the basque, ngermanb, serbian, ukraineb, and welsh languages.
 - Corrected a bug in `\sltname` definition (`mlt` should be `s1t`).
- ★ 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

- 2000/12/13
 - Added in the documentation a section for use with the `tocbibind` package.
 - Added `.mld` files for alternate names of languages: so, `american.mld` just loads `english.mld`, which contains the real definitions.

★ version 35

- 2001/01/09
 - Added macros to test if a file is “empty” (i.e., empty, blank or inexistent) or “non empty” (i.e., useful). I used some code from Stephan P. von BECHTOLSHEIM.
 - Added the `checkfiles/nocheckfiles` package options.
 - Replaced `\The@chapter` by `\The@mtc`.
- 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.

★ 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 `0pt` 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` languages (synonyms of french).
 - Corrected `secttocs`, at least.
- 2003/03/18
 - Corrected some vertical spacings and struts (I added some mods by Frank MITTELBACH, 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 ARSENEAU); 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 `castillan` 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 `fminitoc.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 `fminitoc.pdf` (french documentation in PDF format).
- ★ version 41
- 2005/01/05
 - Corrections in documentation.
 - Message added if some sectionning commands are not available.
 - Replaced `\typeout` commands in `minitoc.sty` 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 `.log` file, not on the terminal).
 - 2005/01/06
 - Added the `\mtcsetfont` (Benjamin BAYART) and `\mtcsetttitlefont` commands, with a much simpler syntax.

- 2005/01/10
 - Added bibliography.
- 2005/01/11
 - \mathcal{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 coherence of `\dominitoc/\minitoc` and `co`.
 - Improved documentation about hints.
- 2005/01/20
 - Added a hint about using short extensions with more than 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
 - Added many new language files: `serbianc.mld`, `chinese1.mld`, `chinese2.mld`, `hangul1.mld`, `hangul2.mld`, `hangul3.mld`, `hangul4.mld`, `hanja1.mld`, `hanja2.mld`, `japanese2.mld`, `japanese3.mld`, `japanese4.mld`, `japanese5.mld`, `thai.mld`.
- 2005/02/02
 - Added `\mtcsetrules`.
- 2005/02/03
 - Added `\plfrule`, `\noplfrule`, `\mlfrule`, `\nomlfrule`, `\slfrule`, `\noslfrule`, `\pltrule`, `\nopltrule`, `\mltrule`, `\nomltrule`, `\sltrule`, `\nosltrule`.
- 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 big bug in `\mtcsetformat`.
 - Fixed a bug in `mtcoeff.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 coherence of `\dominitoc`/`\minitoc` and `co`.
- 2005/03/08
 - Added a hint about coherence 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 secttocs (...).
- 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 portugues.mld.
- 2005/04/12
 - Correction in \mtcskip.
 - First version in .dtx 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.sty (Don HOSEK) to allow more marginpars and floats.
 - Added minitoc.ist to format the index correctly.
- 2005/05/26
 - 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 `fminitoc.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 `fminitoc.dtx` is now generated by `minitoc.ins`.
- 2005/06/21
 - Added “OUT”, “NON”, “oui”, “non”, “O”, and “o” as true/false keywords.
 - Compacted the code about detection of short/long extensions.
- 2005/06/22
 - Added “VRAI”, “FAUX”, “vrai”, “faux”, “V”, and “v” as true/false keywords.
- 2005/06/23
 - Correctly set the `\iffTR` flag to have the names of months in the right language in the bibliography.
- 2005/06/29
 - Set the flag `\mtcoffwarn@true` in `mtcoff.sty` 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.sty`.
 - Improved some messages in `mtcoff.sty`.
 - 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 `minitocs`. Updated `minitoc.bug`.
 - Made two versions of the `mtchideinmainlof` and `mtchideinmainlot` environments, depending of 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`.
 - 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 script.
- 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 `memoir.cls` only if necessary; do not load it if `memoir.cls` is dated after 2005/09/25.
 - 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^AT_EX Companion [160].
- 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`, `\pltSfont`, `\mlfSfont`, `\mltSfont`, `\slfSfont`, and `\sltSfont` for subfigures and subtables entries in the mini-tables.

- 2005/11/16
 - Continuing the “Jargon” chapter.
 - Fixed a bug about fonts for subfigures and subtables entries in the mini-tables.
 - Added `bicig2.mld`, `bi the.mld`, `manju.mld`, `xalx.mld`, and `khalkha.mld`.
- 2005/11/17
 - Continuing the “Jargon” chapter.
 - Added testing via internal *quarks* commands in `\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.
 - Added `mailto:` in the mailing URLs.
- 2005/12/05
 - Fixed typos in the documentation.
 - Continuing the “Jargon” chapter.
 - Added a hint about the `varsects` package (must be loaded *before* `minitoc`).
- 2005/12/06
 - Continuing the “Jargon” chapter.
 - Correcting an hyperlink in the bibliography (for the `xr` package).
 - Attempting to avoid broken URLs, using `quote`, footnotes and `\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 \mathcal{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 `addsec.tex`

- 2006/01/04
 - 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 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`.
- 2006/01/16
 - Correction in `\mtcaddsection`.
- 2006/01/17
 - Correction in `\mtcfixindex` and `\mtcfixglossary`.
 - 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, 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
 - Corrections in the “Postface” chapter.
 - Updated the bibliography.
- 2006/02/06
 - Corrections in the documentation.
 - Updated the bibliography.
 - Added `dblacnt.sty` for the “The pdfTeX Program” entry in the bibliography. Its author’s first name needs a double accent (Hàn Thê THÀNH).
- 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.mld` into `hangul-u8` and `hanja-u8`.
 - Updated the bibliography.

- 2006/02/22
 - 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 the position of tables in the “Jargon” chapter.
 - 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` [103].
- 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-lh.mld`, with their `.mlo` files.
- 2006/03/09
 - Update the bibliography.
- 2006/03/10
 - Update the bibliography.
 - Added `russian-lhcyralt.mld`, `russian-lhcyrkoi.mld`, and `russian-lhcyrwin.mld`, with their `.mlo` files.
- 2006/03/13
 - Corrections in the documentation.
- 2006/03/14
 - Added `mtcmess.sty`.
- 2006/03/16
 - The messages are now numbered.

- Update the bibliography.
- 2006/03/20
 - 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.sty` to better format section titles.
- 2006/04/07
 - Corrections in the documentation.
 - Added `\ifmtcsecondpart` 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
 - Corrections in the documentation.
- 2006/04/13
 - In the `insection` package option, load the `flafter` package *before* the `placeins` package.
 - Added a figure about the float barriers.
- 2006/04/27
 - Added notes in FAQ 20, about the use with the `appendix` package.
 - Added comments about the `insection` option.
 - Update the bibliography.
 - Begin correction of the `mtchideinmaintoc` environment.
- 2006/05/02
 - Added notes in `minitoc.bug`, point 20, about the use with the `appendix` package.
 - End correction of the `mtchideinmaintoc` environment.
 - Analogous corrections in the `mtchideinmainlof` and `mtchideinmainlot` environments.
 - Update the bibliography.
- 2006/05/03
 - Added notes about the `mtchideinmainlof` and `mtchideinmainlot` environments.
 - Added `hide1.tex` and `hide2.tex`.
 - Added a hint about the `fncychap` package (must be loaded *before* `minitoc`).
 - Added a hint about the `quotchap` package (must be loaded *before* `minitoc`).
 - Update the bibliography.
- 2006/05/04
 - Update the bibliography.
 - Added a hint about the `romannum` package (must be loaded *before* `minitoc`).
 - Added a hint about the `sfheaders` package (must be loaded *before* `minitoc`).
 - Added a hint about the `alnumsec` package (must be loaded *before* `minitoc`).
 - Corrections in the documentation.
- 2006/05/05
 - Corrections in the documentation.
- 2006/05/24
 - Corrections in the documentation.

- Renamed `hide.tex` to `hide1.tex`.
 - Update the bibliography.
- 2006/05/30
 - Corrections in the documentation.
 - Use `\MakeUpperCase` in `\markboth` for page styles.
- 2006/05/31
 - Update the bibliography.
 - 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 `imk` script in `INSTALL` and the “Installation” chapter.
 - Added an entry about “package” in the jargon.
 - Added the `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 [106] in some of the examples of documents.
 - Update the bibliography.
- 2006/06/27
 - Update the examples of documents.
 - Added the `second.tex` example file.
- 2006/06/29
 - Added the `app-mem.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 `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 `hie.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.mtc1` in the list of files.
 - In the scripts, the backup directory (OLD) is now `/tmp/username/OLD`.
 - In the scripts, the repartition directories (`CL[0-9]`) are now `/tmp/username/CL[0-9]`.
- ★ 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 ~~Arabi~~ system).
 - Update the bibliography.
 - 2006/07/28
 - Update the jargon.
 - 2006/07/31
 - 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 some example documents by using the `report` class in place of the `book` class (hence using one side printing).

Index

Symbols

.lof	249	\@dopartlot@used@false	92
.log	234	\@dopartlot@used@true	1628
.lot	249	\@doparttoc	1570, 1636
.maf@.maf	49, 365, 392	\@doparttoc@used@false	90
.mld	35, 41, 67, 214, 412, 415, 416	\@doparttoc@used@true	1594
.mlo	41, 214, 362, 412, 416	\@dosectlof	2187, 2250
.toc	249	\@dosectlof@used@false	103
\@@dominilof	912, 946	\@dosectlof@used@true	2224
\@@dominilot	929, 946	\@dosectlot	2191, 2251
\@@dominitoc	880, 895, 946	\@dosectlot@used@false	104
\@@dopartlof	1618, 1636	\@dosectlot@used@true	2241
\@@dopartlot	1635, 1636	\@dosecttoc	2183, 2249
\@@doparttoc	1601, 1636	\@dosecttoc@used@false	102
\@dosectlof	2231, 2249	\@dosecttoc@used@true	2207
\@dosectlot	2248, 2249	\@dotsep	2825, 2838, 2843, 2848, 2853, 2858, 2863, 2868, 2873, 2878
\@dosecttoc	2214, 2249	\@dottedtocline	647, 720, 788, 803, 849, 861, 864, 867, 872, 875, 878, 1384, 1461, 1536, 1552, 1972, 2042, 2109, 2123
\@M	373, 803, 826, 839, 1552, 1727, 1737, 2123, 2124, 2156	\@dottedxxxline	1438
\@Undottedtocline	821, 3374, 4641	\@ensure@R	5937, 5940, 5943, 5947, 5950, 5953, 5957, 5960, 5963
\@Undottedtoclinep	834, 849, 861, 864, 867, 872, 875, 878, 4641	\@fileswfalse	651, 724, 792, 1391, 1465, 1540, 1977, 2046, 2114
\@addtoreset	224, 484	\@firstchapteris@used@false	106
\@afterheading	1270, 1284, 1298, 4723, 4731	\@firstchapteris@used@true	546
\@afterindenttrue	372, 825, 838	\@firstoftwo	205
\@centercr	586, 1319, 1911	\@firstpartis@used@false	105
\@chapter	1, 309, 805, 3756, 4334, 4697, 4699	\@firstpartis@used@true	1219
\@dominilof	885, 947	\@firstsectionis@used@false	107
\@dominilof@used@false	97	\@firstsectionis@used@true	1882
\@dominilof@used@true	921	\@gobble	4523, 4526, 4527, 4641
\@dominilot	890, 948	\@gobbletwo	4463, 4508
\@dominilot@used@false	98	\@hangfrom	2146
\@dominilot@used@true	938	\@ifclasslater	197
\@dominitoc	880, 946	\@ifclassloaded	176, 3611, 3699, 3706, 3716, 3734
\@dominitoc@used@false	96	\@ifnextchar	597, 667, 735, 884, 889, 894, 1330, 1403, 1477, 1582–1584, 1922, 1988, 2055, 2195–2197, 2467, 4444, 4450, 4456, 4462, 4697
\@dominitoc@used@true	904		
\@dopartlof	1574, 1637		
\@dopartlof@used@false	91		
\@dopartlof@used@true	1611		
\@dopartlot	1578, 1638		

- \@ifpackagelater 3689
- \@ifpackageloaded . . 135, 152, 154, 156,
158, 160, 162, 164, 166, 168, 170,
172, 174, 3594, 3599, 3605, 3629,
3635, 3640, 3723, 4272
- \@ifpackagewith 3640, 3723
- \@ifundefined 17, 2467, 2932, 3001, 3260,
3300, 3378, 3394, 3739, 3756, 3773,
4331, 4334, 4337, 4361
- \@input 1036, 1102, 1161, 3340, 3353
- \@inputcheck 227,
358, 1022, 1032, 1052, 1089, 1098,
1115, 1148, 1157, 1174, 1719, 1720,
1729, 1748, 1780, 1781, 1790, 1808,
1840, 1841, 1850, 1868, 2319, 2320,
2329, 2345, 2378, 2379, 2388, 2404,
2437, 2438, 2447, 2463
- \@longextensions@false . 334, 339, 3391
- \@longextensions@true 336, 4591
- \@m@nchapter 4695
- \@makechapterhead 1724, 4722, 4727, 4729
- \@makephead@c 1298
- \@makephead@l 1270
- \@makephead@r 1284
- \@makeschapterhead 1734
- \@minilof@used@false 94
- \@minilof@used@true 669
- \@minilot@used@false 95
- \@minilot@used@true 737
- \@minitoc@used@false 93
- \@minitoc@used@true 599
- \@mkboth 517, 1345, 1418, 1493
- \@mtc@FEfalse 238, 251
- \@mtc@FEtrue 228, 242, 249
- \@mtc@Himlof@false 3194, 3225
- \@mtc@Himlof@true 3201, 3219
- \@mtc@Himlot@false . . . 3227, 3242, 3258
- \@mtc@Himlot@true 3234, 3252
- \@mtc@Himtoc@false . . . 3179, 3192, 3209
- \@mtc@Himtoc@true 3186
- \@mtc@LIfalse 232, 236, 238
- \@mtc@LItrue 227, 242
- \@mtc@abstract@loaded@false . . . 3586
- \@mtc@abstract@loaded@true . . . 3724
- \@mtc@alnumsecLoaded@a@false 63
- \@mtc@alnumsecLoaded@a@true 165
- \@mtc@alnumsecLoaded@false 62
- \@mtc@alnumsecLoaded@true 164
- \@mtc@captcontLoaded@a@false 65
- \@mtc@captcontLoaded@a@true 167
- \@mtc@captcontLoaded@false 64
- \@mtc@captcontLoaded@true 166
- \@mtc@captionIIloaded@a@false 47
- \@mtc@captionIIloaded@a@true 171
- \@mtc@captionIIloaded@false 46
- \@mtc@captionIIloaded@true 170
- \@mtc@captionLoaded@a@false 45
- \@mtc@captionLoaded@a@true 169
- \@mtc@captionLoaded@false 44
- \@mtc@captionLoaded@true 168
- \@mtc@ccaptionLoaded@a@false 49
- \@mtc@ccaptionLoaded@a@true 173
- \@mtc@ccaptionLoaded@false 48
- \@mtc@ccaptionLoaded@true 172
- \@mtc@chapter@def@false 77, 78
- \@mtc@chapter@def@true 78
- \@mtc@chapter@undef@false 84
- \@mtc@chapter@undef@true 83, 84
- \@mtc@checkfilesfalse 3387
- \@mtc@checkfilestrue 229, 3386
- \@mtc@empty@minilof@false 70
- \@mtc@empty@minilof@true 679
- \@mtc@empty@minilot@false 71
- \@mtc@empty@minilot@true 747
- \@mtc@empty@minitoc@false 69
- \@mtc@empty@minitoc@true 609
- \@mtc@empty@partlof@false 67
- \@mtc@empty@partlof@true . . 1415, 1490
- \@mtc@empty@partlot@false 68
- \@mtc@empty@parttoc@false 66
- \@mtc@empty@parttoc@true 1342
- \@mtc@empty@sectlof@false 73
- \@mtc@empty@sectlof@true 2000
- \@mtc@empty@sectlot@false 74
- \@mtc@empty@sectlot@true 2067
- \@mtc@empty@secttoc@false 72
- \@mtc@empty@secttoc@true 1934
- \@mtc@fncychapLoaded@a@false 55
- \@mtc@fncychapLoaded@a@true 157
- \@mtc@fncychapLoaded@false 54
- \@mtc@fncychapLoaded@true 156
- \@mtc@hints@false 3585
- \@mtc@hints@given@false 37
- \@mtc@hints@given@true 895, 912,
929, 1585, 1602, 1619, 2198, 2215,
2232, 2487, 3595, 3600, 3606, 3612,
3618, 3624, 3630, 3636, 3643, 3659,
3674, 3690, 3700, 3707, 3712, 3717,
3726, 3735, 3746, 3763, 3781, 3829,
3838, 3846, 3854, 3864, 3872, 3880,
3890, 3898, 3915, 3923, 3931, 3941,
3949, 3957, 3967, 3975, 3994, 4003,
4012, 4023, 4032, 4041, 4052, 4061,
4070, 4079, 4087, 4095, 4109, 4117,
4125, 4133, 4141, 4149, 4157, 4165,
4173, 4181, 4189, 4197, 4203, 4208,
4213, 4218, 4223, 4228, 4233, 4238,
4243, 4248, 4253, 4258

- \@mtc@hints@true 35, 3584
- \@mtc@hints@w@false 36, 3739, 3756, 3773
- \@mtc@hints@w@true .. 3741, 3743, 3745, 3758, 3760, 3762, 3776, 3778, 3780, 3829
- \@mtc@hyper@used@false 136
- \@mtc@hyper@used@true 139
- \@mtc@insellang@false 3340, 3371
- \@mtc@listfiles@false 3411, 3413
- \@mtc@listfiles@true 3412
- \@mtc@lof@used@false 3588, 3793
- \@mtc@lof@used@true 3800, 3809
- \@mtc@lot@used@false 3589, 3794
- \@mtc@lot@used@true 3803, 3812
- \@mtc@mcaptionLoaded@a@false 51
- \@mtc@mcaptionLoaded@a@true 175
- \@mtc@mcaptionLoaded@false 50
- \@mtc@mcaptionLoaded@true 174
- \@mtc@memoirLoaded@false 40, 181
- \@mtc@memoirLoaded@true 177
- \@mtc@memoirnew@false 41, 184
- \@mtc@memoirnew@true 186
- \@mtc@notoccite@false 130
- \@mtc@notoccite@true 3390
- \@mtc@part@def@false 75, 76
- \@mtc@part@def@true 76
- \@mtc@part@undef@false 82
- \@mtc@part@undef@true 81, 82
- \@mtc@placeinsLoaded@false .. 39, 4276
- \@mtc@placeinsLoaded@true 4276
- \@mtc@quotchapLoaded@a@false 57
- \@mtc@quotchapLoaded@a@true 159
- \@mtc@quotchapLoaded@false 56
- \@mtc@quotchapLoaded@true 158
- \@mtc@romannumLoaded@a@false 59
- \@mtc@romannumLoaded@a@true 161
- \@mtc@romannumLoaded@false 58
- \@mtc@romannumLoaded@true 160
- \@mtc@sect@floats@false 38
- \@mtc@sect@floats@true 3816–3819
- \@mtc@section@def@false 79, 80
- \@mtc@section@def@true 80
- \@mtc@section@undef@false 86
- \@mtc@section@undef@true 85, 86
- \@mtc@sectstyLoaded@a@false 43
- \@mtc@sectstyLoaded@a@true 153
- \@mtc@sectstyLoaded@false 42
- \@mtc@sectstyLoaded@true 152
- \@mtc@setdepth@false 3146
- \@mtc@setdepth@true 3141, 3144
- \@mtc@setfeature@false 3094, 3110
- \@mtc@setfeature@true 3088, 3092
- \@mtc@setfont@false 2600, 2617, 2655–2658, 2661, 2662, 2665, 2666, 2669–2674, 2677–2680, 2683–2686, 2689–2695, 2698–2702, 2705–2709
- \@mtc@setfont@true ... 2595, 2598, 2653
- \@mtc@setformat@false 2886, 2902
- \@mtc@setformat@true 2880, 2884
- \@mtc@setpagenumbers@false 2930, 2955, 2975
- \@mtc@setpagenumbers@true 2952
- \@mtc@setrules@false .. 2999, 3024, 3045
- \@mtc@setrules@true 3021
- \@mtc@settitle@false 2767
- \@mtc@settitle@true 2762, 2765
- \@mtc@settitlefont@false 2731
- \@mtc@settitlefont@true ... 2726, 2729
- \@mtc@sfheadersLoaded@a@false 61
- \@mtc@sfheadersLoaded@a@true 163
- \@mtc@sfheadersLoaded@false 60
- \@mtc@sfheadersLoaded@true 162
- \@mtc@spn@ok@false 2972
- \@mtc@spn@ok@true 2970
- \@mtc@sru@ok@false 3042
- \@mtc@sru@ok@true 3040
- \@mtc@ss@insection@false 3394
- \@mtc@ss@insection@true 3408
- \@mtc@toc@used@false 3587, 3792
- \@mtc@toc@used@true 3797, 3806
- \@mtc@varsectsLoaded@a@false 53
- \@mtc@varsectsLoaded@a@true 155
- \@mtc@varsectsLoaded@false 52
- \@mtc@varsectsLoaded@true 154
- \@mtcfixglossary@false 3278, 3280
- \@mtcfixglossary@true .. 3282, 3284, 3286
- \@mtcfixindex@false 3318, 3320
- \@mtcfixindex@true ... 3322, 3324, 3326
- \@mtcpatchmemoir@false 176
- \@mtcpatchmemoir@true 193
- \@namedef .. 2532, 2541, 2550, 2584, 2798
- \@nameuse . 2532, 2597, 2615, 2728, 2764, 2899, 2901, 2968, 2985, 3020, 3055, 3107, 3109, 3143
- \@nobreakfalse . 510, 656, 729, 797, 1396, 1470, 1545, 1982, 2049, 2117
- \@part 1, 264, 1557, 3739, 4331
- \@partlof@used@false 88
- \@partlof@used@true 1405
- \@partlot@used@false 89
- \@partlot@used@true 1480
- \@parttoc@used@false 87
- \@parttoc@used@true 1332
- \@plus 823, 836
- \@pnumwidth . 846, 2807, 2836, 2841, 2846, 2851, 2856, 2861, 2866, 2871, 2876
- \@resetfont 1177, 1197
- \@schapter 1, 810, 3756, 4334

- \@secondoftwo 205
- \@sect 1, 297, 2129, 2136, 2175, 3773, 4337
- \@sectlof@used@false 100
- \@sectlof@used@true 1990
- \@sectlot@used@false 101
- \@sectlot@used@true 2057
- \@secttoc@used@false 99
- \@secttoc@used@true 1924
- \@spart 1, 264, 1557, 3739, 4331
- \@ssect 1, 297, 3773, 4337
- \@svsec 2146
- \@svsechd 2146
- \@tempdima 375, 828, 829, 841, 842
- \@tempskipa 2146
- \@tocfile 600, 605, 608, 653,
670, 675, 678, 726, 738, 743, 746,
794, 1331, 1338, 1384, 1404, 1411,
1461, 1478, 1486, 1536, 1923, 1930,
1972, 1989, 1996, 2042, 2056, 2063,
2109
- \@tocrmarg 371, 824, 837, 2816, 2837, 2842,
2847, 2852, 2857, 2862, 2867, 2872,
2877
- \@topnewpage 1270, 1284, 1298, 4727
- \@undottedtocline 367, 647,
720, 788, 1384, 1461, 1536, 1972,
2042, 2109
- \@width 555
- \@xsect 2146
- \\ . 586, 645, 657, 715, 730, 783, 798, 1319
- \\@mainaux 115
- \~ 7143, 7147, 7151
- 2c.tex **119, 216**

- _ 5072, 5073, 5075, 5076, 5081, 5082, 5084,
5085, 5090, 5091, 5093, 5094, 5462,
5463, 5466, 5467, 5470, 5471, 5911,
5913, 5915, 5918, 5920, 5922, 5925,
5927, 5929, 5938, 5941, 5944, 5948,
5951, 5954, 5958, 5961, 5964, 7093,
7095, 7099, 7101, 7105, 7107

- A**
- \a@tok 4823
- abstract **50, 67**
- abstract (environment) 1, 174
- acadian.mld **416**
- acadien.mld **417**
- \accent 5233
- add.bib **216**
- add.tex **48, 96, 216**
- \addappheadtotoc 1221
- \addcoffeeline 3374, 4641
- \addcontentsline 1,
28, 259, 274, 301–304, 312–315,
358, 361, 364, 367, 370, 514, 519,
779, 805, 834, 835, 1120, 1121,
1559–1561, 1625, 1628, 2045, 2129,
2136, 2146, 3278, 3318, 4614, 4660,
4736, 4739, 4743, 4746
- addsec.tex **48, 101, 216**
- \addst@rred 519, 538, 815
- \addstarredchapter .. 1, 355, 815, 4660
- \addstarredpart 1, 815, 4660
- \addstarredsection 1, 538, 599, 815, 4660
- \addtocontents .. 1, 455, 810, 1557, 2175,
3180, 3194, 3227, 3374
- \addtocounter 243, 261, 549–
551, 1036, 1102, 1161, 1222–1224,
1746, 1806, 1866, 1884–1886, 2343,
2402, 2461
- \addtolength 1705, 1708, 2017
- addtotoc **50, 67**
- \addvspace 13, 14
- \adjustmtc .. 1, 311, 542, 831, 1214, 1637,
4424
- \adjustptc .. 1, 29, 274, 1212, 1955, 4424
- \adjuststc 1, 300, 1117, 1879, 4424
- afrikaan.mld **417**
- afrikaans.mld **417**
- \afterminilof 478, 730, 4573
- \afterminilot 478, 798, 4573
- \afterminitoc 478, 657, 4573
- \afterpartlof 471, 1472, 4564
- \afterpartlot 471, 1547, 4564
- \afterparttoc 471, 1398, 4564
- \aftersectlof 481, 2051, 4582
- \aftersectlot 481, 2119, 4582
- \aftersecttoc 481, 1984, 4582
- albanian.mld **418**
- \alef .. 4842–4844, 4846–4848, 4850–4852,
5461–5463, 5465–5467, 5469–5471,
5914, 5916, 5921, 5923, 5928, 5930
- \alefhamza 4843, 4847, 4851
- \alloc@ 318
- alnumsec **51, 73, 243, 386**
- \Alph 1809, 1810
- alphanum **50, 71**
- american.mld **418**
- amsart **50**
- amsart.cls **63**
- amsbook **50**
- amsbook.cls **63**
- amsproc **50**
- amsproc.cls **63**
- app-mem.tex **108, 216**
- appendices (environment) 1

- appendix **50, 60**
 \appendix . 1, 27, 424, 677, 839, 1124, 1424
 \appendixmtc 4463
 apx.tex **64, 216**
 arab.mld **418, 420**
 arab2.mld **419**
 arabi.mld **419**
 \arabic 956, 1063, 1122, 2290, 3180, 3194,
 3227
 arabic.mld **420**
 armenian.mld **420**
 article **248**
 \AtBeginDocument 109, 135, 152, 154, 156,
 158, 160, 162, 164, 166, 168, 170,
 172, 174, 511, 1206, 1871, 2806,
 3194, 3227, 3378, 3586, 3739, 4327,
 4597
 \AtEndDocument . . 115, 3586, 3815, 4326,
 4327, 4598
 australian.mld **421**
 austrian.mld **421**
 \author 2025
 \ayin 5912, 5919, 5926
- B**
- bahasa.mld **421**
 bahasai.mld **421, 422, 458**
 bahasam.mld **422, 469, 472**
 bangla.mld **423**
 bangtex **423**
 \baselinestretch 1800
 basque.mld **423**
 \bcg . . . 4961–4963, 4965–4967, 4969–4971
 \beforeminilof 465, 683, 4573
 \beforeminilot 465, 751, 4573
 \beforeminitoc 465, 613, 4573
 \beforepartlof 458, 1417, 4564
 \beforepartlot 458, 1492, 4564
 \beforeparttoc 458, 1344, 4564
 \beforesectlof 468, 2002, 4582
 \beforesectlot 468, 2069, 4582
 \beforesecttoc 468, 1936, 4582
 \bet 5916, 5923, 5930
 \bfseries 564, 573, 1235, 1244, 1248, 1258,
 1315, 1318, 1361, 1364, 1449–1453,
 1455–1458, 1729, 1731, 1755, 1758,
 1780, 1785, 1789, 1792, 1891, 1899,
 2021
 \bibitem 1634, 2304, 2312
 \bibliography 1, 830, 1116
 \bibliographystyle 829, 1115
 bicig.mld **424, 495, 496**
 bicig2.mld **424, 495**
 bicig3.mld **425**
- \bigskipamount 1, 11, 4349
 bithe.mld **426, 472**
 bo.tex **120, 216**
 book **248, 250**
 brazil.mld **426, 427**
 brazilian.mld **427**
 breton.mld **427**
 british.mld **428**
 bulgarian.mld **428**
 bulgarianb.mld **429**
 buryat.mld **429**
 buryat2.mld **430**
- C**
- \c@lofdepth . 511, 715, 1206, 1458, 1459,
 1871, 2039, 2040
 \c@lotdepth . 511, 783, 1206, 1533, 1534,
 1871, 2106, 2107
 \c@minilofdepth 715
 \c@minilotdepth 783
 \c@minitocdepth 645, 4349
 \c@mtc . . 519, 4299, 4303, 4316, 4320, 4356
 \c@mti . . 575, 613, 683, 751, 895, 912, 929
 \c@page 2000
 \c@partlofdepth 1458
 \c@partlotdepth 1533
 \c@parttocdepth 1382, 4349
 \c@ptc . . 519, 4293, 4297, 4311, 4315, 4356
 \c@pti 1260, 1298, 1351, 1424, 1499, 1585,
 1602, 1619
 \c@secnumdepth . . 1728, 2146, 2159, 2168,
 4715, 4733
 \c@sectlofdepth 2036
 \c@sectlotdepth 2103
 \c@secttocdepth 1970, 4349
 \c@stc . . 519, 4304, 4308, 4321, 4325, 4356
 \c@sti 1900, 1938, 2005, 2072, 2198, 2215,
 2232
 \c@tocdepth 369, 645, 822, 835, 1382, 1970,
 2036, 2103
 canadian.mld **430**
 canadien.mld **431**
 captcont **51, 73, 243, 387**
 caption **51, 69, 244, 387**
 \caption 65, 68, 71, 74, 83, 86,
 89, 92, 101, 104, 107, 110, 127, 130,
 133, 136, 143, 146, 151, 154, 161,
 164, 167, 170, 331, 337, 350, 391,
 415, 440, 580, 593, 635, 661, 692,
 748, 751, 769, 772, 813, 816, 822,
 825, 850, 853, 859, 862, 874, 877,
 883, 886, 950, 953, 971, 974, 1010,
 1013, 1019, 1022, 1037, 1040, 1059,
 1062, 1099, 1102, 1108, 1111, 1131,

- 1134, 1140, 1143, 1153, 1156, 1162,
1165, 1419, 1480, 1572, 1906, 1914,
1925, 1933, 2334, 2338, 2343, 2347,
2354, 2359, 2363, 2382, 2386, 2391,
2395, 2402, 2407, 2411
- caption2 **51, 69, 244, 387**
- castillan.mld **431**
- castillian.mld **431**
- catalan.mld **431**
- catalog **216**
- ccaption **51, 69, 244, 388**
- \centering . 1306, 1412, 1468, 1515, 1536,
1904, 1912, 1923, 1931
- \cftpagenumbersoff . . 388–392, 399–402,
411–416, 418, 420, 422, 424, 426,
428, 1709, 1710
- \cftpagenumberson . . . 382–386, 394–397,
404–409, 417, 419, 421, 423, 425,
427
- \cftsecfont 1272, 1281
- \cftsecindent 1702, 1704
- \cftsecnumwidth 1703, 1705
- \cftsubsecfont . . . 1273–1275, 1282–1284
- \cftsubsecindent . 1704, 1705, 1707, 2017
- \cftsubsecnumwidth 1706, 1708
- \cftsubsubsecindent 1707, 1708
- ch0.tex **131, 216**
- \ch@pt@c 4696, 4700, 4702, 4708, 4710
- \chapbegin 880
- chapter **238**
- \chapter 1, 13, 20, 35, 42, 61, 76, 94,
123, 138, 157, 174, 185, 190, 195,
203, 249, 281, 308, 309, 318, 354,
376, 396, 428, 443, 458, 471, 527,
538, 597, 667, 669, 735, 739, 774,
789, 803, 815, 840, 864, 956, 1192,
1193, 1196, 1197, 1202, 1203, 1215,
1222, 1229, 1248, 1258, 1292, 1323,
1369, 1405, 1425, 1462, 1529, 1601,
1603, 1607, 1611, 1615, 1619, 1621,
1652, 1659, 1669, 1674, 1678, 1818,
1898, 1917, 1961, 1965, 1971, 1973,
1977, 1984, 1988, 2033, 2110, 2172,
2200, 2259, 2330, 2350, 3263, 3284,
3303, 3324, 3739, 3756, 4334, 4591
- \chapter@mtcqlk . . 2643, 2670, 2678, 2684,
2690, 2699, 2706
- \chapterbegin 810, 1036, 1102, 1161, 4437
- \chapterend 810, 1036, 1102, 1161, 4437
- \chapterfont 2021
- \chaptermark 4720
- \chapternumberline 4737, 4740
- \chardef 318
- checkfiles **28, 31, 57**
- chinese1.mld **432**
- chinese1.mlo **432**
- chinese2.mld **432**
- chinese2.mlo **432**
- \cite 1, 2293, 2294
- CJK **211**
- \cleardoublepage . . 462, 475, 1927, 1998
- \closeout . . . 257, 260, 332, 946–948, 960,
991, 1036, 1067, 1102, 1126, 1161,
1636–1638, 1656, 1670, 1743, 1767,
1803, 1827, 1863, 2249–2251, 2271,
2290, 2340, 2366, 2399, 2425, 2458,
4285, 4288
- \coffee 988, 1674, 2274
- \coffeeFont . . . 574, 821, 834, 1197, 4507
- \coffeeline 3374, 4641
- \colorbox 1769
- \columnbreak 1757, 1774
- \columnrule 2086
- \columnsep 1765
- \columnseprule 1764
- \columnwidth 542, 613, 663, 683,
751, 1228, 1368, 1436, 1515, 1879,
1953, 2019, 2089, 2093, 2505
- \contentsline 1, 274, 300, 311,
880, 1036, 1102, 1161, 1177, 1735,
1796, 1856, 2333, 2392, 2451
- \copy 323, 327
- \count@ 4349
- cri.tex **126, 216**
- croatian.mld **433**
- \cyr 5070, 5072, 5075, 5079, 5081, 5084,
5088, 5090, 5093, 5102, 5104, 5106,
5109, 5111, 5113, 5116, 5118, 5120,
6724, 6727, 6730, 6733, 6737, 6739,
6742, 6746, 6748, 6751, 6880, 6882,
6886, 6888, 6889, 6892, 6894, 7091,
7092, 7094, 7097, 7098, 7100, 7103,
7104, 7106
- \CYRa . . . 6727, 6731, 6734, 6737, 6740, 6743,
6746, 6749, 6752
- \cyra . . . 5070, 5073, 5076, 5079, 5082, 5085,
5088, 5091, 5094, 5102, 5106, 5109,
5113, 5116, 5120, 6724, 6880, 6883,
6886, 6889, 6892, 6895, 7093, 7095,
7099, 7101, 7105, 7107
- \CYRb 6734, 6743, 6752
- \cyrb . . . 5076, 5085, 5094, 5106, 5113, 5120,
6883, 6889, 6895, 7095, 7101, 7107
- \CYRc . . . 6731, 6734, 6740, 6743, 6749, 6752
- \cyrc . . . 5076, 5085, 5094, 5106, 5113, 5120,
6883, 6889, 6895, 7093, 7095, 7099,
7101, 7105, 7107

- \cyrd . . 5070, 5079, 5088, 5102, 5109, 5116,
 6724, 6880, 6886, 6892
 \CYRe 6727, 6737, 6746
 \cyre . . 5070, 5073, 5076, 5079, 5082, 5085,
 5088, 5091, 5094, 5102, 5109, 5116,
 6724, 6882, 6883, 6888, 6889, 6894,
 6895, 7092, 7094, 7098, 7100, 7104,
 7106
 \CYRF 5104, 5111, 5118
 \cyrf 5073, 5082, 5091
 \CYRg 6727, 6737, 6746
 \cyrq . . 5073, 5082, 5091, 5104, 5111, 5118
 \cyrhrdsn . . 5070, 5072, 5075, 5079, 5081,
 5084, 5088, 5090, 5093, 5102, 5109,
 5116
 \CYRi . . 6727, 6730, 6731, 6733, 6734, 6737,
 6739, 6740, 6742, 6743, 6746, 6748,
 6749, 6751, 6752
 \cyri . . 5070, 5072, 5073, 5075, 5076, 5079,
 5081, 5082, 5084, 5085, 5088, 5090,
 5091, 5093, 5094, 5102, 5104, 5106,
 5109, 5111, 5113, 5116, 5118, 5120,
 6724, 6882, 6883, 6888, 6889, 6894,
 6895, 7095, 7101, 7107
 \cyrii . 7091–7094, 7097–7100, 7103–7106
 \CYRishrt 6731, 6740, 6749
 \cyrishrt 7093, 7099, 7105
 \cyrje 6880, 6886, 6892
 \CYRk . . 6730, 6733, 6739, 6742, 6748, 6751
 \cyrk . . 5072, 5075, 5081, 5084, 5090, 5093,
 6882, 6888, 6894, 7092, 7094, 7098,
 7100, 7104, 7106
 \CYRl . . 6727, 6731, 6734, 6737, 6740, 6743,
 6746, 6749, 6752
 \cyr1 . . 5076, 5085, 5094, 5106, 5113, 5120,
 6882, 6883, 6888, 6889, 6894, 6895,
 7092–7095, 7098–7101, 7104–7107
 \cyrn 7091, 7097, 7103
 \CYRn 6727, 6737, 6746
 \cyrn . . 5070, 5073, 5076, 5079, 5082, 5085,
 5088, 5091, 5094, 5102, 5109, 5116,
 6724
 \CYRO 6727, 6737, 6746
 \CYRo . . 6730, 6733, 6739, 6742, 6748, 6751
 \cyro 6724
 \CYRP . . 7092, 7094, 7098, 7100, 7104, 7106
 \CYRp . . 6730, 6733, 6739, 6742, 6748, 6751
 \cyrp . . 5072, 5075, 5081, 5084, 5090, 5093
 \CYRr 6731, 6740, 6749
 \cyrR . . 5070, 5073, 5079, 5082, 5088, 5091,
 5102, 5104, 5109, 5111, 5116, 5118,
 6724, 6880, 6886, 6892, 7092–7094,
 7098–7100, 7104–7106
 \CYRS . . 5070, 5072, 5075, 5079, 5081, 5084,
 5088, 5090, 5093, 5102, 5109, 5116,
 6724, 6730, 6733, 6739, 6742, 6748,
 6751, 6880, 6882, 6886, 6888, 6892,
 6894
 \CYRs . . 6730, 6731, 6733, 6739, 6740, 6742,
 6748, 6749, 6751
 \cyrs . . 5072, 5075, 5081, 5084, 5090, 5093,
 7091, 7093, 7097, 7099, 7103, 7105
 \cyrstsn 7095, 7101, 7107
 \CYRT . . 5106, 5113, 5120, 6883, 6889, 6895
 \CYRt . . 6731, 6734, 6740, 6743, 6749, 6752
 \cyrt . . 5073, 5076, 5082, 5085, 5091, 5094,
 7091, 7093, 7095, 7097, 7099, 7101,
 7103, 7105, 7107
 \cyru . . 5073, 5082, 5091, 5104, 5111, 5118
 \CYRv 6727, 6737, 6746
 \CYRyu 6731, 6740, 6749
 \cyrYu 7093, 7099, 7105
 \CYRZ 7091, 7097, 7103
 \cyrzh . 5070, 5079, 5088, 5102, 5109, 5116,
 6724, 6880, 6886, 6892
 \cz . . . 6626, 6627, 6630, 6631, 6634, 6635
 czech.mld **433**
- D**
- \dal . . . 4844, 4848, 4852, 5461, 5463, 5465,
 5467, 5469, 5471
 danish.mld **434**
 \DeclareOption . 3382, 3386, 3388, 3390,
 3391, 3394, 3411, 3414, 3584
 \decrementmtc 1, 542, 4424
 \decrementptc 1, 1212, 4424
 \decrementstc 1, 1879, 4424
 \df@mtic 613, 683, 751
 \df@mtilf 575, 580, 690, 912
 \df@mtilt 575, 580, 758, 929
 \df@mtitc 575, 580, 620, 895
 \df@pti 1351, 1424, 1499
 \df@ptilf 1313, 1430, 1602
 \df@ptilt 1313, 1505, 1619
 \df@ptitc 1313, 1357, 1585
 \df@sti 1938, 2005, 2072
 \df@stilf 1905, 2011, 2215
 \df@stilt 1905, 2078, 2232
 \df@stitc 1905, 1945, 2198
 \dimen@ 4349
 \divide 1501
 \do@mtilf 575, 580, 685–690, 692, 702, 706
 \do@mtilt 575, 580, 753–758, 760, 770, 774
 \do@mtitc 575, 580, 613, 683, 751
 \do@ptilf 1313, 1424, 1436
 \do@ptilt 1313, 1499, 1515
 \do@ptitc 1313, 1351, 1368

- \do@stilf 1905, 2005, 2019
 \do@stilt 1905, 2072, 2086
 \do@stitc 1905, 1938, 1953
 \documentclass 2, 53, 115, 175, 206,
 207, 467, 714, 715, 917, 1174, 1207,
 1236, 1270, 1303, 1341, 1387, 1434,
 1493, 1591, 1641, 1691, 1888, 1938,
 1995, 2167, 2217, 2251, 2323, 2369
 \dominilof 1, 57, 119,
 235, 667, 731, 880, 885, 916, 1459,
 1526, 1893, 2467, 3871, 3947, 4450
 \dominilof@ 889, 912
 \dominilot 1, 58, 120,
 236, 732, 735, 880, 890, 933, 2327,
 2467, 3879, 3955, 4450
 \dominilot@ 894, 929
 \dominitoc . 1, 10, 234, 597, 730, 880, 899,
 1212, 1245, 1289, 1309, 1355, 1397,
 1596, 1650, 1816, 1892, 1951, 2030,
 2170, 2257, 2467, 3863, 3939, 4450
 \dominitoc@ 880, 884, 895
 \dopartlof . 1, 238, 492, 930, 1570, 1582,
 1606, 2467, 3845, 3921, 4444
 \dopartlof@ 1583, 1602
 \dopartlot . 1, 239, 493, 930, 1570, 1582,
 1623, 2467, 3853, 3929, 4444
 \dopartlot@ 1584, 1619
 \doparttoc 1, 9, 181, 237, 491,
 930, 1188, 1308, 1354, 1570, 1582,
 1589, 1950, 2467, 3837, 3913, 4444
 \doparttoc@ 1582, 1585
 \dosectlof . 1, 38, 489, 931, 2187, 2195,
 2219, 2467, 3824, 3897, 3973, 4456
 \dosectlof@ 2196, 2215
 \dosectlot 1, 38, 490,
 931, 2191, 2195, 2236, 2375, 2467,
 3825, 3905, 3981, 4456
 \dosectlot@ 2197, 2232
 \dosecttoc . 1, 488, 931, 2183, 2195, 2202,
 2228, 2467, 3889, 3965, 4456
 \dosecttoc@ 2195, 2198
 \dotfil 723, 725, 923, 925
 \dotfill 724, 924
 dutch.mld **434**
- E**
- \e@mti .. 575, 613, 683, 751, 895, 912, 929
 \e@pti 1260, 1267, 1351, 1424, 1499, 1585,
 1602, 1619
 \e@sti 1900, 1938, 2005, 2072, 2198, 2215,
 2232
 \em 2308, 2315
 emk **216**
 \emph 2072, 2074, 2142, 2153
- \empty .. 459–461, 465–470, 472–474, 478–
 483, 485–487, 491–496, 1272, 1273,
 1281, 1282, 4474, 4509, 4512, 4513,
 4564–4590, 4619, 4628, 4637, 4647,
 4649, 4651–4657
 \endlist 596, 1329, 1921
 \endmtc@verse 596
 \endmtchideinmainlof 147
 \endmtchideinmainlot ... 155, 2355, 2403
 \endptc@verse 1319
 \endstc@verse 1921
 english.mld **418, 421, 428, 430, 435, 473, 496,**
 498
 english1.mld **436**
 english2.mld **436**
 \enlargethispage 1
 abstract 1, 174
 appendices 1
 filecontents 4751
 mtc@verse .. 586, 639, 657, 709, 730, 777,
 798, 2840, 2855, 2870
 mtchideinmainlof 1, 3194, 4646
 mtchideinmainlot 1, 3227, 4646
 mtchideinmaintoc 1, 3180, 4646
 ptc@verse . 1319, 1382, 1384, 1455, 1461,
 1530, 1536, 2835, 2850, 2865
 samepage 613, 657, 683, 730, 751,
 798, 1351, 1398, 1424, 1472, 1499,
 1547, 1938, 1984, 2005, 2051, 2072,
 2119
 stc@verse . 1911, 1970, 1972, 2036, 2042,
 2103, 2109, 2845, 2860, 2875
 tabular . 613, 683, 751, 1368, 1436, 1515,
 1953, 2019, 2086
 esperant.mld **437**
 esperanto.mld **437**
 estonian.mld **438**
 \eth@doaltchar .. 5390, 5391, 5393, 5394,
 5398, 5399, 5401, 5402, 5406, 5407,
 5409, 5410
 ethiopia.mld **438**
 ethiopian2.mld **439**
 ethopia.mld **439**
 ethopian.mld **439**
 \ExecuteOptions 4270
 \ExecuteOptions{english} 4270
 \ext@subfigure 1438
- F**
- F **267**
 \f@rhdr 4704, 4710, 4712, 4720, 4748
 \f@rtoc 4703, 4708, 4737, 4740, 4743, 4746,
 4748
 \fa ... 4842, 4846, 4850, 5461, 5465, 5469

- \fake@starttoc 497
 - \fakelistoffigures . 241, 495, 497, 1895,
3804, 4002, 4031, 4060, 4424
 - \fakelistoftables . 242, 496, 497, 3804,
4011, 4040, 4069, 4424
 - \faketableofcontents 182, 240, 494,
497, 1189, 1817, 2229, 3804, 3993,
4022, 4051, 4424
 - farsi1.mld 440
 - farsi1.mlo 440
 - farsi2.mld 440
 - farsi2.mlo 440
 - farsi3.mld 441
 - \fbox 1469, 1474, 1477, 1479, 1510
 - \fboxrule 1508
 - \fboxsep 1508
 - \figbox 1509, 1539, 1544, 1549, 1554, 1559,
1564, 1569
 - \figure 1079, 1770, 2369
 - \figure@mtcck . . . 2649, 2655, 2665, 2671,
2685, 2692, 2708
 - filecontents (environment) 4751
 - \finalmem . . 5912, 5914, 5919, 5921, 5926,
5928
 - \finalnun 5911, 5918, 5925
 - finnish.mld 441
 - finnish2.mld 442
 - \firstchapteris 1, 542, 4252, 4523
 - \firstpartis 1, 1212, 4247, 4523
 - \firstsectionis 1, 1879, 4257, 4523
 - \FloatBarrier . 1, 582, 595, 637, 663, 676,
694, 697
 - \flushbottom 220, 480
 - fmintoc.bib 216
 - fmintoc.dtx 214, 216
 - fmintoc.ist 216
 - fmintoc.lan 216
 - fmintoc.pdf 216
 - fmk 216
 - fncychap 50, 72, 242, 385
 - fo1.tex 110, 216
 - fo2.tex 111, 216
 - \footnotesize 2225
 - \FR . . . 5461–5463, 5465–5467, 5469–5471
 - franc.sty 216
 - francais.mld 442
 - frbib.sty 216
 - french.mld 431, 442, 443, 445
 - french1.mld 443
 - french2.mld 106, 444
 - frenchb.mld 445
 - frenchle.mld 445
 - frenchpro.mld 445
 - frnew.sty 216
 - frplain1.bst 216
 - \fup 2016, 2039, 2041, 2108
 - \fussy 317, 568
- ## G
- galician.mld 445
 - gaps 73
 - gaps.tex 73, 216
 - german.mld 421, 446
 - germanb.mld 446
 - germanb2.mld 447
 - \gobbleopt@ 4443, 4444–4462
 - \goodap 1443
 - greek-mono.mld 448
 - greek-polydemo.mld 449
 - greek-polykatha.mld 450
 - greek.mld 447
 - guarani.mld 451
- ## H
- \ha . . . 4842, 4846, 4850, 5461, 5465, 5469
 - hangul-u8.mld 454
 - hangul-u8.mlo 454
 - hangul1.mld 451
 - hangul1.mlo 451
 - hangul2.mld 452
 - hangul2.mlo 452
 - hangul3.mld 453
 - hangul3.mlo 453
 - hangul4.mld 453
 - hanja-u8.mld 455
 - hanja-u8.mlo 455
 - hanja1.mld 454
 - hanja1.mlo 454
 - hanja2.mld 455
 - hanja2.mlo 455
 - \hb@xt@ 846
 - \hebalef 5942, 5945, 5952, 5955, 5962, 5965
 - \hebayin 5939, 5949, 5959
 - \hebbet 5945, 5955, 5965
 - \hebfinalmem 5939, 5942, 5949, 5952, 5959,
5962
 - \hebfinalnun 5938, 5948, 5958
 - \hebkafe 5938, 5948, 5958
 - \heblamed 5945, 5955, 5965
 - \hebmeme 5941, 5944, 5951, 5954, 5961, 5964
 - \hebnun 5939, 5949, 5959
 - \hebresh 5941, 5942, 5944, 5951, 5952, 5954,
5961, 5962, 5964
 - hebrew.mld 456
 - hebrew2.mld 457
 - \hebshin 5941, 5944, 5951, 5954, 5961, 5964
 - \hebtav 5938, 5941, 5944, 5945, 5948, 5951,
5954, 5955, 5958, 5961, 5964, 5965

- `\hebtet` 5945, 5955, 5965
`\hebvav` 5938, 5942, 5945, 5948, 5952, 5955,
 5958, 5962, 5965
`\hebyod` 5939, 5941, 5942, 5944, 5949, 5951,
 5952, 5954, 5959, 5961, 5962, 5964
`hia.tex` **135, 216**
`hide1.tex` **65, 216**
`hide2.tex` **66, 216**
`hints` **49, 54, 239, 393**
`hir.tex` **134, 216**
`\hline` ... 636, 706, 774, 1368, 1436, 1515,
 1953, 2019, 2086
`hop.tex` **125, 216**
`\hrule` ... 555, 1798, 2508–2510, 2517–2519,
 2526–2528
`\Huge` 1729, 1731, 1738
`\huge` 2021
`hungarian.mld` **457**
`hyperref` **50, 59, 69, 241**
- I**
- `icelandic.mld` **458**
`\if@dominilof@used@` 93, 912, 3858, 3935
`\if@dominilot@used@` 93, 929, 3858, 3935
`\if@dominitoc@used@` 93, 880, 895, 3858,
 3935
`\if@dopartlof@used@` 87, 1602, 3832, 3909
`\if@dopartlot@used@` 87, 1619, 3832, 3909
`\if@doparttoc@used@` 87, 1585, 3832, 3909
`\if@dosectlof@used@` ... 99, 2215, 3816,
 3884, 3961
`\if@dosectlot@used@` ... 99, 2232, 3816,
 3884, 3961
`\if@dosecttoc@used@` 99, 2198, 3884, 3961
`\if@filesw` 121, 506
`\if@firstchapteris@used@` 105, 542, 4250
`\if@firstpartis@used@` . 105, 1212, 4245
`\if@firstsectionis@used@` .. 105, 4255
`\if@longextensions` 600
`\if@longextensions@` 332, 600, 670, 738,
 960, 991, 1067, 1126, 1331, 1404,
 1478, 1647, 1661, 1758, 1818, 1923,
 1989, 2056, 2262, 2290, 2357, 2416,
 3391, 4074, 4281, 4591
`\if@mainmatter` 4716, 4734
`\if@minilof@used@` . 93, 668, 3858, 3935,
 4016
`\if@minilot@used@` . 93, 736, 3858, 3935,
 4016
`\if@minitoc@used@` . 93, 598, 3858, 3935,
 4016
`\if@mtc@abstract@loaded@` 3586
`\if@mtc@alnumsecLoaded@` . 39, 164, 4152
`\if@mtc@alnumsecLoaded@a@` 39, 164, 4152
`\if@mtc@captcontLoaded@` . 39, 166, 4160
`\if@mtc@captcontLoaded@a@` 39, 166, 4160
`\if@mtc@captionIILoaded@` 39, 170, 4176
`\if@mtc@captionIILoaded@a@` . . 39, 170,
 4176
`\if@mtc@captionLoaded@` . . 39, 168, 4168
`\if@mtc@captionLoaded@a@` 39, 168, 4168
`\if@mtc@ccaptionLoaded@` . 39, 172, 4184
`\if@mtc@ccaptionLoaded@a@` 39, 172, 4184
`\if@mtc@chapter@def@` . . . 77, 3858, 3935,
 4016, 4074
`\if@mtc@chapter@undef@` . 83, 1877, 2466
`\if@mtc@checkfiles` 227, 3386
`\if@mtc@empty@minilof@` . . 66, 675, 4220
`\if@mtc@empty@minilot@` . . 71, 743, 4225
`\if@mtc@empty@minitoc@` . . 66, 605, 4215
`\if@mtc@empty@partlof@` 66, 4205
`\if@mtc@empty@partlot@` 66, 4210
`\if@mtc@empty@parttoc@` 66, 4200
`\if@mtc@empty@sectlof@` 66, 4235
`\if@mtc@empty@sectlot@` 66, 4240
`\if@mtc@empty@secttoc@` 72, 4230
`\if@mtc@FE` . 227, 605, 613, 675, 683, 743,
 751, 1338, 1351, 1411, 1424, 1486,
 1499, 1930, 1938, 1996, 2005, 2063,
 2072
`\if@mtc@fncychapLoaded@` . 39, 156, 4120
`\if@mtc@fncychapLoaded@a@` 39, 156, 4120
`\if@mtc@Himlof@` 3194
`\if@mtc@Himlot@` 3227
`\if@mtc@Himtoc@` 3179, 3180
`\if@mtc@hints@` 35, 895, 912,
 929, 1585, 1602, 1619, 2198, 2215,
 2232, 2486, 3584, 3791, 4327
`\if@mtc@hints@giben@` 35
`\if@mtc@hints@given@` . . 37, 3594, 3599,
 3605, 3611, 3629, 3635, 3640, 3699,
 3706, 3716, 3723, 3734, 3739, 3756,
 3773, 3821, 3832, 3858, 3884, 3909,
 3935, 3961, 3987, 4016, 4045, 4104,
 4112, 4120, 4128, 4136, 4144, 4152,
 4160, 4168, 4176, 4184, 4192, 4260
`\if@mtc@hints@given@true` 4074
`\if@mtc@hints@w@` . 35, 3739, 3756, 3773,
 3821
`\if@mtc@hyper@used@` . . 135, 3739, 3756,
 3773
`\if@mtc@insellang@` 3340, 3353
`\if@mtc@LI` 227
`\if@mtc@listfiles@` 3411, 4326
`\if@mtc@lof@used@` . . 3586, 3791, 3795,
 3804, 3998, 4027, 4056
`\if@mtc@lot@used@` . . 3586, 3791, 3795,
 3804, 4007, 4036, 4065

- \if@mtc@mcaptionLoaded@ . 39, 174, 4192
 - \if@mtc@mcaptionLoaded@a@ 39, 174, 4192
 - \if@mtc@memoirLoaded@ 39, 176, 380
 - \if@mtc@memoirnew@ 39, 176
 - \if@mtc@notoccite@ . . . 130, 3390, 4278
 - \if@mtc@part@def@ 75, 3832, 3909, 3987, 4074
 - \if@mtc@part@undef@ 81
 - \if@mtc@placeinsLoaded@ 39, 3821, 4272
 - \if@mtc@quotchapLoaded@ . 39, 158, 4128
 - \if@mtc@quotchapLoaded@a@ 39, 158, 4128
 - \if@mtc@romannumLoaded@ . 39, 160, 4136
 - \if@mtc@romannumLoaded@a@ 39, 160, 4136
 - \if@mtc@sect@floats@ . . . 38, 3816, 3821
 - \if@mtc@section@def@ . . . 79, 1877, 2465, 3816, 3884, 3961, 4045, 4074
 - \if@mtc@section@undef@ 85
 - \if@mtc@sectstyLoaded@ . . 39, 152, 4104
 - \if@mtc@sectstyLoaded@a@ 39, 152, 4104
 - \if@mtc@setdepth@ . . . 3141, 3143, 3160
 - \if@mtc@setfeature@ . 3088, 3109, 3127
 - \if@mtc@setfont@ 2595, 2597, 2615, 2633, 2711
 - \if@mtc@setformat@ 2880, 2916
 - \if@mtc@setpagenumbers@ . . 2930, 2970
 - \if@mtc@setrules@ 2999, 3020
 - \if@mtc@settitle@ . . . 2762, 2764, 2782
 - \if@mtc@settitlefont@ 2726, 2728, 2746
 - \if@mtc@sfheadersLoaded@ 39, 162, 4144
 - \if@mtc@sfheadersLoaded@a@ . . 39, 162, 4144
 - \if@mtc@spn@ok@ 2930, 2970
 - \if@mtc@sru@ok@ 2999, 3040
 - \if@mtc@ss@insection@ 3394, 3640, 4272
 - \if@mtc@toc@used@ . . . 3586, 3791, 3795, 3804, 3989, 4018, 4047
 - \if@mtc@varsectsLoaded@ . 39, 154, 4112
 - \if@mtc@varsectsLoaded@a@ 39, 154, 4112
 - \if@mtcfixglossary@ 3278
 - \if@mtcfixindex@ 3318
 - \if@mtcpatchmemoir@ 176, 195
 - \if@partlof@used@ 87, 1404, 3832, 3909, 3987
 - \if@partlot@used@ 87, 1478, 3832, 3909, 3987
 - \if@parttoc@used@ 87, 1331, 3832, 3909, 3987
 - \if@sectlof@used@ 99, 1989, 3816, 3884, 3961, 4045
 - \if@sectlot@used@ 99, 2056, 3816, 3884, 3961, 4045
 - \if@secttoc@used@ 99, 1923, 3884, 3961, 4045
 - \if@setrules@false 3040
 - \if@twocolumn 1270, 1284, 1298
 - \if@twoside 1999
 - \ifanappendix 4735
 - \ifartopt 4721
 - \ifdim 2153
 - \IfFileExists 196, 244, 2467, 3340, 3353, 4281
 - \ifheadnameref 4748
 - \ifktightmtc 1, 133, 586, 1319, 1911, 4591
 - ifmtarg **245**
 - \ifmtc@lof@used@ 3987, 4016, 4045
 - \ifmtc@lot@used@ 3987, 4016, 4045
 - \ifmtc@toc@used@ 3987, 4016, 4045
 - \ifmtcoffwarn@ 4597, 4614, 4660
 - \ifmtcsecondpart . . 108, 109, 115, 1182, 4591, 5558
 - \ifodd 2000
 - \ifscan@allowed 720, 920
 - \iftightmtc 1, 133, 586, 1319, 1911, 3382, 4591
 - \ifundottedmtc . . 367, 851, 854, 857, 859, 869, 3388, 4591
 - \ignorespaces 805
 - imk **216**
 - \in 7052, 7056, 7060
 - \incrementmtc 1, 542, 4424
 - \incrementptc 1, 1212, 4424
 - \incrementstc 1, 1879, 4424
 - indon.mld **458**
 - indonesian.mld **458**
 - \input 332, 4341
 - \InputIfFileExists 3342, 3355
 - insection **29, 50**
 - \insertchapterspace 4725
 - INSTALL **214**
 - \interlinepenalty . . 373, 826, 839, 1727, 1737, 2156
 - interlingua.mld **459**
 - irish.mld **459**
 - italian.mld **460**
 - italian2.mld **460**
 - \itemindent 586, 642, 712, 780, 1319, 1380, 1453, 1528, 1913, 1914, 1968, 2034, 2101
 - \itemsep 586, 1319, 1912
- J**
- japanese.mld **461**
 - japanese.mlo **461**
 - japanese2.mld **461**
 - japanese2.mlo **461**
 - japanese3.mld **462**
 - japanese3.mlo **462**
 - japanese4.mld **462**

- japanese4.mlo **462**
japanese5.mld **463**
japanese5.mlo **463**
\jeem .. 4844, 4848, 4852, 5461, 5463, 5465,
5467, 5469, 5471
\jobname 332, 2467, 4281, 4341
jura **50, 71**
- K**
- k-loose **39**
k-tight **39, 59, 241, 263, 292**
\kaf ... 4843, 4847, 4851, 5462, 5466, 5470,
5911, 5918, 5925
kanlel.sty **463**
kannada.mld **463**
\kernafterminilof 17, 731, 4361
\kernafterminilot 17, 799, 4361
\kernafterminitoc 17, 658, 4361
\kernafterpartlof 17, 1472, 4361
\kernafterpartlot 17, 1547, 4361
\kernafterparttoc 17, 1398, 4361
\kernaftersectlof .. 17, 2051, 2119, 4361
\kernaftersectlot 17, 4361
\kernaftersecttoc 17, 1984, 4361
khalkha.mld **464**
\kiftightmtc 3382
\ktightmtcfalse 1, 134, 3385, 4593
\ktightmtctrue 1, 3384
- L**
- \l@chapter .. 311, 803, 1389, 1390, 2497,
2499, 4517
\l@chapter@SVPN 1389, 1390
\l@coffee 3374, 4643
\l@listof 2496, 4591
\l@mti .. 575, 580, 613, 683, 751, 895, 912,
929
\l@paragraph 2503, 4521
\l@part 274, 2498, 4516
\l@pchapter 1552
\l@psect 1552
\l@pti 1260, 1270, 1313, 1351, 1424, 1499,
1585, 1602, 1619
\l@schapter 2123
\l@section 300, 2496, 2500, 4518
\l@starchapter 2496, 4516
\l@starparagraph 2496, 4516
\l@starpart 2496, 4516
\l@starsection 2496, 4516
\l@starsubparagraph 2496, 4516
\l@starsubsection 2496, 4516
\l@starsubsubsection 2496, 4516
\l@sti 1900, 1905, 1938, 2005, 2072, 2198,
2215, 2232
\l@subfigure 1437
\l@subparagraph 2504, 4522
\l@subsection 2501, 4519
\l@subsubsection 2502, 4520
\l@xchapter 311, 803, 4463
\l@xpart 274, 1552, 4463
\l@xsect 300, 2123, 4463
\l@xsection 300, 2123
\label .. 1405, 1409, 1428, 1478, 1481, 1538,
1543, 1548, 1553, 1558, 1563, 1568,
1573, 2034, 2115, 2259
\labelitemi 1781, 1786
\labelsep 643, 713, 781, 1381, 1454, 1529,
1969, 2035, 2102
\labelwidth ... 642, 712, 780, 1380, 1453,
1528, 1968, 2034, 2101
\lam ... 4842–4844, 4846–4848, 4850–4852,
5462, 5463, 5466, 5467, 5470, 5471
\lamed 5916, 5923, 5930
lamed.eps **216**
lamed.pdf **216**
lamed.tex **216**
\LangSig 2255, 2291
\ LARGE 1258
\ Large 1244, 1285, 2023
\ large .. 573, 1755, 1780, 1785, 1789, 1792,
1899
latin.mld **464**
latin2.mld **465**
latvian.mld **465, 466**
\ldots 1959, 1963, 1967, 2238, 2247
\leftmargin 594, 595, 640, 710, 778, 1327,
1328, 1378, 1451, 1526, 1919, 1920,
1966, 2032, 2099
letton.mld **466**
\linewidth .. 1753, 1762, 1770, 2260, 2261
lipsum **199**
\lipsum 17,
19, 24, 26, 39, 41, 46, 48, 251, 253,
255, 257, 260, 262, 264, 266, 268,
270, 272, 274, 276, 278, 280, 282,
284, 286, 288, 290, 292, 294, 296,
298, 300, 302, 304, 306, 308, 310,
312, 314, 316, 324, 326, 341, 343,
345, 353, 356, 359, 362, 365, 368,
371, 378, 380, 382, 384, 386, 393,
395, 404, 406, 408, 410, 417, 419,
421, 423, 431, 433, 435, 448, 450,
452, 458, 506, 508, 510, 512, 515,
517, 519, 521, 523, 525, 527, 529,
531, 533, 535, 537, 539, 541, 543,
545, 547, 549, 551, 553, 555, 557,
559, 561, 563, 565, 567, 573, 575,
584, 586, 588, 597, 602, 604, 606,

- 608, 610, 612, 622, 624, 626, 628,
630, 638, 640, 650, 652, 654, 656,
666, 668, 679, 683, 685, 687, 696,
702, 704, 706, 708, 710, 1217, 1219,
1224, 1227, 1232, 1250, 1252, 1254,
1256, 1260, 1262, 1264, 1266, 1295,
1297, 1299, 1325, 1327, 1329, 1331,
1333, 1335, 1337, 1371, 1373, 1375,
1377, 1379, 1381, 1383, 1655, 1657,
1662, 1664, 1666, 1672, 1674, 1676,
1682, 1684, 1686, 1902, 1910, 1921,
1929, 2332, 2336, 2340, 2345, 2349,
2352, 2357, 2361, 2365, 2380, 2384,
2388, 2393, 2397, 2400, 2405, 2409,
2413
\list 586, 1319, 1912
listfiles **49**
\listoffigures . 1, 59, 121, 241, 291, 735,
937, 1461, 1527, 1630, 3795, 4001,
4030, 4059
\listoftables . . 1, 60, 122, 242, 291, 737,
939, 1399, 2329, 2377, 3795, 4010,
4039, 4068
\listparindent . 586, 643, 713, 781, 1319,
1381, 1454, 1529, 1914, 1969, 2035,
2102
lithuanian.mld **466**
livre.tex **127, 216**
\localgreek 5685, 5688, 5692, 5696, 5699,
5703, 5707, 5710, 5714, 5725, 5728,
5732, 5736, 5739, 5743, 5747, 5750,
5754, 5765, 5768, 5772, 5776, 5779,
5783, 5787, 5790, 5794
\locallatvian . . 6186–6188, 6190–6192,
6194–6196
\localpolish . . . 6518–6520, 6522–6524,
6526–6528
\localrussian . . . 6644, 6647, 6651, 6655,
6658, 6662, 6666, 6669, 6673, 6684,
6687, 6691, 6695, 6698, 6702, 6706,
6709, 6713
\localspanish . . . 6971–6973, 6975–6977,
6979–6981
lofdepth **31, 259**
loose **39**
lotdepth **31, 259, 272**
lowersorbian.mld **467**
lsorbian.mld **467**
- M**
- M **264**
\M@getttitle 4748
magyar.mld **457, 468**
magyar2.mld **468**
magyar3.mld **469**
\maketitle 2028
\MakeUppercase 612, 682, 750, 1345, 1418,
1493, 1938, 2003, 2070
malay.mld **469**
malayalam-keli.mld **469**
malayalam-omega.mld **470**
malayalam-omega.mlo **470**
malayalam-rachana.mld **470**
malayalam-rachana2.mld **471**
manju.mld **472**
mcaption **51, 69, 244, 388**
\mdseries 475, 563, 574, 1234, 1247, 1249,
1321, 1367, 1890
\meem . . 4843, 4844, 4847, 4848, 4851, 4852,
5461, 5465, 5469
\mem . . . 5913, 5915, 5920, 5922, 5927, 5929
mem.tex **108, 216**
mem1.tex **109, 216**
memoir **50, 62, 69, 237, 245, 254**
\MessageBreak 4676
meyalu.mld **472**
mini-art.tex **90, 216**
\minilof . 1, 62, 77, 98, 124, 139, 158, 321,
445, 667, 742, 807, 844, 869, 1463,
1466, 1530, 1533, 1900, 1918, 3870,
3948, 4029, 4450
\minilof@ 667, 668, 849
\minilof@mtcck 2637, 2676
minilofdepth **259, 269**
\minilot . 1, 63, 78, 99, 125, 140, 159, 323,
446, 735, 744, 809, 846, 871, 2331,
2351, 3878, 3956, 4038, 4450
\minilot@ 735, 736, 849
\minilot@mtcck 2640, 2682
minilotdepth **272**
\minitoc 1, 14, 21,
36, 43, 250, 319, 376, 377, 400, 429,
444, 597, 740, 805, 842, 867, 1216,
1223, 1230, 1249, 1259, 1293, 1324,
1370, 1426, 1604, 1608, 1612, 1616,
1623, 1653, 1660, 1670, 1680, 1772,
1819, 1899, 1918, 1974, 1978, 1985,
1989, 2035, 2111, 2174, 2201, 2263,
3862, 3940, 4020, 4450
minitoc-ex.tex **84, 216**
minitoc.bib **216**
minitoc.bug **49, 54, 216**
minitoc.dtx **214, 216**
minitoc.ins **41, 214, 216**
minitoc.ist **216**
minitoc.l **214, 216**
minitoc.lan **216**
minitoc.pdf **216**

- minitoc.pre **216**
- minitoc.sty **214**
- minitoc.sum **216**
- \minitoc@ 597, 598, 849
- \minitoc@mtcck 2634, 2668
- minitocdepth **31, 261, 265**
- minutes **262**
- mlf **267**
- \MLF@ 1100, 1102
- \MLF@contentsline 1063, 1102
- \MLF@explist 1061, 1088
- \MLF@line 1099, 1100
- \MLF@list 1058, 1061, 1088, 1102
- \MLF@lof 1061, 1089, 1102
- \MLF@loop 1058, 1089
- \MLF@next 885, 1058, 1088
- \mlf@pgno 436, 437
- \MLF@read 1089, 1098, 1102
- \mlf@rule 542, 683, 732, 2515, 2518, 2521
- \mlf@setform 720, 2855
- \MLF@test 1100, 1102
- \mlf@dotsep 2825, 2858
- \mlffont 1, 563, 683, 4486
- \mlflofrmarg 2816
- \mlfname 1067
- \mlfpagenumbers 417, 430, 436, 449, 4546
- \mlfpnumwidth 2807, 2856
- \mlfrule 2505, 4528
- \mlfSfont 1, 563, 4486
- \mlftitle 1, 682, 692, 702, 706, 4751, 4769, 4790, 4811, 4829, 4847, 4869, 4898, 4915, 4932, 4950, 4966, 4982, 4998, 5014, 5031, 5052, 5080, 5110, 5131, 5147, 5182, 5220, 5239, 5256, 5273, 5290, 5313, 5339, 5356, 5377, 5397, 5428, 5466, 5483, 5500, 5521, 5544, 5577, 5606, 5621, 5638, 5653, 5671, 5699, 5703, 5739, 5779, 5809, 5920, 5950, 5982, 6007, 6024, 6042, 6060, 6134, 6135, 6155, 6172, 6191, 6213, 6230, 6254, 6271, 6288, 6309, 6332, 6349, 6373, 6402, 6417, 6436, 6452, 6467, 6483, 6504, 6523, 6541, 6562, 6579, 6597, 6630, 6658, 6698, 6738, 6763, 6831, 6848, 6865, 6887, 6907, 6924, 6945, 6960, 6976, 6997, 7014, 7031, 7055, 7092, 7120, 7147, 7168, 7183, 7198, 7213
- \mlftocrmarg 2820, 2857
- mlt **270**
- \MLT@ 1159, 1161
- \MLT@contentsline 1122, 1161
- \MLT@explist 1120, 1147
- \MLT@line 1158, 1159
- \MLT@list 1117, 1120, 1147, 1161
- \MLT@loop 1117, 1148
- \MLT@lot 1120, 1148, 1161
- \MLT@next 890, 1117, 1147
- \mlt@pgno 442, 443
- \MLT@read 1148, 1157, 1161
- \mlt@rule 542, 751, 800, 2524, 2527, 2530
- \mlt@setform 788, 2870
- \MLT@test 1159, 1161
- \mltdotsep 2825
- \mltfont 1, 563, 751, 4486
- \mlttitle 4791
- \mltlofrmarg 2816
- \mltname 1126
- \mltpagenumbers 423, 430, 436, 442, 449, 4546
- \mltpnumwidth 2807
- \mltrule 2505, 4528
- \mltSfont 1, 563, 4486
- \mltttitle 1, 750, 760, 770, 774, 4751, 4770, 4812, 4830, 4848, 4870, 4899, 4916, 4933, 4951, 4967, 4983, 4999, 5015, 5032, 5053, 5083, 5112, 5132, 5148, 5183, 5221, 5240, 5257, 5274, 5291, 5314, 5340, 5357, 5378, 5400, 5430, 5467, 5484, 5501, 5522, 5545, 5578, 5607, 5622, 5639, 5654, 5672, 5743, 5783, 5810, 5922, 5953, 5983, 6008, 6025, 6043, 6061, 6156, 6173, 6192, 6214, 6231, 6255, 6272, 6289, 6310, 6333, 6350, 6374, 6403, 6418, 6437, 6453, 6468, 6484, 6505, 6524, 6542, 6563, 6580, 6598, 6615, 6616, 6631, 6662, 6702, 6741, 6764, 6832, 6849, 6866, 6889, 6908, 6925, 6946, 6961, 6977, 6998, 7015, 7032, 7056, 7094, 7121, 7148, 7169, 7184, 7199, 7214
- \mlttocrmarg 2823
- \mm ... 6304–6306, 6308–6310, 6312–6314, 6327–6329, 6331–6333, 6335–6337, 6344–6346, 6348–6350, 6352–6354
- \mnr ... 5126–5128, 5130–5132, 5134–5136, 5142–5144, 5146–5148, 5150–5152, 6368–6370, 6372–6374, 6376–6378, 7178–7180, 7182–7184, 7186–7188, 7193–7195, 7197–7199, 7201–7203
- mongol.mld **472**
- mtc **261, 264**
- \MTC@ 1034, 1038
- \mtc@While 213, 215, 220
- \mtc@addtomaf 4281
- \mtc@arg@dotinterval 2799
- \mtc@arg@numwidth 2805
- \mtc@arg@pagenumwidth 2803

- \mtc@arg@tocrightmargin 2801
- \mtc@BBR . . . 318, 639, 645, 709, 715, 777,
783, 1377, 1382, 1450, 1455, 1525,
1530, 1965, 1970, 2031, 2036, 2098,
2103
- \mtc@Body 227
- \mtc@bottom@rule . . . 657, 663, 730, 798
- \mtc@ck 4614
- \mtc@CkFile . . . 227, 605, 675, 743, 1338,
1411, 1486, 1930, 1996, 2063
- \mtc@CkStr 227, 613, 683, 751, 1351, 1424,
1499, 1938, 2005, 2072
- \MTC@contentsline 956, 1036
- \mtc@dot 1177
- \mtc@EndWhile 202, 227
- \MTC@explist 953, 1021
- \mtc@featparam@abbrev 3090, 3109, 3127
- \mtc@featparam@third 3109, 3127
- \mtc@featureparam@after 3081
- \mtc@featureparam@before 3079
- \mtc@featureparam@pagestyle . . . 3083
- \mtc@fparam@abbrev . . 2882, 2901, 2916
- \mtc@glofix@level 3260, 3279, 3288–3290,
3296
- \mtc@hints@@chapter 3756, 4334
- \mtc@hints@@part 3739, 4331
- \mtc@hints@@schapter 3756, 4334
- \mtc@hints@@sect 3773, 4337
- \mtc@hints@@spart 3739, 4331
- \mtc@hints@@sssect 3773, 4337
- \mtc@hints@begindoc . 3586, 3814, 4327
- \mtc@hints@chapter . . 3739, 3756, 4334
- \mtc@hints@checklongext 4074
- \mtc@hints@enddoc 3586, 3815, 4260, 4327
- \mtc@hints@part 3739, 4331
- \mtc@hints@section . . 3739, 3773, 4337
- \mtc@hook@beforeinputfile . . 130, 647,
720, 788, 1384, 1461, 1536, 1972,
2042, 2109
- \mtc@hstrut . 318, 1447, 1522, 2027, 2094
- \mtc@hstrutbox 318
- \mtc@ifmtarg . . . 202, 237, 274, 300, 311
- \mtc@ixfix@level 3300, 3319, 3328–3330,
3336
- \mtc@level@abbrev . . . 2615, 2632, 2711
- \MTC@line 1032, 1034
- \MTC@list 949, 953, 1021, 1036
- \MTC@loop 949, 1022
- \mtc@ltypetable@minilof 3073
- \mtc@ltypetable@minilot 3074
- \mtc@ltypetable@minitoc 3072
- \mtc@ltypetable@partlof 3070
- \mtc@ltypetable@partlot 3071
- \mtc@ltypetable@parttoc 3069
- \mtc@ltypetable@sectlof 3076
- \mtc@ltypetable@sectlot 3077
- \mtc@ltypetable@secttoc 3075
- \mtc@maf 4281, 4326
- \mtc@maf@long 4281
- \mtc@maf@short 4281
- \mtc@markboth . . 517, 612, 682, 750, 1345,
1418, 1493, 1938, 2003, 2070
- \mtc@mta@abbrev 2597, 2632, 2711
- \mtc@mtade@abbrev 3143, 3160
- \mtc@mtatf@abbrev 2728, 2746
- \mtc@mtati@abbrev 2764, 2782
- \mtc@mtf@abbrev 2882, 2916
- \mtc@mtfeat@abbrev 3090, 3127
- \mtc@mttpn@abbrev 2951, 2970
- \mtc@mttru@abbrev 3020, 3040
- \MTC@next 880, 949, 1021
- \mtc@noX 2987, 2989, 3040
- \mtc@onebackchap 262
- \mtc@onebackchapter 261
- \mtc@onebackpart 261
- \mtc@onebacksect 263, 291
- \mtc@onebacksection 261
- \mtc@param 1177
- \mtc@pgno 430, 431, 647, 720, 788
- \mtc@pns@abbrev 2951, 2970
- \mtc@pns@abbrevX 2986
- \MTC@read 1022, 1032, 1036
- \mtc@restorelofdepth 3214, 4652
- \mtc@restorelotdepth . . 3247, 3259, 4652
- \mtc@restoretocdepth . 3180, 3210, 3226,
3243, 4652
- \mtc@Rline 227
- \mtc@rule 542, 613, 663, 2506, 2509, 2512
- \mtc@rusw@abbrev 3020, 3040
- \mtc@rusw@abbrevX 3056
- \mtc@savelofdepth . . . 3213, 3220, 4652
- \mtc@savelotdepth . . . 3246, 3253, 4652
- \mtc@savetocdepth 3180, 3204, 3237, 4652
- \mtc@schapter 810
- \mtc@sectlevel@chapter 2584
- \mtc@sectlevel@figure 2591
- \mtc@sectlevel@paragraph 2588
- \mtc@sectlevel@section 2585
- \mtc@sectlevel@subfigure 2593
- \mtc@sectlevel@subparagraph . . . 2589
- \mtc@sectlevel@subsection 2586
- \mtc@sectlevel@subsubsection . . . 2587
- \mtc@sectlevel@subtable 2594
- \mtc@sectlevel@table 2592
- \mtc@setform 647, 2840
- \mtc@string 318, 1036, 1102, 1161, 1177,
1197, 1734, 1795, 1855, 2332, 2391,
2450, 4463

- \mtc@strut 318, 647, 720, 788, 1384, 1461, 1536, 1972, 2042, 2109
- \mtc@strutbox 318
- \mtc@sv@fakelistoffigures 3804
- \mtc@sv@fakelistoftables 3804
- \mtc@sv@faketableofcontents ... 3804
- \mtc@sv@listoffigures 3795
- \mtc@sv@listoftables 3795
- \mtc@sv@lofdepth 3194
- \mtc@sv@lotdepth 3227
- \mtc@sv@tableofcontents 3795
- \mtc@sv@tocdepth 3180, 3194, 3227
- \mtc@svchapter 309
- \mtc@svf@tocdepth 3194
- \mtc@svpart 264, 1557
- \mtc@svsection 297
- \mtc@svspart 264
- \mtc@svss 297
- \mtc@svt@tocdepth 3227
- \MTC@test 1034, 1036
- \mtc@tmp@name 2632, 2711
- \mtc@tmpde@name 3160
- \mtc@tmpfeat@name 3127
- \mtc@tmpfm@name 2916
- \mtc@tmppn@name 2970, 3040
- \mtc@tmptf@name 2746
- \mtc@tmpti@name 2782
- \MTC@toc 953, 1022, 1036
- \mtc@toks 318, 972, 1079, 1138, 1177, 1197, 1673, 1770, 1830, 2274, 2369, 2428, 2711, 2746, 2782, 3160
- \mtc@typetable@minilof 2536
- \mtc@typetable@minilot 2537
- \mtc@typetable@minitoc 2535
- \mtc@typetable@partlof 2533
- \mtc@typetable@partlot 2534
- \mtc@typetable@parttoc 2532
- \mtc@typetable@sectlof 2539
- \mtc@typetable@sectlot 2540
- \mtc@typetable@secttoc 2538
- \mtc@typetitle@minilof 2545
- \mtc@typetitle@minilot 2546
- \mtc@typetitle@minitoc 2544
- \mtc@typetitle@partlof 2542
- \mtc@typetitle@partlot 2543
- \mtc@typetitle@parttoc 2541
- \mtc@typetitle@sectlof 2548
- \mtc@typetitle@sectlot 2549
- \mtc@typetitle@secttoc 2547
- \mtc@v 318, 632, 636, 702, 706, 770, 774, 1368, 1436, 1515, 1953, 2019, 2086
- \mtc@verse 586
- mtc@verse (environment) 586, 639, 657, 709, 730, 777, 798, 2840, 2855, 2870
- \mtc@While 202, 227
- \mtc@WhileBody 202
- \mtc@WhileCondition 202
- \mtc@WhileNext 202
- \mtc@WhilePreCondition 202
- \MTC@WriteCoffeeline .. 988, 1197, 1684, 2274
- \MTC@WriteContentsLine ... 1644, 2257, 2352, 2411
- \MTC@WriteContentsline 973, 1005, 1079, 1138, 1177, 1674, 1698, 1770, 1830, 2274, 2305, 2369, 2428
- \mtc@xifmtarg 202
- \mtc@YN@F 2561
- \mtc@YN@f 2560
- \mtc@YN@FALSE 2557
- \mtc@YN@false 2556
- \mtc@YN@FAUX 2559
- \mtc@YN@faux 2558
- \mtc@YN@N 2555
- \mtc@YN@n 2554
- \mtc@YN@NO 2553
- \mtc@YN@no 2552
- \mtc@YN@NON 2562
- \mtc@YN@non 2563
- \mtc@YN@O 2581
- \mtc@YN@o 2582
- \mtc@YN@OFF 2551
- \mtc@YN@off 2550
- \mtc@YN@ON 2566
- \mtc@YN@on 2565
- \mtc@YN@OUI 2579
- \mtc@YN@oui 2580
- \mtc@YN@T 2574
- \mtc@YN@t 2573
- \mtc@YN@TRUE 2572
- \mtc@YN@true 2571
- \mtc@YN@TRUEV 2578
- \mtc@YN@v 2577
- \mtc@YN@VRAI 2576
- \mtc@YN@vrai 2575
- \mtc@YN@Y 2570
- \mtc@YN@y 2569
- \mtc@YN@YES 2568
- \mtc@YN@yes 2567
- \mtc@zrule . 318, 639, 660, 709, 733, 777, 801, 1377, 1398, 1450, 1472, 1525, 1547, 1965, 1984, 2031, 2051, 2098, 2119
- \mtcaddchapter 1, 12, 174, 311, 734, 736, 738, 777, 778, 781, 782, 787, 788, 836, 1400, 1600,

- 1602, 1620, 1622, 1632, 1962, 1966,
1972, 2173, 4614
- \mtcaddpart . . . 1, 274, 1897, 1954, 4614
- \mtcaddsection 1, 174, 300, 600, 936, 938,
940, 979–982, 987, 988, 1068–1071,
1076, 1077, 1122, 4614
- \mtcArabTok 4823–4826, 4828–4830,
4832–4834
- \mtcdotsep 2825, 2843
- \mtcEnglishIIpart 5325, 5334–5336
- \mtcfixglossary 1, 3264, 3269, 3271, 3278,
4658
- \mtcfixindex . . . 1, 833, 1119, 3304, 3309,
3311, 3318, 4659
- \mtcfont 1, 215, 563, 613, 4486
- \mtcFrenchIIpart . 5556, 5568, 5571, 5574
- \mtcgapafterheads . . 15, 188, 193, 1270,
1284, 1298, 4359
- \mtcgapbeforeheads . 15, 187, 192, 1270,
1284, 1298, 4359
- \mtchideinmainlof 142
- mtchideinmainlof (environment) . 1, 3194,
4646
- \mtchideinmainlot 150, 2342, 2390
- mtchideinmainlot (environment) . 1, 3227,
4646
- mtchideinmaintoc (environment) . 1, 3180,
4646
- \mtcindent 9, 214, 542, 586, 639, 709, 777,
1648, 1713, 2253, 4349
- \mtcloadmlo 3353, 4527, 5197, 5208, 5446,
5455, 5826, 5839, 5851, 5863, 5872,
5884, 5896, 5905, 6074, 6086, 6098,
6110, 6123, 6319, 6777, 6787, 6795,
6803, 6811, 6819, 7043
- \mtclofrmarg 2816
- mtcmess.sty 214
- \mtcname 960, 991
- mtcoff 52, 55
- mtcoff.sty 214
- \mtcoffwarn@false 4597
- \mtcoffwarn@true 4363, 4369, 4375, 4385,
4391, 4397, 4406, 4412, 4418, 4614,
4623, 4632, 4660, 4665, 4670
- \mtcPackage 198
- \mtcPackageError 2602, 2619, 2719, 2733,
2755, 2769, 2791, 2888, 2904, 2923,
2958, 2977, 3027, 3046, 3096, 3113,
3135, 3148, 3163, 3174, 3183, 3189,
3198, 3206, 3216, 3222, 3231, 3239,
3249, 3255, 3267, 3292, 3307, 3332,
3345, 3358, 3367, 3397, 4676
- \mtcPackageInfo . . . 2, 135, 178, 184, 186,
188, 268, 281, 283, 289, 308, 332,
343, 345, 607, 677, 745, 898, 915,
932, 961, 965, 994, 998, 1028, 1068,
1072, 1095, 1127, 1131, 1154, 1340,
1413, 1488, 1588, 1605, 1622, 1648,
1652, 1662, 1666, 1725, 1759, 1763,
1786, 1819, 1823, 1846, 1932, 1998,
2065, 2201, 2218, 2235, 2263, 2267,
2294, 2298, 2325, 2358, 2362, 2384,
2417, 2421, 2443, 2467, 2714, 2749,
2785, 2918, 2990, 2993, 3060, 3063,
3130, 3169, 3343, 3356, 3591, 3601,
3607, 3613, 3619, 3625, 3631, 3701,
3718, 3727, 4265, 4282, 4676, 4693
- \mtcPackageWarning 544, 1024, 1091, 1150,
1217, 1721, 1782, 1842, 1880, 2321,
2380, 2439, 4364, 4370, 4376, 4386,
4392, 4398, 4407, 4413, 4419, 4615,
4624, 4633, 4661, 4666, 4671, 4676
- \mtcPackageWarningNoLine 265, 285, 349,
359, 363, 3262, 3270, 3302, 3310,
3392, 3596, 3637, 3644, 3660, 3675,
3691, 3708, 3713, 3736, 3747, 3751,
3764, 3768, 3782, 3786, 3823, 3835,
3843, 3851, 3861, 3869, 3877, 3887,
3895, 3903, 3912, 3920, 3928, 3938,
3946, 3954, 3964, 3972, 3980, 3990,
3999, 4008, 4019, 4028, 4037, 4048,
4057, 4066, 4080, 4088, 4096, 4106,
4114, 4122, 4130, 4138, 4146, 4154,
4162, 4170, 4178, 4186, 4194, 4201,
4206, 4211, 4216, 4221, 4226, 4231,
4236, 4241, 4246, 4251, 4256, 4261,
4599, 4676
- \mtcpagenumbers 228, 381, 430, 449, 4546
- \mtcPfont 1, 218, 563, 4486
- \mtcpnumwidth 2807, 2841
- \mtcprepare 1, 2467, 4462
- \mtcprepare@ 2467
- \mtcrule 2505, 4528
- \mtcsecondpartfalse . 108, 113, 119, 127,
4595
- \mtcsecondparttrue 111, 117, 124
- \mtcselectlanguage 1,
174, 3340, 3414, 4526, 4751, 4753,
4757, 4778, 4799, 4857, 4878, 4882,
4886, 5040, 5061, 5156, 5160, 5165,
5170, 5365, 5415, 5509, 5586, 5590,
5594, 5969, 5991, 5995, 6143, 6200,
6297, 6358, 6362, 6382, 6386, 6390,
6550, 7064, 7068, 7072, 7076, 7129,
7156
- \mtcsetdepth . . 1, 1448, 1465, 1532, 2226,
3142, 3149, 3164, 3170, 3175, 3178,
4508

- \mtcsetfeature . 1, 497, 3088, 3097, 3114, 3131, 3136, 4508
 - \mtcsetfont 1, 1279, 1280, 1315, 1318, 1321, 1361, 1364, 1367, 1449–1453, 1455–1458, 2224, 2595, 2596, 2603, 2620, 2715, 2720, 4508
 - \mtcsetformat . . . 1, 647, 720, 788, 2835, 2880, 2889, 2905, 2919, 2924, 4508
 - \mtcsetpagenumber@ 2950
 - \mtcsetpagenumbers . 1, 2932, 2959, 2978, 4508
 - \mtcsetpagenumbers@ 2932, 2950
 - \mtcsetrules . . . 1, 3001, 3028, 3047, 4508
 - \mtcsetrules@ 3001, 3019
 - \mtcsettitle . . . 1, 174, 1287, 2762, 2763, 2770, 2786, 2792, 4508
 - \mtcsettitlefont . . 1, 1285, 2726, 2727, 2734, 2750, 4508
 - \mtcSfont 1, 216, 563, 4486
 - \mtcskip 1, 9, 320, 322, 505, 571, 601, 620, 646, 672, 674, 681, 741, 743, 806, 808, 843, 845, 868, 870, 942, 946, 1006, 1029, 1033, 1095, 1127, 1149, 1464, 1531, 1776, 4424
 - \mtcskipamount 1, 9, 4349
 - \mtcSPfont 1, 218, 563, 4486
 - \mtcSSfont 1, 217, 563, 4486
 - \mtcSSSfont 1, 218, 563, 4486
 - \mtcsv@listoffigures 291
 - \mtcsv@listoftables 291
 - \mtcsv@tableofcontents 291
 - \mtctitle 1, 612, 613, 683, 751, 1717, 4751, 4768, 4789, 4810, 4828, 4846, 4868, 4897, 4914, 4931, 4949, 4965, 4981, 4997, 5013, 5030, 5051, 5078, 5108, 5130, 5146, 5181, 5219, 5238, 5255, 5272, 5289, 5312, 5338, 5355, 5376, 5396, 5427, 5465, 5482, 5499, 5520, 5543, 5576, 5605, 5620, 5637, 5652, 5670, 5696, 5736, 5776, 5808, 5918, 5947, 5981, 6006, 6023, 6041, 6059, 6133, 6154, 6171, 6190, 6212, 6229, 6253, 6270, 6287, 6308, 6331, 6348, 6372, 6401, 6416, 6435, 6451, 6466, 6482, 6503, 6522, 6540, 6561, 6578, 6596, 6614, 6629, 6655, 6695, 6736, 6762, 6830, 6847, 6864, 6885, 6906, 6923, 6944, 6959, 6975, 6996, 7013, 7030, 7054, 7091, 7119, 7146, 7167, 7182, 7197, 7212
 - \mtctocrmarg 2817, 2842
 - \mtifont 1, 563, 613, 683, 751, 4486
 - mu.tex **133, 216**
 - \mz 6626, 6627, 6630, 6631, 6634, 6635
- N**
- \n@mti . . . 575, 613, 683, 751, 895, 912, 929
 - \n@pti 1260, 1267, 1351, 1424, 1499, 1585, 1602, 1619
 - \n@sti 1900, 1938, 2005, 2072, 2198, 2215, 2232
 - naustrian.mld **473**
 - needspace **54**
 - \NeedsTeXFormat 2, 4345, 4678, 4689
 - \newblock 2306, 2308, 2314, 2315
 - \newbox 321, 325
 - \newcommand 274, 300, 311, 549, 1222, 1443, 1498, 1509, 1884, 2255, 2596, 2727, 2763, 2881, 2932, 2950, 3001, 3019, 3089, 3142, 3180, 3181, 3213, 3214, 3246, 3247, 3279, 3319, 3377, 4428–4430, 4614, 4623, 4632, 4658, 4659, 4679, 4681, 4683, 4685
 - \newcounter 513, 515, 547, 553, 1206, 1212, 1871, 1883, 1887
 - \newif 35–75, 77, 79, 81, 83, 85, 87–108, 130, 133, 134, 136, 176, 227–229, 334, 367, 720, 920, 2595, 2726, 2762, 2880, 2930, 2931, 2999, 3000, 3088, 3141, 3179, 3194, 3227, 3278, 3318, 3340, 3394, 3411, 3586–3589, 4591–4595, 4597
 - \newlength 9, 1231, 1889
 - \newread 318, 361
 - \newwrite 8, 506
 - newzealand.mld **473**
 - ngerman.mld **473**
 - ngermanb.mld **473**
 - ngermanb2.mld **474**
 - nocheckfiles **31, 51**
 - \nocite 827, 1113
 - nohints **49**
 - \noindent 629, 699, 767, 1366, 1439, 1514, 1954, 2020, 2087
 - nolistfiles **49**
 - \nomlfpagenumbers . . 232, 417, 430, 436, 4546
 - \nomlfrule 2505, 4528
 - \nomltpagenumbers . . 233, 423, 430, 436, 442, 4546
 - \nomltrule 2505, 4528
 - \nomtcpagenumbers . . 231, 381, 430, 1715, 4546
 - \nomtcrule 230, 1714, 2505, 4528
 - \noopsort 828, 1114
 - \noplfpagenumbers 417, 441, 4546

- \noplfrule 2505, 4528
 - \nopltpagenumbers 423, 447, 4546
 - \nopltrule 2505, 4528
 - \noptcpagenumbers 403, 435, 4546
 - \noptcrule 229, 2505, 4528
 - \normalcolor 846
 - \normalfont 846, 1726, 1736
 - \normalsize 1247–1249, 1274, 1275, 1279,
1280, 1283, 1284, 1315, 1318, 1321,
1361, 1364, 1367
 - norsk.mld **475**
 - norsk2.mld **475**
 - \noslfpagenumbers 417, 439, 4546
 - \noslfrule 2505, 4528
 - \nosltpagenumbers 423, 445, 4546
 - \nosltrule 2505, 4528
 - \nostcpagenumbers 393, 433, 4546
 - \nostcrule 2505, 4528
 - notoccite **49, 240**
 - \numberline 259, 2146
 - \nun ... 5461, 5465, 5469, 5912, 5919, 5926
 - nynorsk.mld **476**
 - nynorsk2.mld **476**
- O**
- \oe 2106, 2136
 - \openin 1022, 1089, 1148
 - \openout 255, 259, 332, 507,
960, 991, 1036, 1067, 1102, 1126,
1161, 1657, 1671, 1744, 1768, 1804,
1828, 1864, 2272, 2290, 2341, 2367,
2400, 2426, 2459, 4286
- P**
- \PackageError 4676
 - \PackageInfo 4676
 - \PackageWarning 4676
 - \PackageWarningNoLine 4676
 - \pagebreak 662, 734, 802, 1401, 1475, 1550,
1987, 2054, 2122
 - \pagenumbering 1396, 1402
 - \pageref 1576
 - \pagestyle 1802
 - \paragraph .. 344, 366, 385, 418, 585, 607,
627, 655, 973, 1674, 2274
 - \paragraph@mtcck 2647
 - \parbox 1514, 1770
 - \parsep 586, 1319, 1916
 - \parskip 9, 1319, 1360, 1433, 1508, 1948,
2014, 2081
 - part **238**
 - \part .. 1, 183, 186, 191, 244, 264, 372, 425,
497, 522, 613, 670, 815, 941, 1028,
1190, 1194, 1200, 1311, 1357, 1644,
1755, 1815, 1896, 1953, 1969, 1982,
3268, 3282, 3308, 3322, 3739, 4331
 - \part@mtcck 2642, 2669, 2677, 2683, 2689,
2698, 2705
 - \partbegin 1, 455, 1557, 1745, 1805, 1865,
4437
 - \partend ... 1557, 1734, 1795, 1855, 4437
 - \partlof 1, 246, 374, 499, 615, 673,
942, 1029, 1260, 1330, 1403, 3844,
3922, 4000, 4444
 - \partlof@ 859, 1403, 1404
 - \partlof@mtcck 2636, 2660
 - partlofdepth **288, 298**
 - \partlot 1, 247, 500, 616, 675,
942, 1029, 1260, 1330, 1477, 3852,
3930, 4009, 4444
 - \partlot@ 859, 1477, 1478
 - \partlot@mtcck 2639, 2664
 - partlotdepth **288, 301**
 - \partopsep 590, 1319, 1915
 - \parttoc .. 1, 30, 184, 189, 194, 245, 373,
426, 498, 614, 671, 942, 1029, 1191,
1195, 1201, 1260, 1313, 1316, 1319,
1322, 1330, 1359, 1362, 1365, 1368,
1956, 1970, 1983, 3836, 3914, 3991,
4444
 - \parttoc@ 859, 1330, 1331
 - \parttoc@mtcck 2633, 2654
 - parttocdepth **31, 289, 295**
 - \pchapter 1552, 1674, 4463
 - \pfill 724, 924
 - placeins **50, 237**
 - plainurl.bst **216**
 - \PLF@ 1791, 1795
 - \PLF@contentsline ... 1755, 1794, 1795
 - \PLF@explist 1753, 1779
 - \PLF@line 1789, 1791
 - \PLF@list 1750, 1753, 1779, 1795
 - \PLF@lof 1753, 1780, 1795
 - \PLF@loop 1750, 1780
 - \PLF@next 1574, 1750, 1779
 - \plf@pgno 440, 441, 1461
 - \PLF@read 1780, 1789, 1795
 - \plf@rule .. 542, 1436, 1472, 2514, 2517,
2520
 - \plf@setform 1461, 2850
 - \PLF@test 1791, 1793, 1795
 - \PLF@WriteContentsLine 1755
 - \plfdotsep 2825, 2853, 2873, 2878
 - \plffont .. 1, 1233, 1246, 1349, 1433, 4474
 - \plfname 1755, 1758
 - \plfpagenumbers ... 417, 440, 449, 4546
 - \plfpnumwidth ... 2807, 2851, 2871, 2876
 - \plfrule 2505, 4528

- \plfSfont 1, 1233, 1246, 1350, 4474
- \plftitle 1, 1418, 1424, 1436, 4751, 4765,
 - 4786, 4807, 4825, 4843, 4865, 4894,
 - 4911, 4928, 4946, 4962, 4978, 4994,
 - 5010, 5027, 5048, 5071, 5103, 5127,
 - 5143, 5178, 5216, 5235, 5252, 5269,
 - 5286, 5305, 5335, 5352, 5373, 5389,
 - 5422, 5462, 5479, 5496, 5517, 5536,
 - 5569, 5602, 5617, 5634, 5649, 5667,
 - 5688, 5692, 5728, 5768, 5805, 5913,
 - 5940, 5978, 6003, 6020, 6038, 6056,
 - 6130, 6131, 6151, 6168, 6187, 6209,
 - 6226, 6250, 6267, 6284, 6305, 6328,
 - 6345, 6369, 6398, 6413, 6432, 6448,
 - 6471, 6487, 6500, 6519, 6537, 6558,
 - 6575, 6593, 6626, 6647, 6687, 6729,
 - 6759, 6827, 6844, 6861, 6881, 6903,
 - 6920, 6941, 6956, 6972, 6993, 7010,
 - 7027, 7051, 7098, 7116, 7143, 7164,
 - 7179, 7194, 7209
- \plftocrmarg 2816, 2852, 2872, 2877
- \PLT@ 1851, 1855
- \PLT@contentsline . . . 1815, 1854, 1855
- \PLT@explist 1813, 1839
- \PLT@line 1849, 1851
- \PLT@list 1810, 1813, 1839, 1855
- \PLT@loop 1810, 1840
- \PLT@lot 1813, 1840, 1855
- \PLT@next 1578, 1810, 1839, 2191
- \plt@pgno 446, 447, 1536
- \PLT@read 1840, 1849, 1855
- \plt@rule 542, 1515, 1548, 2523, 2526, 2529
- \plt@setform 1536, 2865
- \PLT@test 1851, 1853, 1855
- \PLT@WriteContentsLine 1815
- \pltdotsep 2825, 2868
- \pltfont . 1, 1233, 1246, 1351, 1508, 4474
- \pltname 1815, 1818
- \pltpagenumbers 423, 446, 449, 4546
- \pltpnumwidth 2807, 2866
- \pltrule 2505, 4528
- \pltSfont 1, 1233, 1246, 1352, 4474
- \plttitle 1, 1493, 1499, 1515, 4751, 4766,
 - 4787, 4808, 4826, 4844, 4866, 4895,
 - 4912, 4929, 4947, 4963, 4979, 4995,
 - 5011, 5028, 5049, 5074, 5105, 5128,
 - 5144, 5179, 5217, 5236, 5253, 5270,
 - 5287, 5308, 5336, 5353, 5374, 5392,
 - 5424, 5463, 5480, 5497, 5518, 5539,
 - 5572, 5603, 5618, 5635, 5650, 5668,
 - 5732, 5772, 5806, 5915, 5943, 5979,
 - 6004, 6021, 6039, 6057, 6152, 6169,
 - 6188, 6210, 6227, 6251, 6268, 6285,
 - 6306, 6329, 6346, 6370, 6399, 6414,
 - 6433, 6449, 6472, 6488, 6501, 6520,
 - 6538, 6559, 6576, 6594, 6611, 6612,
 - 6627, 6651, 6691, 6732, 6760, 6828,
 - 6845, 6862, 6883, 6904, 6921, 6942,
 - 6957, 6973, 6994, 7011, 7028, 7052,
 - 7100, 7117, 7144, 7165, 7180, 7195,
 - 7210
- \plttocrmarg 2816, 2867
- pmk 216
- polish.mld 477
- polish2.mld 477
- polski.mld 478
- portuges.mld 478
- portuguese.mld 478, 479
- \printindex 1, 832, 1118
- \ProcessOptions 4271
- \ProcessOptions* 4271
- \protect 1
- \providecommand . 1272, 1273, 1281, 1282,
 - 2016
- \ProvidesFile . . . 4342, 4752, 4756, 4760,
 - 4777, 4781, 4798, 4802, 4819, 4837,
 - 4856, 4860, 4877, 4881, 4885, 4889,
 - 4906, 4923, 4940, 4958, 4974, 4990,
 - 5006, 5022, 5039, 5043, 5060, 5064,
 - 5097, 5123, 5139, 5155, 5159, 5163,
 - 5168, 5173, 5190, 5200, 5211, 5228,
 - 5247, 5264, 5281, 5298, 5321, 5347,
 - 5364, 5368, 5385, 5413, 5418, 5440,
 - 5449, 5458, 5474, 5491, 5508, 5512,
 - 5529, 5552, 5585, 5589, 5593, 5597,
 - 5614, 5629, 5646, 5661, 5679, 5719,
 - 5759, 5799, 5817, 5829, 5842, 5854,
 - 5866, 5875, 5887, 5899, 5908, 5933,
 - 5968, 5972, 5990, 5994, 5998, 6015,
 - 6032, 6050, 6068, 6077, 6089, 6101,
 - 6113, 6126, 6142, 6146, 6163, 6180,
 - 6199, 6203, 6221, 6238, 6245, 6262,
 - 6279, 6296, 6300, 6317, 6323, 6340,
 - 6357, 6361, 6365, 6381, 6385, 6389,
 - 6393, 6410, 6425, 6444, 6460, 6479,
 - 6495, 6512, 6531, 6549, 6553, 6570,
 - 6587, 6605, 6623, 6638, 6678, 6718,
 - 6755, 6771, 6780, 6790, 6798, 6806,
 - 6814, 6822, 6839, 6856, 6873, 6898,
 - 6915, 6932, 6953, 6968, 6984, 7005,
 - 7022, 7039, 7046, 7063, 7067, 7071,
 - 7075, 7079, 7086, 7110, 7128, 7132,
 - 7139, 7155, 7159, 7176, 7191, 7206
- \ProvidesPackage . . . 2, 4345, 4677, 4690
- \ps@chapterverso 1997, 2006
- \ps@empty 1997, 2006
- \psect 1552, 4463
- ptc 247, 289, 302

- \PTC@ 1730, 1734
 - \PTC@contentsline . . . 1644, 1733, 1734
 - \PTC@explist 1642, 1718
 - \PTC@line 1728, 1730
 - \PTC@list 1639, 1642, 1718, 1734
 - \PTC@loop 1639, 1719
 - \PTC@next 1570, 1639, 1718
 - \ptc@pgno 434, 435, 1384
 - \PTC@read 1719, 1728, 1734
 - \ptc@rule . . 1228, 1368, 1398, 1547, 2505,
2508, 2511
 - \ptc@setform 1384, 2835
 - \ptc@spart 1557
 - \PTC@test 1730, 1732
 - \PTC@toc 1642, 1719, 1734
 - \ptc@verse 1319
 - ptc@verse (environment) 1319, 1382, 1384,
1455, 1461, 1530, 1536, 2835, 2850,
2865
 - \ptcCfont 1, 1246, 4474
 - \ptcdotsep 2825, 2838
 - \ptcfont . 1, 1233, 1246, 1345–1352, 1360,
4474
 - \ptcindent 1231, 1319, 1377, 1450, 1525,
4353
 - \ptcname 1644, 1661
 - \ptcpagenumbers . . . 403, 434, 449, 4546
 - \ptcPfont 1, 1233, 1246, 1347, 4474
 - \ptcpnumwidth 2807, 2836
 - \ptcrule 2505, 4528
 - \ptcSfont 1, 1233, 1246, 4474
 - \ptcSPfont . . . 1, 1233, 1246, 1348, 4474
 - \ptcSSfont . . . 1, 1233, 1246, 1345, 4474
 - \ptcSSSfont . . 1, 1233, 1246, 1346, 4474
 - \ptctitle 1, 1345, 1351, 1368, 4751, 4764,
4785, 4806, 4824, 4842, 4864, 4893,
4910, 4927, 4945, 4961, 4977, 4993,
5009, 5026, 5047, 5069, 5101, 5126,
5142, 5177, 5215, 5234, 5251, 5268,
5285, 5302, 5334, 5351, 5372, 5388,
5421, 5461, 5478, 5495, 5516, 5533,
5566, 5601, 5616, 5633, 5648, 5666,
5685, 5725, 5765, 5804, 5911, 5937,
5977, 6002, 6019, 6037, 6055, 6129,
6150, 6167, 6186, 6208, 6225, 6249,
6266, 6283, 6304, 6327, 6344, 6368,
6397, 6412, 6431, 6447, 6470, 6486,
6499, 6518, 6536, 6557, 6574, 6592,
6610, 6625, 6644, 6684, 6723, 6726,
6758, 6826, 6843, 6860, 6879, 6902,
6919, 6937, 6939, 6955, 6971, 6989,
6991, 7009, 7026, 7050, 7097, 7115,
7142, 7163, 7178, 7193, 7208
 - \ptctocrmarg 2816, 2837
 - \ptifont 1, 1233, 1246, 1270, 1284, 1298,
1368, 1436, 1515, 4474
- ## Q
- \qaf . . . 4843, 4844, 4847, 4848, 4851, 4852
 - \quad 1541, 1551, 1556, 1566, 1729
 - quotchap **50, 72, 242, 385**
- ## R
- \r@mti . . 575, 613, 683, 751, 895, 912, 929
 - \r@pti 1260, 1284, 1351, 1424, 1499, 1585,
1602, 1619
 - \r@sti 1900, 1938, 2005, 2072, 2198, 2215,
2232
 - \ra . . . 4842, 4846, 4850, 5461, 5465, 5469
 - \raggedbottom 220, 480
 - \raggedright 623, 693,
761, 1278, 1360, 1433, 1471, 1508,
1736, 1948, 2014, 2081
 - \read . . . 230, 318, 1032, 1098, 1157, 1729,
1790, 1850, 2329, 2388, 2447
 - README **216**
 - \ref . . . 1485, 1576–1580, 1583, 1586, 2105
 - \refstepcounter 2146, 4717
 - \removelastskip 12
 - \renewcommand 187, 188, 192,
193, 215–217, 475, 477, 1274, 1275,
1283, 1284, 1437, 1717, 1781, 1786,
1800, 1809, 1810, 2022, 4695
 - report **248, 250**
 - \RequirePackage 2, 197, 4272, 4278, 4348,
4692
 - \reset@font 358, 625, 632, 636, 695,
702, 706, 763, 770, 774, 1181, 1186,
1199, 1201, 1362, 1370, 1374, 1435,
1442, 1446, 1510, 1517, 1521, 1950,
1958, 1962, 2016, 2023, 2028, 2083,
2090, 2095
 - \resh . . 5913–5915, 5920–5922, 5927–5929
 - \rightmargin . . . 595, 641, 711, 779, 1328,
1379, 1452, 1527, 1920, 1967, 2033,
2100
 - \rmfamily 475, 563, 564, 573, 574,
1234, 1235, 1244, 1247–1249, 1258,
1283, 1284, 1315, 1361, 1449–1453,
1455–1458, 1890, 1891, 1899
 - rmk **216**
 - \Roman . 5304, 5307, 5310, 5333, 5535, 5538,
5541, 5565
 - romanian.mld **479**
 - romanian2.mld **480**
 - romanian3.mld **480**
 - romannum **50, 72, 243, 386**
 - \ru 1498, 1522

- rubber@rubber **58**
 \rule 322, 326, 330, 666,
 1229, 1753, 1762, 1888, 2511–2513,
 2520–2522, 2529–2531
 russian-cca.mld **485**
 russian-cca.mlo **485**
 russian-cca1.mld **485**
 russian-cca1.mlo **485**
 russian-lh.mld **486**
 russian-lh.mlo **486**
 russian-lhcyralt.mld **486**
 russian-lhcyralt.mlo **486**
 russian-lhcyrkoi.mld **487**
 russian-lhcyrkoi.mlo **487**
 russian-lhcyrwin.mld **487**
 russian-lhcyrwin.mlo **487**
 russian.mld **481**
 russian2m.mld **481**
 russian2o.mld **482**
 russianb.mld **483**
 russianc.mld **484**
- S**
- samepage **294, 297, 300**
 samepage (environment) 613, 657, 683, 730,
 751, 798, 1351, 1398, 1424, 1472,
 1499, 1547, 1938, 1984, 2005, 2051,
 2072, 2119
 samin.mld **488**
 \scan@allowedtrue 721, 921
 \chapter 1, 2123
 scottish.mld **488**
 scr.tex **113, 216**
 scrartcl **50**
 scrbook **50, 125**
 scrreprt **50**
 \scshape 1274, 1275, 1279, 1280, 1283–1285
 second.tex **106, 216**
 \sectbegin . 2175, 2332, 2391, 2450, 4437
 \sectend . . 2175, 2332, 2391, 2450, 4437
 section **50, 238**
 \section 1, 281, 297,
 815, 973, 1674, 2136, 2257, 2352,
 2411, 3739, 3773, 4337
 \section@mtcck . . 2644, 2691, 2700, 2707
 \sectionfont 2023
 \sectlof 1, 505, 572, 601, 620, 646, 681, 700,
 946, 1006, 1033, 1095, 1127, 1149,
 1988, 3825, 3896, 3974, 4058, 4456
 \sectlof@ 869, 1988, 1989
 \sectlof@mtcck 2638, 2697
 sectlofdepth **313, 320**
 \sectlot . . . 1, 701, 946, 1006, 1033, 1095,
 1127, 1149, 2055, 2379, 2399, 3825,
 3904, 3982, 4067, 4456
 \sectlot@ 869, 2055, 2056
 \sectlot@mtcck 2641, 2704
 sectlotdepth **313, 323**
 sectsty **50, 67, 237, 242, 384**
 \secttoc 1, 504, 570, 601, 618,
 619, 645, 680, 699, 946, 1006, 1033,
 1095, 1127, 1149, 1922, 2232, 2241,
 3888, 3966, 4049, 4456
 \secttoc@ 869, 1922, 1923
 \secttoc@mtcck 2635, 2688
 secttocdepth **31, 313, 317**
 \seen 4842, 4846, 4850, 5461–5463,
 5465–5467, 5469–5471
 \selectlanguage 6242, 7083, 7136
 serbian.mld **489**
 serbianc.mld **489**
 \setcounter . . 1, 1206, 1212, 1570, 1574,
 1578, 1871, 3160, 3180, 3194, 3227,
 3378
 \setvruler 180
 \sf 217, 477
 \sffamily 1729, 1731, 1738, 2021
 sfheaders **51, 72, 243, 386**
 \sh 5126, 5130, 5134
 \sheen . 4843, 4847, 4851, 5462, 5466, 5470
 \shin . . 5913, 5915, 5920, 5922, 5927, 5929
 shorttext **50, 56**
 \sixt@an 318
 \SLF@ 2389, 2391
 \SLF@contentsline 2352, 2391
 \SLF@explist 2350, 2377
 \SLF@line 2387
 \SLF@list 2347, 2350, 2377, 2391
 \SLF@lof 2350, 2378, 2391
 \SLF@loop 2347, 2378
 \SLF@next 2187, 2347, 2377
 \slf@pgno 438, 439, 2042
 \SLF@read 2378, 2387, 2391
 \slf@rule 542, 2019, 2052, 2516, 2519, 2522
 \slf@setform 2042, 2860
 \SLF@test 2387, 2391
 \slfdotsep 2825, 2863
 \slffont 1, 1890, 2014, 4497
 \slflotrmarg 2816
 \slfname 2352, 2357
 \slfpagenumbers . . . 417, 438, 449, 4546
 \slfpnumwidth 2807, 2861
 \slfrule 2505, 4528
 \slfsfont 1, 1890, 4497
 \slftitle 1, 2003, 2005, 2019, 4751, 4773,
 4794, 4815, 4833, 4851, 4873, 4902,

- 4919, 4936, 4954, 4970, 4986, 5002,
5018, 5035, 5056, 5089, 5117, 5135,
5151, 5186, 5224, 5243, 5260, 5277,
5294, 5317, 5343, 5360, 5381, 5405,
5434, 5470, 5487, 5504, 5525, 5548,
5581, 5610, 5625, 5642, 5657, 5675,
5710, 5714, 5750, 5790, 5813, 5927,
5960, 5986, 6011, 6028, 6046, 6064,
6138, 6139, 6159, 6176, 6195, 6217,
6234, 6258, 6275, 6292, 6313, 6336,
6353, 6377, 6406, 6421, 6440, 6456,
6475, 6491, 6508, 6527, 6545, 6566,
6583, 6601, 6634, 6669, 6709, 6747,
6767, 6835, 6852, 6869, 6893, 6911,
6928, 6949, 6964, 6980, 7001, 7018,
7035, 7059, 7104, 7124, 7151, 7172,
7187, 7202, 7217
- \slftocrmarg 2821, 2862
\sloppy 248, 501
slovak.mld **490**
slovene.mld **490**
\slshape 574, 1450, 1452, 1455, 1457
\SLT@ 2448, 2450
\SLT@contentsline 2411, 2450
\SLT@explist 2409, 2436
\SLT@line 2446
\SLT@list 2406, 2409, 2436, 2450
\SLT@loop 2406, 2437
\SLT@lot 2409, 2437, 2450
\SLT@next 2194, 2406, 2436
\slt@pgno 444, 445, 2109
\SLT@read 2437, 2446, 2450
\slt@rule 542, 2088, 2120, 2525, 2528, 2531
\slt@setform 2109, 2875
\SLT@test 2446, 2450
\sltdotsep 2825
\sltfont 1, 1890, 2081, 4497
\sltlotrmarg 2816
\sltname 2411, 2416
\sltpagenumbers 423, 444, 449, 4546
\sltpnumwidth 2807
\sltrule 2505, 4528
\sltSfont 1, 1890, 4497
\slttitle 1, 2070, 2072, 2086, 4751, 4774,
4795, 4816, 4834, 4852, 4874, 4903,
4920, 4937, 4955, 4971, 4987, 5003,
5019, 5036, 5057, 5092, 5119, 5136,
5152, 5187, 5225, 5244, 5261, 5278,
5295, 5318, 5344, 5361, 5382, 5408,
5436, 5471, 5488, 5505, 5526, 5549,
5582, 5611, 5626, 5643, 5658, 5676,
5754, 5794, 5814, 5929, 5963, 5987,
6012, 6029, 6047, 6065, 6160, 6177,
6196, 6218, 6235, 6259, 6276, 6293,
6314, 6337, 6354, 6378, 6407, 6422,
6441, 6457, 6476, 6492, 6509, 6528,
6546, 6567, 6584, 6602, 6619, 6620,
6635, 6673, 6713, 6750, 6768, 6836,
6853, 6870, 6895, 6912, 6929, 6950,
6965, 6981, 7002, 7019, 7036, 7060,
7106, 7125, 7152, 7173, 7188, 7203,
7218
- \slttocrmarg 2824
spanish.mld **431, 491**
spanish2.mld **492**
spanish3.mld **492**
spanish4.mld **493**
\spart 1
\star 1781, 1786
\starchapter 880, 991, 1698
\starparagraph 1005, 1698, 2305
\starpart 1659
\starsection 1005, 1698, 2136, 2290
\starsubparagraph ... 1005, 1698, 2305
\starsubsection 1005, 1698, 2305
\starsubsubsection .. 1005, 1698, 2305
stc **249, 313**
\STC@ 2330, 2332
\STC@contentsline 2257, 2332
\STC@explist 2255, 2318
\STC@line 2328, 2330
\STC@list 2252, 2255, 2318, 2332
\STC@loop 2252, 2319
\STC@next 2183, 2252, 2318
\stc@pgno 432, 433, 1972
\STC@read 2319, 2328, 2332
\stc@rule . 1879, 1953, 1984, 2051, 2086,
2119, 2507, 2510, 2513
\stc@sect 2175
\stc@setform 1972, 2845
\STC@test 2330, 2332
\STC@toc 2255, 2319, 2332
\stc@verse 1911
stc@verse (environment) 1911, 1970, 1972,
2036, 2042, 2103, 2109, 2845, 2860,
2875
\stcdotsep 2825, 2848
\stcfont 1, 475, 1890, 1948, 4497
\stcindent . 473, 1879, 1919, 1965, 2031,
2098, 2223, 4349
\stclotrmarg 2816
\stcname 2257, 2262, 2290
\stcpagenumbers 393, 432, 449, 4546
\stcPfont 1, 478, 1890, 4497
\stcpnumwidth 2807, 2846
\stcPSfont 1
\stcrule 2505, 4528
\stcSPfont 478, 1890, 4497

- \stcSSfont 1, 477, 1890, 4497
 - \stcSSSfont 1, 478, 1890, 4497
 - \stctitle 1, 1938, 1953, 4751, 4772, 4793,
4814, 4832, 4850, 4872, 4901, 4918,
4935, 4953, 4969, 4985, 5001, 5017,
5034, 5055, 5087, 5115, 5134, 5150,
5185, 5223, 5242, 5259, 5276, 5293,
5316, 5342, 5359, 5380, 5404, 5433,
5469, 5486, 5503, 5524, 5547, 5580,
5609, 5624, 5641, 5656, 5674, 5707,
5747, 5787, 5812, 5925, 5957, 5985,
6010, 6027, 6045, 6063, 6137, 6158,
6175, 6194, 6216, 6233, 6257, 6274,
6291, 6312, 6335, 6352, 6376, 6405,
6420, 6439, 6455, 6474, 6490, 6507,
6526, 6544, 6565, 6582, 6600, 6618,
6633, 6666, 6706, 6745, 6766, 6834,
6851, 6868, 6891, 6910, 6927, 6948,
6963, 6979, 7000, 7017, 7034, 7058,
7103, 7123, 7150, 7171, 7186, 7201,
7216
 - \stctocmarg 2818, 2847
 - \stepcounter 264, 297, 309, 519, 959, 991,
1066, 1125, 1557, 1646, 1660, 1699,
1757, 1817, 2260, 2290, 2355, 2414
 - \stifont . 1, 1890, 1953, 2019, 2086, 4497
 - subf.tex **114, 216**
 - subfig **31, 356**
 - \subfigbottomskip 1445, 1475
 - \subfigcapskip 1584
 - \subfigtopskip 1444, 1475
 - subfigure **31**
 - \subfigure . 1079, 1473, 1476, 1478, 1537,
1542, 1547, 1552, 1557, 1562, 1567,
1770, 2369
 - \subfigure@mtcck 2651, 2656, 2666, 2672,
2686, 2693, 2709
 - \subparagraph 420, 587, 609, 629, 665, 973,
1674, 2274
 - \subparagraph@mtcck 2648
 - \subref 1577–1580
 - \subsection . 313, 340, 360, 381, 407, 507,
509, 511, 513, 516, 518, 520, 522,
524, 526, 528, 530, 532, 534, 536,
538, 540, 542, 544, 546, 548, 550,
552, 554, 556, 558, 560, 562, 564,
566, 574, 596, 603, 611, 623, 639,
651, 667, 682, 684, 686, 695, 703,
705, 707, 709, 947, 956, 973, 1007,
1016, 1034, 1043, 1096, 1105, 1128,
1137, 1150, 1159, 1328, 1330, 1334,
1336, 1374, 1376, 1380, 1382, 1674,
1852, 1857, 1869, 1874, 2044, 2063,
2082, 2089, 2099, 2103, 2115, 2121,
2132, 2139, 2145, 2188, 2233, 2238,
2242, 2247, 2274
 - \subsection@mtcck 2645
 - \subsubsection 342, 363, 383,
409, 583, 605, 625, 653, 973, 1025,
1026, 1674, 2067, 2072, 2077, 2126,
2150, 2156, 2161, 2234, 2236, 2243,
2245, 2274
 - \subsubsection@mtcck 2646
 - \subtable 1138, 1830, 2428
 - \subtable@mtcck . 2652, 2658, 2662, 2674,
2680, 2695, 2702
 - \sv@chapter 805
 - \sv@minilof@ 849
 - \sv@minilot@ 849
 - \sv@minitoc@ 849
 - \sv@part 1557
 - \sv@partlof@ 859
 - \sv@partlot@ 859
 - \sv@parttoc@ 859
 - \sv@sect 2129, 2135
 - \sv@sectlof@ 869
 - \sv@sectlot@ 869
 - \sv@secttoc@ 869
 - \sv@spart 1557
 - swedish.mld **493**
 - swedish2.mld **494**
- T**
- T **270**
 - \taa . . . 5461–5463, 5465–5467, 5469–5471
 - \table 1138, 1830, 2428
 - \table@mtcck 2650, 2657, 2661, 2673, 2679,
2694, 2701
 - \tableofcontents 1, 11, 240, 291, 733, 935,
1213, 1246, 1290, 1310, 1356, 1398,
1460, 1597, 1651, 1894, 1952, 2031,
2171, 2258, 3795, 3992, 4021, 4050
 - tabular (environment) 613, 683, 751, 1368,
1436, 1515, 1953, 2019, 2086
 - \tav . . . 5911, 5913, 5915, 5916, 5918, 5920,
5922, 5923, 5925, 5927, 5929, 5930
 - tbi.tex **118, 216**
 - \testtemp 1497, 1499–1502, 1505
 - \tet 5916, 5923, 5930
 - \textheight 2220
 - \textit 1756
 - \textsc . 765, 2025, 2039, 2041, 2096, 2108,
2255
 - \textsf 762
 - \textsl 765, 967, 1054
 - \textsuperscript 2016
 - \texttt . . 789, 893, 896, 901, 904, 909, 912,
989, 1078

- `\textwidth` 2218
`\tf@mtc` 8, 227, 332, 946, 956, 960,
 991, 1036, 1067, 1102, 1126, 1161,
 1177, 1197, 1636, 1644, 1670, 1671,
 1743, 1744, 1755, 1767, 1768, 1803,
 1804, 1815, 1827, 1828, 1863, 1864,
 2249, 2257, 2271, 2272, 2302, 2303,
 2340, 2341, 2352, 2366, 2367, 2399,
 2400, 2411, 2425, 2426, 2458, 2459,
 4281, 4463
`thai.ldf` **494**
`thai.mld` **494**
`thai.mlo` **494**
`thaicjk.ldf` **494**
`thailatex` **211, 494**
`\The@mtc` . 542, 601, 603, 671, 673, 739, 741
`\thechapter` 1729, 1810, 4737, 4740
`\themtc` .. 542, 956, 960, 991, 1064, 1067,
 1123, 1126
`\Thepart` 1212, 1331, 1404, 1478
`\theptc` 1212, 1644, 1663, 1664, 1667,
 1668, 1755, 1760, 1761, 1764, 1765,
 1815, 1820, 1821, 1824, 1825
`\thesection` 1809, 1810, 2022
`\thestc` 1923, 1989, 2056,
 2113, 2257, 2264, 2265, 2268, 2269,
 2290, 2352, 2359, 2360, 2363, 2364,
 2411, 2418, 2419, 2422, 2423
`\thispageminilofstyle` .. 484, 681, 4573
`\thispageminilotstyle` .. 484, 749, 4573
`\thispageminitocstyle` .. 484, 611, 4573
`\thispagepartlofstyle` . 484, 1418, 4564
`\thispagepartlotstyle` . 484, 1493, 4564
`\thispageparttocstyle` . 484, 1345, 4564
`\thispagesectlofstyle` . 484, 2002, 4582
`\thispagesectlotstyle` . 484, 2069, 4582
`\thispagesecttocstyle` . 484, 1936, 4582
`\thispagestyle` 484
`tight` **39, 59, 241, 263, 292**
`\tightmtcfalse` 1, 133, 3383, 4592
`\tightmtctrue` 1, 3382
`\title` 2024
`titlesec` **50**
`titletoc` **50**
`tlc.tex` **132, 216**
`tocbibind` **47, 50, 96, 101, 105**
`tocdepth` **265, 269**
`\toclevel@chapter` 145
`\toclevel@paragraph` 149
`\toclevel@part` 144
`\toclevel@section` 146
`\toclevel@starchapter` 145
`\toclevel@starparagraph` 149
`\toclevel@starpart` 144
`\toclevel@starsection` 146
`\toclevel@starsubparagraph` 150
`\toclevel@starsubsection` 147
`\toclevel@starsubsubsection` 148
`\toclevel@subparagraph` 150
`\toclevel@subsection` 147
`\toclevel@subsubsection` 148
`\toclevel@xchapter` 141
`\toclevel@xpart` 140
`\toclevel@xsect` 142
`\toclevel@xsection` 143
`tocloft` **50, 61**
`TODO` **216**
`\topsep` 586, 1319
`tsfc.tex` **116, 216**
`\ttfamily` 1318, 1321, 1364, 1367
`turkish.mld` **495**
- U**
- `uighur.mld` **495**
`uighur2.mld` **495**
`uighur3.mld` **496**
`UKenglish.mld` **496**
`ukraineb.mld` **496**
`ukrainian.mld` **496**
`undotted` **59**
`\undottedmtcfalse` 367, 3389, 4594
`\undottedmtctrue` 3388
`\unhcopy` 324, 328
`\unitlength` . 328, 334, 347, 388, 412, 437,
 577, 590, 632, 658, 689
`\unpenalty` 331
`unsrt` **49**
`uppersorbian.mld` **497**
`\upshape` .. 475, 563, 564, 573, 1234, 1235,
 1244, 1247–1249, 1258, 1315, 1318,
 1321, 1361, 1364, 1367, 1449, 1451,
 1453, 1456, 1458, 1890, 1891, 1899
`\url` 897, 905, 913
`USenglish.mld` **498**
`\usepackage` 1,
 3–5, 54, 116, 176–178, 208–211,
 468–470, 716–718, 726, 727, 918,
 926, 927, 1175–1180, 1208, 1209,
 1238–1241, 1243, 1271, 1277, 1278,
 1304–1306, 1342–1344, 1390–1394,
 1435, 1440, 1441, 1494–1496, 1592,
 1593, 1643–1645, 1647, 1692, 1694–
 1701, 1711, 1712, 1889, 1890, 1940–
 1943, 2008–2015, 2168, 2222, 2252,
 2254, 2324, 2325, 2371, 2372, 4610
`usorbian.mld` **498**

V

`\value` . . . 116, 4074, 5302, 5305, 5308, 5325,
 5533, 5536, 5539, 5556, 5566, 5569,
 5572
`varsects` **50, 69, 242, 385**
`\vav` . . . 5911, 5914, 5916, 5918, 5921, 5923,
 5925, 5928, 5930
`\vbox` 1474, 1477, 1479, 1511, 1518
`\vfil` 1474, 1477, 1479, 1512, 1517
`\vfill` 1757, 1774
`vietnam.mld` **498, 499**
`vietnam.sty` **498**
`vietnamese.mld` **499**
`\vref` 1488, 1489
`\vrule` 1503, 1504, 1506

W

`\waw` . . . 4844, 4848, 4852, 5463, 5467, 5471
`welsh.mld` **499**
`while.tip` **245**
`\write` 332, 1177, 1197

X

`\xalx` . . 6758–6760, 6762–6764, 6766–6768,
 7208–7210, 7212–7214, 7216–7218
`xalx.mld` **464, 500**
`xalx2.mld` **500**
`xalx3.mld` **501**
`xchapter` **251**
`\xchapter` 803, 1063, 1122, 4463
`xmk` **216**
`xpart` **248**
`\xpart` 1212, 1756, 1816, 4463
`xsect` **249**
`\xsect` 2123, 2354, 2413, 4463
`\xsection` 2123

Y

`\ya` . . . 5462, 5463, 5466, 5467, 5470, 5471
`\yahamza` 4843, 4844, 4847, 4848, 4851, 4852
`\yod` . . . 5912–5915, 5919–5922, 5926–5929